



KATHOLIEKE UNIVERSITEIT LEUVEN  
FACULTEIT INGENIEURSWETENSCHAPPEN  
DEPARTEMENT ELEKTROTECHNIEK  
AFDELING ESAT-SCD: SISTA/COSIC/DOCARCH  
Kasteelpark Arenberg 10 – B-3001 Leuven

DESIGN AND EVALUATION OF DIGITAL SIGNAL  
PROCESSING ALGORITHMS FOR ACOUSTIC  
FEEDBACK AND ECHO CANCELLATION

Promotor:  
Prof. dr. ir. M. Moonen

Proefschrift voorgedragen tot  
het behalen van het doctoraat  
in de ingenieurswetenschappen  
door  
**Toon VAN WATERSHOOT**

Maart 2009





KATHOLIEKE UNIVERSITEIT LEUVEN  
FACULTEIT INGENIEURSWETENSCHAPPEN  
DEPARTEMENT ELEKTROTECHNIEK  
AFDELING ESAT-SCD: SISTA/COSIC/DOCARCH  
Kasteelpark Arenberg 10 – B-3001 Leuven

## DESIGN AND EVALUATION OF DIGITAL SIGNAL PROCESSING ALGORITHMS FOR ACOUSTIC FEEDBACK AND ECHO CANCELLATION

Jury:

Prof. em. dr. ir. Y. Willems, voorzitter  
Prof. dr. ir. M. Moonen, promotor  
Prof. dr. ir. J. Vandewalle, assessor  
Prof. dr. ir. H. Van hamme, assessor, secretaris  
Prof. dr. ir. S. H. Jensen  
(Aalborg Universitet, Denemarken)  
Prof. dr. ir. P. C. W. Sommen  
(Technische Universiteit Eindhoven, Nederland)

Proefschrift voorgedragen tot  
het behalen van het doctoraat  
in de ingenieurswetenschappen  
door

**Toon VAN WATERSCHOOT**

© Katholieke Universiteit Leuven – Faculteit Ingenieurswetenschappen  
Arenbergkasteel, B-3001 Heverlee (Belgium)

Alle rechten voorbehouden. Niets uit deze uitgave mag vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotocopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaande schriftelijke toestemming van de uitgever.

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm or any other means without written permission from the publisher.

D/2009/7515/28

ISBN 978-94-6018-045-3



# Voorwoord

In de zomer van 2002 betrad ik voor het eerst sinds mijn promotie als ingenieursstudent een jaar voordien opnieuw de treden van het Departement Elektrotechniek. Op een jaar tijd was er in mijn leven veel veranderd. Ik was (in die volgorde) verhuisd naar Antwerpen om te gaan samenwonen met mijn vrouw Dorien, had mijn plannen om mijn ingenieursdiploma te verbranden en een conservatoriumopleiding te volgen opgeborgen, had mijn haren kort laten knippen en was vader geworden van onze oudste zoon Aeneas. Na een klein jaar lesgeven aan de Hogere Zeevaartschool in Antwerpen was ik op zoek naar een nieuwe broodwinning waarin ik -meer dan in de nochtans gezellige Zeevaartschool het geval was- mijn creatieve ei zou kwijtraken. Mijn oog was gevallen op een vacature aan de K.U.Leuven in het kader van een onderzoeksproject naar spraak- en audioverwerking, een domein waar ik altijd al een boon voor had. Niet toevallig stond het project onder de supervisie van Marc Moonen, die een jaar voordien ook het promotorschap van mijn eindwerk had waargenomen. Ik solliciteerde en kreeg vrijwel onmiddellijk een bericht van Marc met de vraag of we elkaar eens konden zien. Mijn verbazing was groot toen Marc me in het daaropvolgende gesprek vroeg “wanneer ik kon beginnen” en dat het misschien wel een goed idee zou zijn om *en passant* een doctoraat te maken. Ik had nochtans het idee dat ik tijdens mijn studentencarrière geen bijster goeie indruk had gemaakt, noch kon ik een jaar na mijn promotie een rijk gevuld CV voorleggen, laat staan dat het ooit mijn bedoeling zou geweest zijn om te doctoreren. Maar dat leken voor Marc geen argumenten te zijn om er niet voor te gaan: hij geloofde in mijn kwaliteiten en daardoor begon ik daar zelf ook in te geloven. Het vertrouwen dat ik sindsdien van Marc heb gekregen is iets waar ik hem altijd dankbaar voor zal zijn. Hij heeft me de tijd gegeven om mijn doctoraatsproject grondig aan te pakken, heeft me de juiste kansen op de juiste momenten aangeboden, en heeft altijd veel begrip gehad voor het feit dat ik niet alleen een doctoraatsstudent was maar ook een vader van drie jonge kinderen. Oprecht bedankt daarvoor, Marc, en ik hoop dat dit nog maar het begin is geweest van een vruchtbare samenwerking.

Naast Marc is er nog iemand die in grote mate heeft bijgedragen aan het wel-slagen van dit project, en dat is mijn goede vriend en *ancien* collega Geert

Rombouts. Het mag gezegd worden dat meer dan de helft van de publicaties in dit proefschrift de vruchten zijn van onze samenwerking. Marc, Geert en ik hebben in het eerste jaar van mijn doctoraatsproject vele uren gezamenlijk ons hoofd zitten breken over hoe we dat aartsmoeilijke feedbackprobleem konden aanpakken. Mocht Geert daar met zijn gezonde Kempense boerenverstand niet hebben bijgezet dan waren we er wellicht nooit uitgeraakt.

Verder wil ik oprecht de leden van mijn examencommissie bedanken voor hun deelname in de evaluatie van mijn werk. Het zijn stuk voor stuk onderzoekers waarvoor ik een grote waardering heb. Prof. Joos Vandewalle wil ik in de eerste plaats bedanken voor de prachtige onderzoeksomgeving die hij in de loop der jaren heeft uitgebouwd. Onderzoek zoals het mijne is enkel mogelijk in een groep als SISTA, waar onderzoekers uit verschillende disciplines elkaar ontmoeten en in discussie kunnen treden. Ik hoop dat ik hier nog heel wat jaren zal kunnen verderwerken. Overigens is het ook grotendeels zijn verdienste dat ik als “proefkonijn” een doctoraat op publicaties heb kunnen maken, waarvoor dank. Ook Prof. Hugo Van hamme wil ik van harte danken. Ondanks het feit dat het onderwerp van mijn doctoraat toch wat verder staat van zijn eigen onderzoek, is hij steeds met veel interesse en een grote kennis van zaken de discussie omtrent mijn werk aangegaan. Met Prof. Piet Sommen van de Technische Universiteit Eindhoven heb ik voornamelijk tijdens de eerste jaren van mijn doctoraatsproject zinvolle technische en minder technische discussies gehad, en aangezien hij een expert ter zake is ben ik erg blij dat hij deel wil uitmaken van mijn examencommissie. Hartelijk dank daarvoor. Also, it has been a pleasure to work together with Prof. Søren Jensen from Aalborg University during the past year, and I hope we can continue to do so. I'm very pleased to have him in my committee. Tot slot moet ik bekennen dat ik al een tijdje stiekem hoopte dat de faculteit Prof. Yves Willems als voorzitter van mijn commissie zou aanstellen, omdat ik me hem uit de lessen die ik destijds bij hem volgde herinner als een joviale professor en begeesterd lesgever. Van harte bedankt om het voorzitterschap van mijn commissie te willen waarnemen.

Ik herinner me ook nog dat ik na mijn eerste werkdagen als doctorandus mijn verbazing uitdrukte aan mijn vrouw Dorien over de bijzondere collega's in de onderzoeksgroep van Marc. Ik zal nooit vergeten hoe hartelijk ik destijds als “jongste telg” in de groep werd opgenomen. Nu, ruim zes jaar later, ben ik zelf bijna de oudste van de groep. Het kan verkeren. Hoe dan ook ben ik een grote dankuwel verschuldigd aan al die fijne collega's: Geert, Simon, Ann, Benoît, Gert, Koen, Geert, Hilde, Raphael, Imad, Geert, Olivier, Koen, Geert, T.J, Narendra, Deepak, Paschalis, Jan, Vincent, Prabin, Romain, Kim, Sylwester, Bram, Alexander, Pepe, Amir, and Beier. A special thank you to my Italian friend and colleague Ludo. En aan al wie ik vergeten ben.

Verder moet ik natuurlijk ook mijn broodheren bedanken. Gedurende zes en een half jaar hebben de K.U.Leuven en het Instituut voor de Aanmoediging van Innovatie door Wetenschap en Technologie in Vlaanderen (IWT) mij van

voldoende geld voorzien om mijn onderzoek in alle vrijheid te kunnen uitvoeren en daarnaast een vijfkoppig gezin te onderhouden. Ik kan niet zeggen dat we in al die jaren iets te kort gekomen zijn.

Ook buiten de universiteit bestaat er nog een universum waar het aange-naam toeven is. Mijn dolle vrienden van het theatergezelschap KREUTZFELD Verein, mijn collega-ego's van het theatercollectief G.A.A.S., de collega-ouders en de juffen van de Rudolf Steinerschool De Hazelaar, de peters en meters van onze kinderen, iedereen die mee geholpen heeft om ons huis om te toveren in een bouwwerf, ... al die mensen hebben mij kunnen entertainen, verstrooien, vervoeren en motiveren op de momenten dat het nodig was. Mijn oprechte dank daarvoor.

Er zijn negen mensen aan wie ik dit proefschrift in het bijzonder wil opdra-gen. Ze kennen elkaar niet, maar ze hebben allemaal iets gemeen: ze hebben mij op cruciale momenten in mijn leven begeleid, vanuit een onvoorwaardelijk vertrouwen, en op de één of andere manier beschouw ik hen als mijn belang-rijkste leermeesters. Frans Steurs, Dag Taeldeman, Kaat Van Bouwel, Stefaan Dieltjens, Rita De Clercq, Marc Moonen, Liliane Bierinckx, Dirk Beckers, Serge Ornelis. Dit boek is voor jullie.

Tenslotte komen de meest dierbaren aan bod. Mijn broer Jef en ik hebben elkaar altijd op de voet gevolgd: samen gespeeld, samen in Leuven gestudeerd, en op hetzelfde moment kinderen gekregen. Ik hoop van harte dat hij ook nog een doctoraat zal maken... Voor mijn ouders is het einde van dit project wellicht ook een keerpunt. Ze hebben altijd van dichtbij meegeleefd, en me hun onvoorwaardelijke steun en bewondering laten blijken. Dankjewel daarvoor. Ook mijn schoonouders verdienen een bijzonder woord van dank. Samen met mijn ouders hebben ze ons gezin immers altijd ondersteund wanneer dat nodig was, zowel emotioneel als logistiek.

Aeneas, jij bent nu ongeveer even oud als dit boek, en het lijkt alsof je al veel meer weet dan wat hier in staat. Firas, hoewel jij nu de grootste avonturier bent die ik ken, heb je uren, dagen en weken bij mij in de draagdoek gezeten terwijl ik mijn eerste artikel zat te schrijven. Istar, jij bent de beste speelkameraad die een thuiswerkende papa zich kan inbeelden. Ik hoop dat jullie vooral verder doen zoals jullie bezig zijn, het is een plezier om jullie vader te zijn.

En Dorien, zonder jou zou ik hier nooit aan begonnen zijn, laat staan dat ik het ooit zou hebben afgewerkt...

Toon van Waterschoot,  
17 maart 2009



# Abstract

This thesis deals with several open problems in acoustic echo cancellation and acoustic feedback control. Our main goal has been to develop solutions that provide a high performance and sound quality, and behave in a robust way in realistic conditions. This can be achieved by departing from the traditional ad-hoc methods, and instead deriving theoretically well-founded solutions, based on results from parameter estimation and system identification. In the development of these solutions, the computational efficiency has permanently been taken into account as a design constraint, in that the complexity increase compared to the state-of-the-art solutions should not exceed 50 % of the original complexity.

In the context of acoustic echo cancellation, we have investigated the problems of double-talk robustness, acoustic echo path undermodeling, and poor excitation. The two former problems have been tackled by including adaptive decorrelation filters in the adaptive filtering algorithm, with the aim of whitening the near-end signal component and the residual echo component resulting from undermodeling. These decorrelation filters can be identified concurrently with the acoustic echo path by using the prediction error method (PEM) for system identification. As a result, a 30–40 dB misadjustment improvement (in the double-talk case) and a 20–35 dB variance decrease (in the undermodeling case) have been obtained, at the cost of a complexity increase of 50 % compared to the normalized least mean squares (NLMS) algorithm. The poor excitation problem has been approached from a Bayesian minimum mean square error (MMSE) point of view. This approach has led to the use of a regularization matrix different from the traditional scaled identity matrix, which may incorporate prior knowledge on the acoustic echo path. It has moreover been shown that the existing proportionate adaptation algorithms can be viewed as a special case of the proposed approach to regularization. A misadjustment improvement up to 10 dB has been obtained with a regularized NLMS-type algorithm that requires only 25 % more computations than the original NLMS algorithm.

Two approaches to acoustic feedback control have been considered in this thesis, namely notch-filter-based howling suppression (NHS) and adaptive feedback cancellation (AFC). In the context of NHS, we have developed a novel parametric frequency estimation method, which is characterized by a computational complexity that is linear in the data record length. Also, a new design procedure for biquadratic parametric equalizer filters is proposed, based on a technique known as pole-zero placement. In the context of AFC, the PEM-based AFC approach that was proposed earlier for hearing aid AFC has been generalized to room acoustic and audio applications. The PEM-based approach relies on the identification of a near-end signal model that can be used in the design of decorrelating prefilters. These prefilters are aimed at resolving the AFC closed-loop signal correlation problem and hence providing an unbiased acoustic feedback path model. We have obtained a misadjustment improvement of 7 dB compared to the hearing aid PEM-based AFC algorithm and of 12 dB compared to the NLMS algorithm, at the cost of a 25–50 % complexity increase compared to NLMS. In a comparative evaluation with the state-of-the-art acoustic feedback control methods, the PEM-based AFC approach was shown to outperform the existing phase-modulating feedback control (PFC) and NHS methods, as well as the AFC methods that apply a decorrelation in the closed signal loop, in terms of the achievable maximum stable gain and sound quality, both for speech and audio signals.

# Korte Inhoud

In dit doctoraatsproefschrift worden verscheidene open problemen in akoestische-echo-onderdrukking en akoestische-feedbackbeheersing behandeld. De algemene doelstelling bestaat in het ontwikkelen van oplossingen die een hoge performantie en geluidskwaliteit leveren en die een robuust gedrag vertonen in een realistische omgeving. De voorgestelde aanpak steunt op een solide theoretische basis door de toepassing van resultaten uit de systeemidentificatie en parameterschatting, in tegenstelling tot de ad-hoc aanpak die gevolgd wordt in bestaande oplossingen. De voorgestelde oplossingen werden ontwikkeld met permanente aandacht voor de vereiste rekenkracht, zodat die maximaal 50 % hoger ligt dan de nodige rekenkracht voor de bestaande oplossingen.

In het kader van akoestische-echo-onderdrukking worden de problemen van dubbelspraakrobuustheid, ondermodellering van het akoestische-echopad en onvoldoende excitatie behandeld. De aanpak van de twee eerste problemen is gebaseerd op het gebruik van adaptieve decorrelerende prefilters die de spectrale kleuring in het bronsignaal en in de residuele echocomponent ten gevolge van ondermodellering moeten reduceren. Die decorrelerende filters kunnen gelijktijdig worden geïdentificeerd met het akoestische-echopad door gebruik te maken van de predictiefoutmethode (PEM) voor systeemidentificatie. Het resultaat is een verbetering van de misaanpassing met 30-40 dB (in het geval van dubbelspraak) en een verlaging van de variantie met 20-35 dB (in het geval van ondermodellering), ten koste van een toename in de rekencomplexiteit met 50 % in vergelijking met het genormaliseerde kleinste-gemiddelde-kwadratenalgoritme (NLMS). Het probleem van onvoldoende excitatie wordt benaderd vanuit een Bayesiaans minimale-gemiddelde-kwadratische-fout-perspectief (MMSE). Die aanpak leidt tot het gebruik van een regularisatiematrix die verschilt van de traditionele gescaalde eenheidsmatrix, waarin eventuele a-priorikennis over het akoestische-echopad kan worden opgenomen. Bovendien wordt aangetoond dat de bestaande proportionele-adaptatie-algoritmen genterpreteerd kunnen worden als een speciaal geval van de voorgestelde regularisatie-aanpak. Een verbetering van de misaanpassing tot 10 dB blijkt mogelijk te zijn met een geregulariseerd NLMS-gebaseerd algoritme dat slechts 25 % meer rekenkracht vraagt dan het originele NLMS-algoritme.

Het akoestische-feedbackprobleem wordt in dit proefschrift op twee manieren benaderd, meer bepaald op basis van inkepingsfiltergebaseerde fluittoononderdrukking (NHS) en adaptieve feedbackonderdrukking (AFC). In het kader van NHS wordt een nieuwe parametrische frequentieschattingsmethode ontwikkeld die gekenmerkt wordt door een rekencomplexiteit die lineair toeneemt met de lengte van het gebruikte databestand. Daarnaast wordt een nieuwe ontwerp-procedure voor bikwadratische parametrische-egalisatiefilters voorgesteld die gebaseerd is op de techniek van pool- en nulplaatsing. In het kader van AFC wordt de PEM-gebaseerde AFC-aanpak die eerder werd ontwikkeld voor gebruik in hoorapparaten uitgebreid naar kamerakoestische en audiotoeepassingen. De PEM-gebaseerde aanpak steunt op de identificatie van een bronsignaalmodel dat kan worden gebruikt bij het ontwerp van decorrelerende prefilters. Het doel van die prefilters bestaat erin het geslotenlussignaalcorrelatieprobleem eigen aan AFC op te lossen om een foutloos akoestische-feedbackpadmodel te kunnen bekomen. Het blijkt mogelijk om op die manier de misaanpassing te verbeteren met 7 dB in vergelijking met het PEM-gebaseerde AFC-algoritme voor hoorapparaten en met 12 dB vergeleken met het NLMS-algoritme, en dit ten koste van een verhoging van de rekencomplexiteit met 25-50 % t.o.v. NLMS. Uit een vergelijkende evaluatie met de state-of-the-artmethodes voor akoestische-feedbackbeheersing blijkt de PEM-gebaseerde AFC-aanpak zowel de bestaande fasemodulatiegebaseerde feedbackbeheersingsmethodes (PFC) en NHS-methodes, als de AFC-methodes die een decorrelatie in de gesloten signaallus uitvoeren, te overtreffen in termen van de haalbare maximale stabiele versterking en geluidskwaliteit, en dit voor spraak- én muzieksignalen.



# Glossary

## Mathematical Notation

### Independent variables

$f$	frequency variable (Hz)
$\tilde{f}$	warped frequency variable (Hz)
$i, j$	frame index [Ch. 9]
$k$	discrete time shift index discrete frequency index [Ch. 11]
$n$	LPTV filter frequency response sideband index
$t$	discrete time variable
$z$	complex $z$ -transform variable
$\tilde{z}$	warped/downsampled complex $z$ -transform variable
$\tau$	continuous time variable (s) [Ch. 2] discrete time shift index [Ch. 2]
$\omega$	radial frequency variable (rad)
$\tilde{\omega}$	warped radial frequency variable (rad)
$\omega_k$	discrete radial frequency variable (rad)

### Signals

$d(t), d[t, \cdot]$	echo-/feedback-compensated signal, a posteriori residual
$D(\omega)$	echo-/feedback-compensated signal frequency spectrum (time-invariant data window)
$D[\omega, t, \cdot]$	echo-/feedback-compensated signal frequency spectrum (time-varying data window)
$e(t)$	near-end/source excitation signal [Ch. 3]
$e(t), e(t, \cdot)$	LP/CPZLP residual signal [Ch. 8, 11]
$E(z, \cdot)$	LP residual signal $z$ -transform (time-invariant data window)
$E(e^{j\omega}, \cdot)$	LP residual signal spectrum (time-invariant data window)

<b>e</b>	near-end/source excitation signal vector (time-invariant data window)
$e_n(t, \cdot)$	CPZLP intermediate residual signal ( $n$ th subproblem)
$\mathbf{e}_n(\cdot)$	CPZLP intermediate residual signal vector ( $n$ th subproblem)
$e'(t)$	far-end/loudspeaker excitation signal
$n(t)$	noise signal injected for AFC decorrelation [Ch. 2]
$n_b(t)$	undermodeled AEC disturbance signal [Ch. 4]
$n_b(t)$	undermodeled AEC disturbance signal component in far-end/loudspeaker signal column space
$n_v(t)$	undermodeled AEC disturbance signal component orthogonal to far-end/loudspeaker signal column space
$\mathbf{n}(t)$	undermodeled AEC disturbance signal vector
$\mathbf{n}_b(t)$	undermodeled AEC disturbance signal vector component in far-end/loudspeaker signal column space
$\mathbf{n}_v(t)$	undermodeled AEC disturbance signal vector component orthogonal to far-end/loudspeaker signal column space
$r(t)$	near-end/source excitation signal [Ch. 5]
$R(e^{j\omega})$	noise signal in sinusoidal/tonal signal model [Ch. 8, 11]
$r[t, \cdot]$	residual echo/feedback signal [Ch. 2]
	echo-/feedback-compensated signal after tonal components model prediction error filtering [Ch. 12]
$\mathbf{r}[t, \cdot]$	echo-/feedback-compensated signal vector after tonal components model prediction error filtering [Ch. 12]
$\hat{R}[\omega, t, \cdot]$	residual echo/feedback signal frequency spectrum estimate (time-varying data window)
$s(t)$	far-end echo signal
$u(t)$	far-end/loudspeaker signal
$U(\omega, t)$	far-end/loudspeaker signal frequency spectrum (time-varying data window)
$\mathbf{u}(t)$	far-end/loudspeaker signal vector (time-varying data window)
$\mathbf{u}_1(t)$	far-end/loudspeaker signal vector related to modeled part of echo/feedback path impulse response (time-varying data window)
$\mathbf{u}_2(t)$	far-end/loudspeaker signal vector related to unmodeled part of echo/feedback path impulse response (time-varying data window)

$\mathbf{U}(t)$	far-end/loudspeaker signal Hankel matrix (time-varying data window)
$\mathbf{U}_1(t)$	far-end/loudspeaker signal Hankel matrix related to modeled part of echo/feedback path impulse response (time-varying data window)
$\mathbf{U}_2(t)$	far-end/loudspeaker signal Hankel matrix related to unmodeled part of echo/feedback path impulse response (time-varying data window)
$\mathbf{U}_M(t)$	far-end/loudspeaker signal Hankel matrix ( $M$ th order affine projection)
$\mathbf{U}$	far-end/loudspeaker signal Hankel matrix (time-invariant data window)
$\tilde{u}(t), \tilde{u}[t, \cdot]$	prefiltered far-end/loudspeaker signal
$\tilde{\mathbf{u}}(t), \tilde{\mathbf{u}}[t, \cdot]$	prefiltered far-end/loudspeaker signal vector (time-varying data window)
$\tilde{\mathbf{U}}_M(t)$	prefiltered far-end/loudspeaker signal Hankel matrix ( $M$ th order affine projection)
$\tilde{\mathbf{U}}$	prefiltered far-end/loudspeaker signal Hankel matrix (time-invariant data window)
$\check{u}(t)$	far-end/loudspeaker signal after noise components model prediction error filtering
$\check{\mathbf{u}}(t)$	far-end/loudspeaker signal vector after noise components model prediction error filtering
$\check{u}[t, \cdot]$	far-end/loudspeaker signal after tonal components model prediction error filtering
$\bar{u}(t, 0), \dots, \bar{u}(t, n_\alpha)$	two-dimensional warped far-end/loudspeaker signal
$u_1(t), \dots, u_L(t)$	multi-channel far-end/loudspeaker signals
$\bar{\mathbf{u}}(t)$	multi-channel far-end/loudspeaker signal vector
$v(t)$	near-end/source signal
$V(\omega, t)$	near-end/source signal frequency spectrum (time-varying data window)
$\mathbf{v}$	near-end/source signal vector (time-invariant data window)
$\mathbf{v}(t)$	near-end/source signal vector (time-varying data window)
$v_1(t), \dots, v_S(t)$	multi-channel near-end/source signals
$\bar{\mathbf{v}}(t)$	multi-channel near-end/source signal vector
$w(t)$	local far-end signal [Ch. 1]
$w(t), w[t, \cdot]$	echo-/feedback-compensated signal after noise components model prediction error filtering [Ch. 9, 12]

$\mathbf{w}(t), \mathbf{w}[t, \cdot]$	echo-/feedback-compensated signal vector after noise components model prediction error filtering
$x(t)$	echo/feedback signal
$y(t)$	microphone signal
$Y(z)$	observed signal [Ch. 8, 11]
$Y(\omega)$	microphone signal $z$ -transform (time-invariant data window)
$Y(e^{j\omega})$	observed signal frequency spectrum [Ch. 11]
$Y(\omega, t)$	microphone signal frequency spectrum (time-varying data window)
$\mathbf{y}(t)$	microphone signal vector (time-varying data window)
$\mathbf{y}_M(t)$	microphone signal vector ( $M$ th order affine projection)
$\mathbf{y}$	microphone signal vector (time-invariant data window)
$\mathbf{Y}(t)$	microphone signal discrete frequency spectrum vector (time-varying data window)
$\tilde{y}(t), \tilde{y}[t, \cdot]$	prefiltered microphone signal
$\tilde{\mathbf{y}}_M(t)$	prefiltered microphone signal vector ( $M$ th order affine projection)
$\tilde{\mathbf{y}}$	prefiltered microphone signal vector (time-invariant data window)
$\check{y}(t)$	microphone signal after noise components model prediction error filtering
$\check{y}[t, \cdot]$	microphone signal after tonal components model prediction error filtering
$\bar{y}(t, 0), \dots, \bar{y}(t, n_\alpha)$	two-dimensional warped microphone signal
$y_1(t), \dots, y_S(t)$	multi-channel microphone signals
$\bar{\mathbf{y}}(t)$	multi-channel microphone signal vector
$y\{t\}$	microphone signal (continuous time)
$y_a\{t\}$	analytical microphone signal (continuous time)
$\hat{y}\{t\}$	microphone signal Hilbert transform (continuous time)
$\hat{y}(t)$	microphone signal Hilbert transform [Ch. 2]
$\hat{y}(t), \hat{y}[t, \cdot]$	predicted echo/feedback signal
$\hat{Y}[\omega, t, \cdot]$	predicted echo/feedback signal frequency spectrum (time-varying data window)
$z(t)$	far-end microphone signal
$\varepsilon(t), \varepsilon[t, \cdot]$	prediction error, a priori residual
$\varepsilon_B(t)$	background filter prediction error
$\varepsilon_M(t), \varepsilon_M[t, \cdot]$	a priori residual vector ( $M$ th order affine projection)

$\varepsilon$	prediction error vector (time-invariant data window)
$\tilde{\varepsilon}(t)$	prefiltered a priori residual
$\tilde{\varepsilon}_M(t)$	prefiltered a priori residual vector ( $M$ th order affine projection)

## System models

$A(q, t)$	near-end/source signal model prediction error filter (time-varying)
	near-end/source signal tonal components model prediction error filter (time-varying) [Ch. 2, 12]
	(near-end/source signal tonal components) PZLP model prediction error filter numerator (time-varying) [Ch. 2, 11, 12]
$a_1, a_2$	biquadratic parametric equalizer direct-form denominator coefficients [Ch. 7]
$a_0, \dots, a_{2P}$	CPZLP model direct-form coefficients [Ch. 8]
	LP model direct-form denominator coefficients [Ch. 11]
$a_1(t), \dots, a_{n_A}(t)$	near-end/source signal model prediction error filter impulse response coefficients (time-varying)
$a_l^{(1)}(t), a_l^{(2)}(t)$	$l$ th second-order section denominator coefficients in bank of adjustable notch filters (time-varying)
$\mathbf{a}$	LP model prediction error filter impulse response vector (time-invariant) [Ch. 11]
$\mathbf{a}(t)$	near-end/source signal model prediction error filter impulse response vector (time-varying)
	near-end/source signal tonal components model prediction error filter impulse response vector (time-varying) [Ch. 12]
$A(z)$	LP model denominator transfer function (time-invariant)
$A(e^{j\omega})$	LP model denominator frequency response (time-invariant)
$A_l(e^{j\omega})$	LP model $l$ th second order section denominator frequency response (time-invariant)
$a'_1(t), \dots, a'_{n_A}(t)$	far-end/loudspeaker signal model prediction error filter impulse response coefficients (time-varying)
$\mathbf{a}'(t)$	far-end/loudspeaker signal model prediction error filter impulse response vector (time-varying)
$\hat{A}(q, t)$	near-end/source signal model prediction error filter estimate (time-varying)

	near-end/source signal tonal components model prediction error filter estimate (time-varying) [Ch. 12]
$\hat{\mathbf{a}}_1(t), \dots, \hat{\mathbf{a}}_{n_A}(t)$	near-end/source signal model prediction error filter estimate impulse response coefficients (time-varying)
$\hat{a}_i^{(1)}, \dots, \hat{a}_i^{(n_A)}$	near-end/source signal model prediction error filter estimate impulse response coefficients (piecewise time-invariant, frame index $i$ ) [Ch. 9]
$\hat{\mathbf{a}}(t)$	near-end/source signal model prediction error filter estimate impulse response vector (time-varying) combined near-end/source signal and far-end/loudspeaker signal model prediction error filter estimate impulse response vector (time-varying) [Ch. 4]
	near-end/source signal tonal components model prediction error filter estimate impulse response vector (time-varying) [Ch. 12]
$\hat{\mathbf{a}}_1(t), \dots, \hat{\mathbf{a}}_M(t)$	subband near-end/source signal model prediction error filter estimate impulse response vectors (time-varying, $M$ subbands)
$\hat{\mathbf{a}}_i$	near-end/source signal model prediction error filter estimate impulse response vector (piecewise time-invariant, frame index $i$ )
$B(q, t)$	(near-end/source signal tonal components) PZLP model prediction error filter denominator (time-varying) [Ch. 2, 11, 12]
	cascade model of echo/feedback path and near-end/source signal model prediction error filter (two-channel adaptive filtering) [Ch. 3, 9]
$B_0(q, t)$	true near-end/source signal tonal components PZLP model prediction error filter denominator (time-varying)
$b_0, b_1, b_2$	biquadratic parametric equalizer direct-form numerator coefficients [Ch. 7]
$b_0, \dots, b_{2Q}$	LP model direct-form numerator coefficients [Ch. 11]
$b(t)$	near-end/source excitation signal amplitude in near-end/source signal model [Ch. 9]
$b_0(t), \dots, b_{n_B}(t)$	cascade model impulse response coefficients of echo/feedback path and near-end/source signal model prediction error filter (two-channel adaptive filtering) [Ch. 3]

$b_l^{(0)}(t), b_l^{(1)}(t), b_l^{(2)}(t)$	$l$ th second-order section numerator coefficients in bank of adjustable notch filters (time-varying)
$\hat{\mathbf{b}}(t)$	cascade model estimate impulse response vector of echo/feedback path and near-end/source signal model prediction error filter (two-channel adaptive filtering)
$B(z)$	LP model numerator transfer function (time-invariant)
$B(e^{j\omega})$	LP model numerator frequency response (time-invariant)
$B_l(e^{j\omega})$	LP model $l$ th second order section numerator frequency response (time-invariant)
$\hat{B}(q, t)$	cascade model estimate of echo/feedback path and near-end/source signal model prediction error filter (two-channel adaptive filtering)
$\bar{B}(q, t)$	delay-compensated near-end/source signal tonal components PZLP model prediction error filter denominator (time-varying)
$C(q, t)$	near-end/source signal noise components model prediction error filter (time-varying)
$c^{(1)}(t), \dots, c^{(n_c)}(t),$ $c_1(t), \dots, c_{n_c}(t)$	near-end/source signal noise components model prediction error filter impulse response coefficients (time-varying)
$\mathbf{c}(t)$	near-end/source signal noise components model prediction error filter impulse response vector (time-varying)
$\hat{C}(q, t)$	near-end/source signal noise components model prediction error filter estimate (time-varying)
$\hat{\mathbf{c}}(t)$	near-end/source signal noise components model prediction error filter estimate impulse response vector (time-varying)
$D(q, t)$	cascade of near-end/source signal noise components model PEF and tonal components model PEF numerator (time-varying)
$D_0(q, t)$	true cascade of near-end/source signal noise components model PEF and tonal components model PEF numerator (time-varying)
$D(q, \lambda)$	WLP model bilinear all-pass filter (time-invariant)
$D_0(q, \lambda)$	WLP model orthogonalizing lowpass filter (time-invariant)
$D_0(z)$	WLP model orthogonalizing lowpass filter transfer function (time-invariant)
$F(q)$	echo/feedback path model (time-invariant)
$\hat{F}(q, t)$	echo/feedback path model (time-varying)

$f_0, \dots, f_{n_F}$	echo/feedback path impulse response coefficients (time-invariant)
$f_0(t), \dots, f_{n_F}(t)$	echo/feedback path impulse response coefficients (time-varying)
$\mathbf{f}$	echo/feedback path impulse response vector (time-invariant)
$\mathbf{f}(t)$	echo/feedback path impulse response vector (time-varying)
$\mathbf{f}_1(t)$	modeled part of echo/feedback path impulse response vector (time-varying)
$\mathbf{f}_2(t)$	unmodeled part of echo/feedback path impulse response vector (time-varying)
$F(z)$	echo/feedback path transfer function (time-invariant)
$F(\omega)$	echo/feedback path frequency response (time-invariant)
$F(\omega, t), F(e^{j\omega}, t)$	echo/feedback path frequency response (time-varying)
$F_{ij}(q, t)$	multi-channel echo/feedback path model for loudspeaker-microphone pair $(j, i)$ (time-varying)
$f_{ij}^{(0)}(t), \dots, f_{ij}^{(n_F)}(t)$	multi-channel echo/feedback path impulse response coefficients for loudspeaker-microphone pair $(j, i)$ (time-varying)
$\mathbf{F}(q, t)$	multi-channel echo/feedback path model polynomial matrix (time-varying)
$\hat{F}(q)$	echo/feedback path estimate (time-invariant)
$\hat{F}(q, t)$	echo/feedback path estimate (time-varying)
$\hat{F}_0(q, t)$	AFC cancellation filter (time-varying)
$\hat{f}_0, \dots, \hat{f}_{n_{\hat{F}}}$	echo/feedback path estimate impulse response coefficients (time-invariant)
$\hat{f}_0(t), \dots, \hat{f}_{n_{\hat{F}}}(t)$	echo/feedback path estimate impulse response coefficients (time-varying)
$\hat{\mathbf{f}}$	echo/feedback path estimate impulse response vector (time-invariant)
$\hat{\mathbf{f}}(t)$	echo/feedback path estimate impulse response vector (time-varying)
$\hat{\mathbf{f}}_0(t)$	AFC cancellation filter impulse response vector (time-varying)
$\hat{\mathbf{f}}_1(t)$	undermodeled echo/feedback path estimate impulse response vector (time-varying) [Ch. 4]
$\hat{\mathbf{f}}_1(t), \dots, \hat{\mathbf{f}}_M(t)$	subband echo/feedback path estimate impulse response vectors (time-varying, $M$ subbands) [Ch. 10]



$\hat{\mathbf{f}}_B(t)$	echo/feedback path background filter estimate impulse response vector (time-varying)
$\hat{F}(z)$	echo/feedback path estimate transfer function
$\hat{F}(\omega)$	echo/feedback path estimate frequency response (time-invariant)
$\hat{F}(\omega, t), \hat{F}(e^{j\omega}, t)$	echo/feedback path estimate frequency response (time-varying)
$\hat{F}_{ij}(q, t)$	multi-channel echo/feedback path estimate for loudspeaker-microphone pair $(j, i)$ (time-varying)
$\hat{f}_{ij}^{(0)}(t), \dots, \hat{f}_{ij}^{(n_{\hat{F}})}(t)$	multi-channel echo/feedback path estimate impulse response coefficients for loudspeaker-microphone pair $(j, i)$ (time-varying)
$\bar{F}(q, t)$	delay-compensated echo/feedback path model (time-varying)
$\bar{f}_0, \dots, \bar{f}_{n_{\bar{F}}}$	initial echo/feedback path estimate impulse response coefficients (time-invariant)
$\bar{\mathbf{f}}$	initial echo/feedback path estimate impulse response vector (time-invariant)
$\hat{\bar{F}}(q, t)$	delay-compensated echo/feedback path estimate (time-varying)
$G\{\cdot\}$	electro-acoustic forward path operator (time-invariant)
$G(q)$	electro-acoustic forward path model (time-invariant)
$G(q, t)$	far-end echo path model (time-invariant) [Ch. 1] electro-acoustic forward path model (time-varying) far-end echo path model (time-varying) [Ch. 1]
$g_0(t), \dots, g_{n_G}(t)$	electro-acoustic forward path impulse response coefficients (time-varying)
$G(z)$	LP model transfer function (time-invariant)
$G(\omega)$	electro-acoustic forward path frequency response (time-invariant)
$G(\omega, t), G(e^{j\omega}, t)$	electro-acoustic forward path frequency response (time-varying)
$G_{ji}[\cdot, t]$	multi-channel electro-acoustic forward path operator for microphone-loudspeaker pair $(i, j)$ (time-varying)
$G_{ji}(q, t)$	multi-channel electro-acoustic forward path model for microphone-loudspeaker pair $(i, j)$ (time-varying)
$g_{ji}^{(0)}(t), \dots, g_{ji}^{(n_G)}(t)$	multi-channel electro-acoustic forward path impulse response coefficients for microphone-loudspeaker pair $(i, j)$ (time-varying)

$\mathbf{G}[\cdot, t]$	multi-channel electro-acoustic forward path matrix operator (time-varying)
$\mathbf{G}(q, t)$	multi-channel electro-acoustic forward path model polynomial matrix (time-varying)
$G_l(\omega)$	closed-loop system loop gain
$\hat{G}(q, t)$	far-end echo path estimate (time-varying)
$\tilde{G}(q, t)$	delay-compensated electro-acoustic forward path model (time-varying)
$H(q, t)$	bank of adjustable notch filters (time-varying) [Ch. 2] phase modulation filter (time-varying) [Ch. 2] near-end/source signal model (time-varying) [Ch. 2, 9, 10]
$H_1(q, t)$	near-end/source signal tonal components model (time-varying) [Ch. 12]
$H_2(q, t)$	near-end/source signal noise components model (time-varying) [Ch. 12]
$H_l(q, t)$	$l$ th second-order section in bank of adjustable notch filters (time-varying) [Ch. 2]
$h(\tau, t)$	phase modulation filter impulse response (time-varying)
$H(z)$	biquadratic parametric equalizer filter transfer function (time-invariant) [Ch. 7] LP model prediction error filter transfer function (time-invariant) [Ch. 11]
$H(e^{j\omega})$	biquadratic parametric equalizer filter frequency response (time-invariant) [Ch. 7] LP model prediction error filter frequency response (time-invariant) [Ch. 11]
$H(\omega, t)$	phase modulation filter frequency response (time-varying) [Ch. 2] AFC postfilter frequency response (time-varying) [Ch. 2]
$\tilde{H}(\omega, t)$	AFC postfilter frequency response before smoothing (time-varying)
$\mathcal{H}(n)$	phase modulation filter frequency response DFT (time-varying, frequency-independent)
$\mathcal{H}(\omega, n)$	phase modulation filter frequency response DFT (time-varying, frequency-dependent)
$H(\cdot, t)$	nonlinear signal operation for AFC decorrelation
$\hat{H}(q, t)$	near-end/source signal model estimate (time-varying)
$\hat{H}_1(q, t)$	near-end/source signal tonal components model estimate (time-varying)

$\hat{H}_2(q, t)$	near-end/source signal noise components model estimate (time-varying)
$\hat{\mathbf{h}}(t)$	near-end/source signal model parameter vector estimate (time-varying)
$J(q, t)$	electro-acoustic forward path model before amplification (time-varying)
$J(e^{j\omega}, t)$	electro-acoustic forward path frequency response before amplification (time-varying)
$K(t)$	electro-acoustic forward path gain (time-varying)
$L(q)$	Hilbert filter
$L(q, t)$	cascade of near-end/source signal noise components model PEF and tonal components model PEF numerator and echo/feedback path model (time-varying)
$\bar{L}(q, t)$	delay-compensated cascade of near-end/source signal noise components model PEF and tonal components model PEF numerator and echo/feedback path model (time-varying)
$\alpha^{(-1)}(t), \alpha^{(0)}(t), \alpha^{(1)}(t)$	near-end/source signal tonal components 3-tap PLP model prediction error filter coefficients (time-varying)
$\alpha_{-1}(t), \alpha_0(t), \alpha_1(t)$	near-end/source signal tonal components 1-tap PLP model prediction error filter coefficient (piecewise time-invariant, frame index $j$ ) [Ch. 9]
$\alpha_j$	near-end/source signal tonal components 1-tap PLP model prediction error filter coefficient (piecewise time-invariant, frame index $j$ ) [Ch. 9]
$\alpha_1(t), \dots, \alpha_{n_A}(t)$	near-end/source signal tonal components WLP/SLP model prediction error filter coefficients (time-varying) [Ch. 12]
$\varphi_l(\omega)$	closed-loop system loop phase

### Auxiliary variables and signal statistics

$\mathbf{a}_0$	true near-end/source signal model prediction error filter impulse response expected value
$A_n$	amplitude of $n$ th sinusoidal component in sinusoidal signal model
$\mathbf{A}$	unit upper triangular matrix in inverse near-end/source signal covariance matrix Cholesky decomposition [Ch. 5]
$\mathbf{A}$	unit upper triangular band matrix in inverse autoregressive near-end/source signal covariance matrix Cholesky decomposition [Ch. 5, 9]
$\hat{\mathbf{A}}$	estimated unit upper triangular band matrix in inverse autoregressive near-end/source signal covariance matrix Cholesky decomposition

$\mathbf{A}_M(t)$	unit upper triangular submatrix in inverse near-end/source signal covariance matrix Cholesky decomposition ( $M$ th order affine projection)
$B_k$	CPZLP BFGS Hessian approximation ( $k$ th iteration)
$\mathcal{C}_{H_l}(t)$	set of $l$ th second-order section filter coefficients in bank of adjustable notch filters
$\mathcal{D}_H(t)$	set of NHS design parameters
$\mathcal{D}_{\tilde{\omega}}(t)$	set of candidate howling component frequencies
$f$	fractional pitch lag phase [Ch. 11]
$f_0$	fundamental frequency in monophonic signal model (Hz)
$f_{0,n}$	fundamental frequency of $n$ th monophonic component in polyphonic signal model (Hz)
$f_c$	biquadratic parametric equalizer filter center frequency (Hz)
$\mathbf{f}_0$	true echo/feedback path impulse response expected value
$g_1, \dots, g_P$	PZLP model common numerator and denominator coefficient factors
$g_0(t), \dots, g_{n_F}(t)$	proportionate adaptation weights [Ch. 5]
$\bar{g}(t)$	average proportionate adaptation weight
$\mathbf{G}(t)$	proportionate adaptation matrix
$I(q, \cdot)$	linear interpolation filter
$\mathcal{I}_H(t)$	set of candidate howling component indices for which howling is detected
$K$	integer pitch lag [Ch. 2, 11, 12] biquadratic parametric equalizer filter broadband gain factor [Ch. 7]
$K_j$	integer pitch lag (piecewise time-invariant, frame index $j$ )
$l$	fractional pitch lag phase [Ch. 2, 12]
$p^{(k)}$	CPZLP search direction ( $k$ th iteration)
$\mathcal{P}$	set of frequencies at which Nyquist phase condition holds
$\hat{P}_y(t)$	DFT-based microphone signal power estimate (time-varying)
$\mathbf{Q}_1$	band-diagonal autocorrelation matrix of far-end/loudspeaker signal vector related to modeled part of echo/feedback path impulse response
$\mathbf{Q}_2$	lower triangular cross-correlation matrix of far-end/loudspeaker signal vectors related to modeled and unmodeled part of echo/feedback path impulse response

$r_a(i)$	LP model direct-form denominator coefficients autocorrelation function
$r_b(i)$	LP model direct-form numerator coefficients autocorrelation function
$r_p$	biquadratic parametric equalizer filter pole radius
$r_z$	biquadratic parametric equalizer filter zero radius
$\mathbf{R}$	near-end/source signal covariance matrix
$\mathbf{R}_{11}$	autocorrelation matrix of far-end/loudspeaker signal vector related to modeled part of echo/feedback path impulse response
$\mathbf{R}_{12}$	cross-correlation matrix of far-end/loudspeaker signal vectors related to modeled and unmodeled part of echo/feedback path impulse response
$\mathbf{R}_a$	true near-end/source signal model prediction error filter impulse response covariance matrix
$\hat{\mathbf{R}}_{a,s}$	true near-end/source signal model prediction error filter impulse response covariance matrix estimate (based on ensemble-averaging of speech models)
$\mathbf{R}_f$	true echo/feedback path impulse response covariance matrix
$\hat{\mathbf{R}}_{f,\text{init}}$	true echo/feedback path impulse response covariance matrix estimate (based on initial echo/feedback path measurement)
$\hat{\mathbf{R}}_{f,3}$	true echo/feedback path impulse response covariance matrix estimate (based on 3-parameter echo/feedback path model)
$\mathbf{R}_v$	near-end/source signal covariance matrix
$\mathbf{R}(t)$	Hessian approximation (recursive identification)
$\mathbf{R}_A(t)$	Hessian approximation for near-end/source signal model identification (recursive identification)
$\mathbf{R}_F(t)$	Hessian approximation for echo/feedback path identification (recursive identification)
$s_k$	CPZLP BFGS displacement variable ( $k$ th iteration)
$S_d(f, t), S_d[\omega, t, \cdot]$	echo-/feedback-compensated signal power spectral density (time-varying)
$S_r[\omega, t, \cdot]$	residual echo/feedback signal power spectral density (time-varying)
$\hat{S}_r[\omega, t, \cdot]$	residual echo/feedback signal power spectral density estimate (time-varying)

$S_v(f, t)$	near-end/source signal power spectral density (time-varying)
$\mathcal{S}_{H_l}(t)$	set of $l$ th second-order section filter specifications in bank of adjustable notch filters
$t_0$	speech onset detection time index
$T_0$	pitch period in monophonic signal model (s)
$T_{0,n}$	pitch period of $n$ th monophonic component in polyphonic signal model (s)
$\Delta t_i$	howling occurrence time duration (s)
$u_k$	CPZLP BFGS damped change of gradients variable ( $k$ th iteration)
$v_k$	CPZLP BFGS change of gradients variable ( $k$ th iteration)
$w(t)$	window function
$w_{\text{ERB}}(f)$	perceptual frequency weighting function
$w_h(t)$	Hamming window function
$\mathbf{W}$	weighting matrix (echo/feedback path impulse response estimation)
$\alpha_n$	amplitude of $n$ th sinusoidal component in tonal signal model
$\alpha_{n,m}$	amplitude of $m$ th sinusoidal component in $n$ th monophonic component in polyphonic signal model
$\boldsymbol{\alpha}(t)$	near-end/source signal tonal components model parameter vector (time-varying)
$\hat{\boldsymbol{\alpha}}(t)$	near-end/source signal tonal components model parameter vector estimate (time-varying)
$\beta_n$	amplitude of $n$ th sinusoidal component in sinusoidal signal model
$\gamma(t)$	gain sequence (recursive identification)
$\boldsymbol{\gamma}(t)$	near-end/source signal noise components model parameter vector (time-varying)
$\hat{\boldsymbol{\gamma}}(t)$	near-end/source signal noise components model parameter vector estimate (time-varying)
$\zeta(k, t)$	prediction error criterion inverse weighting factor
$\zeta_1, \dots, \zeta_Q$	LP model prediction error filter pole angles (time-invariant, rad)
$\theta_1, \dots, \theta_P$	CPZLP model pole-zero angles (time-invariant, rad) [Ch. 8] LP model prediction error filter zero angles (time-invariant, rad) [Ch. 11]

$\theta_1(t), \dots, \theta_{n_A/2}(t)$	PZLP model pole-zero angles (time-varying, rad)
$\boldsymbol{\theta}$	parameter vector to be estimated (time-invariant)
$\boldsymbol{\theta}(t)$	parameter vector to be estimated (time-varying)
$\hat{\boldsymbol{\theta}}(t)$	parameter vector estimate (time-varying)
$\boldsymbol{\theta}_n$	$n \times 1$ subvector of parameter vector to be estimated (time-invariant)
$\hat{\boldsymbol{\theta}}_n^{(0)}$	CPZLP pole-zero angle initial estimate (time-invariant, $n$ th subproblem, rad)
$\hat{\boldsymbol{\theta}}_n^{(k)}$	CPZLP pole-zero angle iterative estimate (time-invariant, $k$ th iteration, $n$ th subproblem, rad)
$\hat{\boldsymbol{\theta}}_n^{(k)}$	subvector of CPZLP parameter vector estimate (time-invariant, $k$ th iteration, $n$ th subproblem)
$\nu$	LP model prediction error filter zero radius (time-invariant) [Ch. 11]
$\nu_1, \dots, \nu_{n_A/2}$	PZLP model prediction error filter zero radii (time-invariant) [Ch. 2]
$\nu_1, \dots, \nu_P$	LP model prediction error filter zero radii (time-invariant) [Ch. 11]
$\rho$	LP model prediction error filter pole radius (time-invariant) [Ch. 11]
$\rho_1, \dots, \rho_{n_A/2}$	PZLP model prediction error filter pole radii (time-invariant) [Ch. 2]
$\rho_1, \dots, \rho_Q$	LP model prediction error filter pole radii (time-invariant) [Ch. 11]
$\sigma^2(t)$	prediction error variance estimate (time-varying)
$\sigma_A^2(t)$	prediction error variance estimate from near-end/source signal tonal components model parameter estimation (time-varying)
$\sigma_C^2(t)$	prediction error variance estimate from near-end/source signal noise components model parameter estimation (time-varying)
$\sigma_j^2$	near-end/source excitation signal variance (piecewise time-invariant, frame index $j$ )
$\sigma_r^2$	noise signal variance in tonal signal model (time-invariant)
$\sigma_r^2(t)$	near-end/source excitation signal variance (time-varying)
$\sigma_t^2$	near-end/source excitation signal variance (time-varying)
$\hat{\sigma}_t^2$	near-end/source excitation signal variance estimate (time-varying)

$\sigma_{T,\bar{y}}^2(t)$	twin filter structure prefiltered microphone signal variance estimate (time-varying)
$\sigma_{T,\varepsilon}^2(t)$	twin filter structure foreground filter prediction error variance estimate (time-varying)
$\sigma_{T,\varepsilon_B}^2(t)$	twin filter structure background filter prediction error variance estimate (time-varying)
$\sigma_v^2$	near-end/source signal variance (time-invariant)
$\sigma_\varepsilon^2(t)$	prediction error variance estimate from smoothing instantaneous prediction error variance (time-varying)
$\Sigma$	diagonal matrix in inverse near-end/source signal covariance matrix Cholesky decomposition
$\Sigma_\lambda$	exponentially weighted diagonal matrix in inverse near-end/source signal covariance matrix Cholesky decomposition
$\Sigma_M(t)$	diagonal submatrix in inverse near-end/source signal covariance matrix Cholesky decomposition ( $M$ th order affine projection)
$\phi(t)$	phase modulation function
$\phi_n$	phase of $n$ th sinusoidal component in sinusoidal/tonal signal model (rad)
$\phi_{n,m}$	phase of $m$ th sinusoidal component in $n$ th monophonic component in polyphonic signal model (rad)
$\phi_k$	column vector in true echo/feedback path impulse response covariance matrix rank-one decomposition
$\Phi$	regularization matrix (echo/feedback path impulse response estimation)
$\xi$	reference value (echo/feedback path impulse response estimation) [Ch. 5]
	LP model parameter vector [Ch. 11]
$\xi(t)$	parameter vector to be estimated (time-varying)
$\hat{\xi}(t)$	parameter vector estimate (time-varying)
$\psi(t)$	gradient vector (recursive identification)
$\psi_A(t)$	gradient vector for near-end/source signal model identification (recursive identification)
$\psi_F(t)$	gradient vector for echo/feedback path identification (recursive identification)
$\omega_0$	fundamental frequency in monophonic signal model (rad)
$\omega_{0,n}$	fundamental frequency of $n$ th monophonic component in polyphonic signal model (rad)



$\omega_c$	biquadratic parametric equalizer filter center frequency (rad)
$\omega_{c,l}$	$l$ th second-order section center frequency in bank of adjustable notch filters (rad)
$\omega_i$	critical closed-loop system frequency (rad)
$\tilde{\omega}_i$	candidate howling component frequency (rad)
$\omega_n$	frequency of $n$ th sinusoidal component in sinusoidal/tonal signal model (rad)

### Parameters

$A$	direct path attenuation in 3-parameter echo/feedback path model
$B$	sound reinforcement system bandwidth [Ch. 2]
$B_l$	biquadratic parametric equalizer filter bandwidth [Ch. 7] $l$ th second-order section bandwidth in bank of adjustable notch filters (rad)
$c$	CPZLP Armijo's sufficient decrease condition scaling factor [Ch. 8]
$d$	speech onset detection threshold [Ch. 10] initial delay in 3-parameter echo/feedback path model [Ch. 5]
$d_0$	electro-acoustic forward path delay [Ch. 6, 9, 10]
$d_1$	AFC filterbank implementation processing delay
$d_2$	electro-acoustic forward path delay
$D$	adaptive filter delay
$f_m$	DM filter interpolation ratio [Ch. 2]
$f_s$	fractional pitch lag interpolation ratio [Ch. 2, 11, 12]
$f_s$	LPTV filter modulation frequency (Hz)
$f_s$	sampling frequency (Hz)
$G_0$	biquadratic parametric equalizer DC gain
$G_{0,l}$	$l$ th second-order section DC gain in bank of adjustable notch filters (dB)
$G_\pi$	biquadratic parametric equalizer Nyquist gain
$G_{\pi,l}$	$l$ th second-order section Nyquist gain in bank of adjustable notch filters (dB)
$G_B$	biquadratic parametric equalizer band edge gain
$G_{B,l}$	$l$ th second-order section band edge gain in bank of adjustable notch filters (dB)
$G_c$	biquadratic parametric equalizer notch gain
$G_{c,l}$	$l$ th second-order section notch gain in bank of adjustable notch filters (dB)

$G_{c,l}^{(0)}$	$l$ th second-order section initial notch gain in bank of adjustable notch filters (dB)
$\Delta G_{c,l}$	$l$ th second-order section notch gain decrease in bank of adjustable notch filters (dB)
$I$	half linear interpolation filter impulse response length
$K$	electro-acoustic forward path gain DM filter integer delay [Ch. 2]
$\Delta K$	electro-acoustic forward path gain increase (dB)
$k_0$	saturation level (NL-RLS algorithm)
$k_{\max}$	CPZLP maximum number of iterations per subproblem
$K_{\min}$	minimum integer pitch lag (pitch prediction)
$K_{\max}$	maximum integer pitch lag (pitch prediction)
$l$	DM filter fractional phase [Ch. 2]
$L$	number of multi-channel far-end/loudspeaker signals [Ch. 2] electro-acoustic forward path gain [Ch. 6] observation frame length [Ch. 11]
$L_F$	echo/feedback path impulse response length
$M$	frame length affine projection order [Ch. 2, 5] AFC filterbank implementation number of subbands [Ch. 10]
$M_{\text{BFGS}}$	CPZLP number of multiplications (BFGS method)
$M_{\text{GN}}$	CPZLP number of multiplications (GN method)
$M_{\text{LTP}}$	long-term prediction frame length
$M_n$	number of relevant harmonics (including fundamental frequency) in $n$ th monophonic component in polyphonic signal model
$M_{\text{SD}}$	CPZLP number of multiplications (SD method)
$n_A$	near-end/source signal model order near-end/source signal tonal components model order [Ch. 2, 12] far-end/loudspeaker signal model order [Ch. 4]
$n_B$	cascade model order of echo/feedback path and near-end/source signal model prediction error filter (two-channel adaptive filtering)
$n_C$	near-end/source signal noise components model order
$n_D$	cascade model order of near-end/source signal noise components model PEF and tonal components model PEF numerator

$n_F$	echo/feedback path model order
$n_{\hat{F}}$	echo/feedback path estimate order
$n_G$	electro-acoustic forward path model order
$n_H$	bank of adjustable notch filters order
$n_I$	PLP linear interpolation filter order
$n_L$	cascade model order of near-end/source signal noise components model PEF and tonal components model PEF numerator and echo/feedback path model (time-varying)
$n_S$	SLP anti-aliasing filter order
$n_\alpha$	near-end/source signal tonal components WLP model order
$N$	LPTV filter modulation period [Ch. 2] effective data window length [Ch. 3] simulation length [Ch. 3] observation frame length [Ch. 8] AFC filterbank implementation downsampling factor [Ch. 10] number of tonal/monophonic components in monophonic/polyphonic signal model [Ch. 11] number of sinusoidal components in sinusoidal signal model [Ch. 12]
$N_{\text{HO}}$	number of howling occurrences
$P$	frame hop size number of sinusoidal components in sinusoidal signal model [Ch. 8] LP model half denominator order [Ch. 11]
$P_0$	absolute power threshold for PTPR feature calculation
$Q$	biquadratic parametric equalizer filter Q factor [Ch. 7] LP model half numerator order [Ch. 11]
$Q_F$	echo/feedback path model stationarity time scale variable
$Q_{H_1}$	near-end/source signal tonal components model stationarity time scale variable
$Q_{H_2}$	near-end/source signal noise components model stationarity time scale variable
$Q_M$	number of signal frames taken into account for calculating interframe microphone signal features (IPMP, IMSD)
$Q_P$	number of spectral peaks to remove from microphone signal spectrum before estimating microphone signal power
$S$	number of multi-channel near-end/source signals
$T$	simulation length (s)

$T_{60}, T_{60}$	$T_{60}$ reverberation time (s)
$T_a$	adaptation treshold
$T_m$	LPTV filter modulation period (s)
$T_{\max}$	proactive notch filter activation threshold
$T_{\min}$	proactive notch filter deactivation threshold
$T_{\text{FEP}}$	FEP function threshold for howling detection (dB)
$T_{\text{PAPR}}$	PAPR feature threshold for howling detection (dB)
$T_{\text{PHPR}}$	PHPR feature threshold for howling detection (dB)
$T_{\text{PNPR}}(m)$	$m$ th neighbor PNPR feature threshold for howling detection (dB)
$T_s$	sampling period (s)
$\alpha$	half-wave rectification parameter [Ch. 2]
	regularization parameter [Ch. 2, 5]
$\beta$	LPTV filter modulation index [Ch. 2]
	noise floor power in 3-parameter echo/feedback path model [Ch. 5]
	regularization parameter [Ch. 6, 10]
$\beta_{n,k}$	CPZLP number of backtracking steps needed to satisfy Armijo's sufficient decrease condition ( $k$ th iteration, $n$ th subproblem)
$\bar{\beta}$	CPZLP average number of backtracking steps per iteration
$\bar{\beta}_n$	CPZLP average number of backtracking steps per iteration ( $n$ th subproblem)
$\gamma$	spectral subtraction factor [Ch. 2]
	CPZLP BFGS Powell's parameter [Ch. 8]
$\Gamma$	SLP model downsampling factor
$\gamma_1$	twin filter structure background to foreground prediction error energy threshold
$\gamma_2$	twin filter structure background prediction error to pre-filtered microphone signal energy threshold
$\gamma_3$	twin filter structure foreground to background prediction error energy threshold
$\delta$	proportionate adaptation regularization parameter [Ch. 5]
	regularization parameter [Ch. 12]
$\delta_F$	regularization parameter [Ch. 3]
$\Delta$	adaptation freeze duration in AFC adaptation control
$\Delta_f$	FM filter modulation depth (Hz)
$\Delta_K$	difference of maximum and minimum integer pitch lag (pitch prediction)

$\Delta_T$	twin filter structure energy comparison time interval
$\Delta_\tau$	DM filter modulation depth
$\kappa_n$	CPZLP final iteration index ( $n$ th subproblem)
$\bar{\kappa}$	CPZLP average number of iterations per subproblem
$\lambda$	exponential forgetting factor [Ch. 3]
	CPZLP BFGS damping variable [Ch. 8]
	WLP model warping parameter [Ch. 11, 12]
$\lambda_{2\text{ch}}$	exponential forgetting factor for two-channel adaptive filtering
$\lambda_A$	exponential forgetting factor for near-end/source signal model identification
$\lambda_{\text{Bark}}(\cdot)$	Bark-scale WLP model warping parameter
$\lambda_F$	exponential forgetting factor for echo/feedback path identification
$\lambda_s$	exponential forgetting factor for scale factor estimation (NL-RLS algorithm)
$\lambda_T$	exponential forgetting factor for twin filter structure prediction error variance estimation
$\lambda_\epsilon$	exponential forgetting factor for prediction error variance estimation
$\mu$	NLMS/APA step size parameter
$\mu_B$	NLMS background filter step size parameter
$\mu_F$	NLMS step size parameter
$\mu_k$	CPZLP step length ( $k$ th iteration)
$\mu_k^{(0)}$	CPZLP initial step length ( $k$ th iteration)
$\mu_k^{(m)}$	CPZLP step length after $m$ backtracking steps ( $k$ th iteration)
$\bar{\mu}$	LMS step size parameter
$\nu$	CPZLP backtracking contraction factor [Ch. 8]
$\rho$	proportionate adaptation anti-stalling parameter [Ch. 5]
	CPZLP model pole radius [Ch. 8]
$\Sigma_K$	sum of maximum and minimum integer pitch lag (pitch prediction)
$\tau$	exponential decay time constant in 3-parameter echo/feedback path model [Ch. 5]
	CPZLP termination tolerance [Ch. 8]
$\tau_0$	DM filter time delay offset
$\omega_m$	LPTV filter modulation frequency (rad)

### Mathematical functions and constants

$\forall$	for all
$\exists$	there exists

$\in$	belongs to
$\propto$	proportional to
$\sim$	similar to
$\approx$	approximately equal to
$\triangleq$	defined as
$\equiv$	equivalent to
$\ll$	much less than
$\rightarrow$	approaches
$\cap$	set intersection
$\wedge$	logical conjunction
$[\cdot]$	floor function
$\lceil \cdot \rceil$	ceiling function
$\lceil \cdot \rceil$	nearest integer function
$ \cdot $	absolute value (real numbers)
	modulus (complex numbers)
	Euclidian distance (geometry)
$\angle$	argument (complex numbers)
$\ \cdot\ , \ \cdot\ _2$	Euclidian vector norm, L2 norm
$\ \cdot\ _F$	Frobenius matrix norm
$(\cdot)^T$	matrix transpose
$(\cdot)^{-1}$	matrix inverse
$(\cdot) //$	matrix column space
$(\cdot)^\perp$	matrix orthogonal complement
$[\cdot]_{m \times n}$	matrix dimensions ( $m$ rows, $n$ columns)
$d/dx(\cdot)$	total (first) derivative w.r.t. $x$
$\partial/\partial x(\cdot)$	partial (first) derivative w.r.t. $x$
$\partial^2/\partial x^2(\cdot)$	partial second derivative w.r.t. $x$
$\mathbf{0}$	zero vector/matrix
$\mathbf{0}_{m \times n}$	$m \times n$ zero matrix
$\arctan(\cdot)$	arctangent function
$\cos(\cdot)$	cosine function
$\text{cov}\{\cdot\}$	covariance operator
$d(\cdot)$	Minkowski or $p$ -norm distance
$\text{diag}\{\cdot\}$	vector-to-diagonal-matrix operator
$E\{\cdot\}$	expectation operator
$\exp(\cdot)$	exponential function
$\mathbf{I}$	identity matrix
$\mathbf{I}_n$	$n \times n$ identity matrix
$\text{Im}\{\cdot\}$	imaginary part
$j$	imaginary unit, $\sqrt{-1}$

$J_0(\beta), J_1(\beta), \dots$	Bessel functions of the first kind
$\ln(\cdot)$	natural logarithm
$\log_{10}(\cdot)$	common logarithm
$\max(\cdot)$	maximum
$\min(\cdot)$	minimum
$\cdot \bmod x$	modulo- $x$ operator
$\mathbb{N}$	the set of natural numbers
$O(\cdot)$	asymptotic upper bound (Landau symbol)
$p(\cdot)$	probability density function
$q$	discrete time shift operator
$\mathbf{q}$	discrete time shift vector operator
$\mathbb{R}$	the set of real numbers
$\operatorname{Re}\{\cdot\}$	real part
$\sin(\cdot)$	sine function
$\operatorname{sinc}(\cdot)$	sinc function
$\operatorname{tr}\{\cdot\}$	trace operator
$V(\cdot)$	cost function
$V_n(\cdot)$	cost function ( $n$ th subproblem)
$\operatorname{var}\{\cdot\}$	variance operator
$\mathbb{Z}$	the set of integer numbers
$\delta(t)$	Dirac impulse function

## Acronyms and Abbreviations

2ch-AF	two-channel adaptive filtering
3D	three-dimensional
AAC	advanced audio coding
AC	alternating current
ACF	adaptive comb filter
A/D	analog-to-digital
AEC	acoustic echo cancellation
AEQ	automatic equalization
AFC	adaptive feedback cancellation
	acoustic feedback cancellation [Ch. 9]
AFC-FS	adaptive feedback cancellation with decorrelation by frequency shifting
AFC-NI	adaptive feedback cancellation with decorrelation by noise injection
AFC-PF	adaptive feedback cancellation with decorrelation by pre-filtering
AGC	automatic gain control
AIF	adaptive inverse filtering
AM	amplitude modulation

ANF	adaptive notch filter
ANSI	American National Standards Institute
APA	affine projection algorithm
AR	autoregressive
ARMA	autoregressive moving average
BFGS	BroydenFletcherGoldfarbShanno
BLUE	best linear unbiased estimator
BW	bandwidth
ca.	circa
CAPZ	common acoustical pole and zero
CD	compact disc
cf.	<i>confer</i> , compare with
CHA	cumulative harmonic analysis
cm	centimeters
CPZLP	constrained pole-zero linear prediction
CRLB	Cramér-Rao lower bound
CS	carrier suppression
D/A	digital-to-analog
dB	decibel
dBV	decibel (reference voltage = 1 V)
DC	direct current
DFT	discrete Fourier transform
DI	direct identification
DM	delay modulation
DR-RPE	dually regularized recursive prediction error
DTD	double-talk detector
e.g.	<i>exempli gratia</i> , for example
ENR	echo-to-near-end ratio
Eq.	equation
ERB	equivalent rectangular bandwidth
FDAF	frequency domain adaptive filter
FEP	feedback existence probability
FFT	fast Fourier transform
FIR	finite impulse response
FIR-ANF	finite impulse response adaptive notch filter
FLOPS	floating point operations
FM	frequency modulation
FNR	feedback-to-near-end ratio
FS	frequency shifting
GHz	gigahertz



GLS	generalized least squares
GN	Gauss-Newton
GWN	Gaussian white noise
HA	hearing aid(s)
HOLP	high-order linear prediction
HOP	howling occurrence probability
Hz	hertz
IDD	inter-peak dip depth
i.e.	<i>id est</i> , that is
IIR	infinite impulse response
IIR-ANF	infinite impulse response adaptive notch filter
IMSD	interframe magnitude slope deviation
IPMP	interframe peak magnitude persistence
kHz	kilohertz
LEC	line echo cancellation
LMMSE	linear minimum mean square error
LMR	Levenberg-Marquardt regularization
LMR-APA	Levenberg-Marquardt regularized affine projection algorithm
LMR-NLMS	Levenberg-Marquardt regularized normalized least mean squares
LMR-PEM-AFROW	Levenberg-Marquardt regularized prediction-error-method-based adaptive filtering with row operations
LMR-RLS	Levenberg-Marquardt regularized recursive least squares
LMS	least mean squares
LP	linear prediction
LP	linear prediction
LP <sub>AUTO</sub>	linear prediction with autocorrelation method
LP <sub>COV</sub>	linear prediction with covariance method
LPTV	linear periodically time-varying
LS	least squares
LTP	long-term predictor
LTV	linear time-varying
m	meters
MA	moving average
	misadjustment [Ch. 12]
MAP	maximum a posteriori
MDF	multidelay filter
MIL	matrix inversion lemma
MMSE	minimum mean square error

ms	milliseconds
MSE	mean square error
MSFE	mean square frequency error
MSG	maximum stable gain
$\Delta$ MSG	maximum stable gain increase
N/A	not applicable
NBF	null beamformer
NHS	notch-filter-based howling suppression
NHS-1	notch-filter-based howling suppression with howling detection based on PHPR and IPMP features
NHS-2	notch-filter-based howling suppression with howling detection based on PAPR feature
NHS-3	notch-filter-based howling suppression with howling detection based on PNPR and IMSD features
NLMS	normalized least mean squares
NL-NLMS	nonlinear normalized least mean squares
NL-RLS	nonlinear recursive least squares
no.	number
PA	public address
PBFDAF	partitioned block frequency domain adaptive filtering
PC	personal computer
PAPA	proportionate affine projection algorithm
PAPR	peak-to-average power ratio
PDF	probability density function
PE	prediction error
PEF	prediction error filter
PEM	prediction error method
PEM-AF	prediction-error-method-based adaptive filtering
PEM-AFROW	prediction-error-method-based adaptive filtering with row operations
PFC	phase-modulating feedback control
PFC-DM	phase-modulating feedback control based on sinusoidal delay modulation
PFC-FS	phase-modulating feedback control based on frequency shifting
PFC-PM	phase-modulating feedback control based on sinusoidal phase modulation
PHPR	peak-to-harmonic power ratio
PLP	pitch prediction
PM	phase modulation

PNLMS	proportionate normalized least mean squares
PNPR	peak-to-neighboring power ratio
PSD	power spectral density
PTPR	peak-to-threshold power ratio
PZLP	pole-zero linear prediction
rad	radians
RE	reverberation enhancement
RGLS	recursive generalized least squares
RIR	room impulse response
RLS	recursive least squares
RMS	root mean square
RPE	recursive prediction error
s	seconds
SD	frequency-weighted log-spectral signal distortion [Ch. 2] steepest descent [Ch. 8]
SFM	spectral flatness measure
SG-2ch-AF	stochastic gradient two-channel adaptive filtering
SG-NL-RLS	stochastic gradient nonlinear recursive least squares
SG-PEM-AF	stochastic gradient prediction-error-method-based adaptive filtering
SG-PEM-AFROW	stochastic gradient prediction-error-method-based adaptive filtering with row operations
SG-RLS	stochastic gradient recursive least squares
SG-RPE	stochastic gradient recursive prediction error
SLP	selective linear prediction
SNR	signal-to-noise ratio
SPL	sound pressure level
s.t.	subject to
STFT	short-time Fourier transform
STP	short-term predictor
TR	Tikhonov regularization
TRI	time to recover from instability
TR-RLS	Tikhonov regularized recursive least squares
TVAR	time-varying autoregressive
URLS	underdetermined recursive least squares
VRT	Flemish Radio and Television Network
vs.	versus
WLP	(frequency-)warped linear prediction
w.r.t.	with respect to



# Contents

Voorwoord	i
Abstract	v
Korte Inhoud	vii
Glossary	ix
Contents	xxxvii
Samenvatting	xlvi

## I Introduction

<b>1 Introduction and Overview</b>	<b>3</b>
1.1 Room Acoustic Signal Enhancement . . . . .	3
1.2 Acoustic Echo Cancellation . . . . .	6
1.2.1 Concept . . . . .	6
1.2.2 Double-Talk Robustness . . . . .	7
1.2.3 Acoustic Echo Path Undermodeling . . . . .	8
1.2.4 Poor Excitation and Regularization . . . . .	8
1.3 Acoustic Feedback Control . . . . .	10

1.3.1	Concept . . . . .	10
1.3.2	Notch-Filter-Based Howling Suppression . . . . .	11
1.3.3	Adaptive Feedback Cancellation . . . . .	13
1.4	Chapters and Publications Overview . . . . .	14
	Bibliography . . . . .	16
<b>2</b>	<b>State of the Art in Acoustic Feedback Control</b>	<b>21</b>
2.1	Introduction . . . . .	23
2.2	The Acoustic Feedback Problem . . . . .	25
2.3	State of the Art in Acoustic Feedback Control . . . . .	30
2.3.1	Phase Modulation Methods . . . . .	31
2.3.2	Gain Reduction Methods . . . . .	32
2.3.3	Spatial Filtering Methods . . . . .	36
2.3.4	Room Modeling Methods . . . . .	37
2.4	Phase-Modulating Feedback Control . . . . .	41
2.4.1	Concept . . . . .	41
2.4.2	Realization . . . . .	44
2.4.3	Discussion . . . . .	46
2.5	Notch-Filter-Based Howling Suppression . . . . .	47
2.5.1	Concept . . . . .	47
2.5.2	Realization . . . . .	49
2.5.3	Initialization . . . . .	55
2.5.4	Discussion . . . . .	57
2.6	Adaptive Feedback Cancellation . . . . .	58
2.6.1	Concept . . . . .	58
2.6.2	Realization . . . . .	61
2.6.3	Initialization . . . . .	70

2.6.4	Discussion . . . . .	71
2.7	Evaluation . . . . .	72
2.7.1	Evaluated Algorithms . . . . .	72
2.7.2	Evaluation Procedure . . . . .	75
2.7.3	Simulation Results . . . . .	78
2.8	Conclusion and Future Challenges . . . . .	81
	Bibliography . . . . .	86

## II Acoustic Echo Cancellation

<b>3</b>	<b>Double-Talk-Robust AEC</b>	<b>103</b>
3.1	Introduction . . . . .	105
3.2	Linear Estimation Background . . . . .	109
3.3	Prediction Error Identification Algorithms . . . . .	110
3.3.1	Prediction Error Criterion . . . . .	111
3.3.2	Recursive Prediction Error (RPE) Identification Algorithm	112
3.3.3	Prediction Error Identification Algorithms from AFC . . . . .	115
3.4	Implementation and Complexity . . . . .	118
3.4.1	Implementation Issues . . . . .	118
3.4.2	Computational Complexity . . . . .	119
3.5	Simulation Results . . . . .	121
3.6	Conclusions . . . . .	126
	Bibliography . . . . .	127
	Addendum . . . . .	131
<b>4</b>	<b>Acoustic Echo Path Undermodeling</b>	<b>133</b>
4.1	Introduction . . . . .	135

4.2	Undermodeled Room Impulse Response . . . . .	135
4.3	PEM-AFROW-Based Approach . . . . .	137
4.4	Simulations . . . . .	138
4.5	Conclusion . . . . .	140
	Bibliography . . . . .	140
<b>5</b>	<b>Optimal Regularization for AEC/AFC</b>	<b>147</b>
5.1	Introduction . . . . .	149
5.2	MSE Optimal Weighting and Regularization . . . . .	156
5.3	Source Signal and True RIR Statistics . . . . .	157
5.3.1	Source Signal Covariance Matrix $\mathbf{R}_v$ . . . . .	157
5.3.2	True RIR Covariance Matrix $\mathbf{R}_f$ . . . . .	159
5.4	Regularized RLS Algorithms . . . . .	162
5.5	Regularized APA and NLMS Algorithms . . . . .	165
5.6	Simulation Results . . . . .	168
5.6.1	Regularized RLS and PEM-AFROW Algorithms for Acoustic Echo Cancellation in a Stationary Environment . . . . .	169
5.6.2	Regularized NLMS and PEM-AFROW Algorithms for Acoustic Echo Cancellation in a Nonstationary Environment . . . . .	171
5.6.3	Regularized PEM-AFROW Algorithms for Adaptive Feedback Cancellation in a Stationary Environment . . . . .	175
5.7	Conclusion . . . . .	175
	Bibliography . . . . .	177
<b>6</b>	<b>Dual Regularization for AEC/AFC</b>	<b>183</b>
6.1	Introduction . . . . .	185
6.2	Scaling Ambiguity in the RPE Algorithm . . . . .	187
6.3	Dual Regularization . . . . .	191



6.4 Simulation Results . . . . . 192  
6.5 Conclusion . . . . . 198  
Bibliography . . . . . 199

### III Acoustic Feedback Control

**7 Biquadratic Notch Filter Design 203**  
7.1 Introduction . . . . . 205  
7.2 Design Procedure . . . . . 207  
    7.2.1 Notch Filters . . . . . 208  
    7.2.2 Resonance Filters . . . . . 211  
7.3 Design Examples . . . . . 214  
7.4 Conclusion . . . . . 217  
Bibliography . . . . . 217

**8 Efficient Parametric Frequency Estimation 219**  
8.1 Introduction . . . . . 221  
8.2 Constrained Pole-Zero Linear Prediction . . . . . 222  
    8.2.1 Signal Model . . . . . 222  
    8.2.2 Decoupled Optimization . . . . . 223  
    8.2.3 Gradient and Hessian Calculation . . . . . 226  
8.3 Computational Complexity . . . . . 227  
8.4 Simulation Results . . . . . 229  
8.5 Conclusion . . . . . 230  
Bibliography . . . . . 233

**9 AFC for Speech Applications 235**  
9.1 Introduction . . . . . 237

9.2	Prediction Error Method . . . . .	240
9.3	Proposed Algorithm: PEM-AFROW . . . . .	242
9.4	Long-Term Prediction . . . . .	247
9.5	Algorithm Description . . . . .	249
9.6	Complexity . . . . .	249
9.7	Simulation Results . . . . .	251
9.8	Conclusion . . . . .	253
	Bibliography . . . . .	253
<b>10</b>	<b>Robust and Efficient AFC Implementation</b>	<b>257</b>
10.1	Introduction . . . . .	259
10.2	Robustness . . . . .	262
10.2.1	Adaptation Control . . . . .	262
10.2.2	Prior Knowledge of the Feedback Path . . . . .	264
10.2.3	Foreground/Background Filter . . . . .	266
10.2.4	Notch Filtering . . . . .	267
10.2.5	Nonlinearities . . . . .	272
10.2.6	Highpass Filtering . . . . .	273
10.3	Subband and Frequency Domain Implementations . . . . .	273
10.3.1	Oversampled DFT-Modulated Filterbank Implementation	273
10.3.2	Frequency Domain Implementation . . . . .	276
10.4	Simulations . . . . .	276
10.4.1	Onset Detection . . . . .	277
10.4.2	Foreground/Background Filter . . . . .	277
10.4.3	Proactive Notch Filtering . . . . .	278
10.4.4	Highpass Filter . . . . .	279
10.4.5	Combined Real-Time System . . . . .	279

<i>Contents</i>	xliii
10.5 Conclusions . . . . .	279
Bibliography . . . . .	280
<b>11 Linear Prediction of Audio Signals</b>	<b>283</b>
11.1 Introduction . . . . .	285
11.2 Preliminaries . . . . .	288
11.2.1 Tonal Audio Signal Model . . . . .	288
11.2.2 Linear Prediction Criterion . . . . .	290
11.3 Conventional Linear Prediction Model . . . . .	292
11.4 Alternative Linear Prediction Models . . . . .	297
11.4.1 Constrained Pole-Zero LP Model . . . . .	297
11.4.2 High-Order LP model . . . . .	302
11.4.3 Pitch Prediction Model . . . . .	303
11.4.4 Warped LP Model . . . . .	306
11.4.5 Selective LP Model . . . . .	307
11.5 Simulation Results . . . . .	309
11.5.1 Synthetic Audio Signal . . . . .	311
11.5.2 Monophonic Audio Signal . . . . .	316
11.5.3 Polyphonic Audio Signal . . . . .	318
11.6 Conclusion . . . . .	322
Bibliography . . . . .	324
<b>12 AFC for Audio Applications</b>	<b>331</b>
12.1 Introduction . . . . .	333
12.2 Adaptive Feedback Cancellation . . . . .	337
12.2.1 Problem Description . . . . .	337
12.2.2 Adaptive Feedback Cancellation . . . . .	338

12.3 PEM-Based AFC . . . . .	339
12.3.1 Data Model . . . . .	339
12.3.2 Prediction Error Identification Algorithm . . . . .	342
12.4 Identifiability Conditions . . . . .	344
12.4.1 LP, PLP, and SLP Near-End Tonal Components Models	344
12.4.2 WLP Near-End Tonal Components Model . . . . .	344
12.4.3 PZLP Near-End Tonal Components Model . . . . .	345
12.5 Algorithm Details . . . . .	347
12.5.1 LP, PLP, and SLP Near-End Tonal Components Models	348
12.5.2 WLP Near-End Tonal Components Model . . . . .	349
12.5.3 PZLP Near-End Tonal Components Model . . . . .	351
12.6 Computational Complexity and Model Approximations . . . . .	354
12.6.1 Computational Complexity . . . . .	354
12.6.2 Model Approximations . . . . .	356
12.7 Simulation Results . . . . .	358
12.8 Conclusion . . . . .	363
Bibliography . . . . .	364
<b>13 Conclusions and Suggestions for Future Research</b>	<b>371</b>
13.1 Summary and Conclusions . . . . .	371
13.2 Suggestions for Future Research . . . . .	374
Bibliography . . . . .	375
<b>Publication List</b>	<b>377</b>
<b>Curriculum Vitae</b>	<b>381</b>

# Samenvatting

## Ontwerp en evaluatie van digitale signaalverwerkingsalgoritmen voor akoestische feedback- en echo-onderdrukking

### Hoofdstuk 1: Inleiding en overzicht

In **Paragraaf 1.1** wordt het werk beschreven in dit doctoraatsproefschrift gekaderd en gemotiveerd. Geluidssignalen vormen een fundamentele schakel in de menselijke communicatie, en geluidskwaliteit is daarbij van groot belang. Akoestische galm is n van de verschijnselen die aan de oorsprong ligt van een mogelijk gebrek aan geluidskwaliteit. Zowel het akoestische-echoprobleem als het akoestische-feedbackprobleem zijn het resultaat van akoestische galm, zij het in een verschillende situatie. Wanneer een geluidssignaal wordt weergegeven door een luidspreker die zich in een gesloten akoestische ruimte bevindt en wanneer in die ruimte bovendien een microfoon aanwezig is om een lokaal geluidssignaal te registreren, dan bestaat het risico dat het luidsprekersignaal na reflecties tegen de wanden van de ruimte op de microfoon terecht komt. Dit probleem wordt het akoestische-echoprobleem genoemd en doet zich voornamelijk voor in handenvrije-telefonietoepassingen. Wanneer het luidsprekersignaal bovendien een versterkte versie is van het microfoonsignaal dat in dezelfde ruimte wordt geregistreerd dan ontstaat een gesloten signaallus die kan leiden tot oscillaties die worden waargenomen als onaangename fluittonen. Dit probleem staat bekend als het akoestische-feedbackprobleem en doet zich o.a. voor in publieke omroepssystemen, hoorapparaten en communicatiesystemen in voertuigen.

In **Paragraaf 1.2** wordt het concept van akoestische-echo-onderdrukking beschreven en worden drie open problemen toegelicht die in het proefschrift aan bod zullen komen. Het eerste probleem is gerelateerd aan het zogenaamde dubbelspraakprobleem: wanneer beide sprekers aan weerszijden van het communicatiekanaal tegelijk actief zijn, heeft het adaptief echo-onderdrukkingfilter de neiging om te divergeren. Aangezien dubbelspraakdetectoren dit probleem

slechts gedeeltelijk kunnen oplossen is er nood aan dubbelspraakrobuuste adaptieve-filteralgoritmen (zie **Hoofdstuk 3**). Het tweede probleem is gekend als ondermodellering en doet zich voor wanneer de orde van het akoestische-echopad groter is dan de orde van het adaptief echo-onderdrukkingsfilter. In dat geval kan de kwaliteit van de resulterende schatting van het echopad ondermaats zijn ten gevolge van een grote fout en variantie, waardoor een andere aanpak aangewezen lijkt (zie **Hoofdstuk 4**). Het derde probleem ontstaat wanneer het luidsprekersignaal (afkomstig van de spreker aan de andere kant van het communicatiekanaal) onvoldoende exciterend is om het echopad te kunnen identificeren. Het zal mogelijk blijken om de bestaande oplossing, gekend als regularisatie, op een zodanige manier te veralgemenen dat eventuele a-priorikennis over het akoestische-echopad kan worden geïntegreerd in het adaptieve-filteralgoritme (zie **Hoofdstuk 5**). Bovendien kan met een vergelijkbare aanpak een numeriek probleem worden opgelost dat zich voordoet in enkele van de dubbelspraakrobuuste algoritmen voorgesteld in **Hoofdstuk 3** (zie **Hoofdstuk 6**).

**Paragraaf 1.3** handelt over het concept en de open problemen inzake akoestische-feedbackbeheersing. Eerst en vooral wordt het gebrek aan een vergelijkende studie van bestaande oplossingen aangekaart, wat meteen de motivatie vormt voor het uitgebreide literatuuroverzicht en de vergelijkende evaluatie die beschreven worden in **Hoofdstuk 2**. Op het vlak van inkepingfiltergebaseerde fluittoononderdrukking wordt de state of the art met betrekking tot fluittoondetectie en inkepingfilterontwerp beschreven. De bestaande methodes voor fluittoondetectie zijn hoofdzakelijk reactief van aard, waardoor fluittonen kunnen worden waargenomen voordat ze onderdrukt worden. Een proactieve aanpak kan dit probleem voorkomen en presteert bijgevolg beter op het gebied van geluidskwaliteit (zie **Hoofdstuk 10**). Verder wordt in **Hoofdstuk 8** een nieuwe en goedkope parametrische frequentieschattingmethode voorgesteld die kan worden gebruikt voor fluittoondetectie. Voor het ontwerp van inkepingfilters wordt meestal een beroep gedaan op een ontwerpprocedure voor parametrische-egaliseringsfilters, die typisch bestaat in het ontwerp van een analoog egaliseringsfilter gevolgd door een bilineaire transformatie. In **Hoofdstuk 7** wordt een nieuwe ontwerpmethodologie beschreven waarmee parametrische-egaliseringsfilters (waaronder inkepingfilters) direct in het digitale domein kunnen worden ontworpen. Tenslotte worden enkele open problemen besproken rond de veelbelovende techniek van adaptieve feedbackonderdrukking. Voor spraaktoepassingen zijn de geluidskwaliteit en de betrouwbaarheid van bestaande oplossingen vaak ontoereikend, waardoor de ontwikkeling een nieuwe aanpak noodzakelijk blijkt (zie **Hoofdstuk 9** en **10**). Voor audiotoeepassingen daarentegen zijn nog geen adaptieve feedbackonderdrukkingsalgoritmen beschikbaar, en dergelijke algoritmen worden daarom voorgesteld in **Hoofdstuk 11** en **12**.

In **Paragraaf 1.4** wordt het inleidende hoofdstuk afgerond met een overzicht

van de publicaties die deel uitmaken van dit proefschrift.

## Hoofdstuk 2: State of the art inzake akoestische-feedback-beheersing

Het eerste doel van dit hoofdstuk bestaat erin een overzichtelijk beeld te schep-  
pen van het onderzoekswerk rond akoestische-feedbackbeheersing dat gedurende  
de voorbije 50 jaar werd gerapporteerd in onderzoeksartikels en octrooien. Het  
tweede doel is een vergelijkende evaluatie te maken van de meest gebruikte  
methodes voor akoestische-feedbackbeheersing, waaronder enkele van de metho-  
des die in dit proefschrift worden voorgesteld. Het derde doel bestaat in  
het formuleren van een aantal belangrijke uitdagingen met de bedoeling om de  
richting van toekomstig onderzoek uit te stippelen.

In **Paragraaf 2.2** wordt het akoestische-feedbackprobleem beschreven in een  
discrete-tijd-signaalverwerkingscontext. Het centrale resultaat in deze para-  
graaf is de formulering van de Nyquist-stabiliteitsvoorwaarden, op basis waar-  
van nagenoeg elke methode voor akoestische-feedbackbeheersing kan worden  
afgeleid. Verder worden de eigenschappen van kamerakoestische overdrachts-  
functies summier beschreven en wordt de maximale stabiele versterking (MSG)  
gedefinieerd als belangrijkste performantiemaat voor feedbackbeheersing.

Het literatuuroverzicht in **Paragraaf 2.3** is het resultaat van een uitgebreide  
literatuurstudie van een honderdtal onderzoeksartikels en octrooien rond akoes-  
tische-feedbackbeheersing die werden gepubliceerd tussen 1958 en 2008. In het  
literatuuroverzicht worden de bestaande methodes geclassificeerd in vier cate-  
gorieën: fasemodulatiegebaseerde methodes, signaalverzwakkingsmethodes,  
ruimtelijke-filteringmethodes en methodes gebaseerd op kamermodellering. In  
fasemodulatiegebaseerde methodes wordt de fase van het microfoonsignaal aan-  
gepast in het elektroakoestisch voorwaarts pad zodat constructieve interfe-  
rentie van het feedbacksignaal met het bronsignaal ter hoogte van de micro-  
foon vermeden wordt. Het doel van signaalverzwakkingsmethodes bestaat erin  
een frequentie-afhankelijke reductie van de versterking in het elektroakoestisch  
voorwaarts pad uit te voeren in kritische frequentiebanden. Bij ruimtelijke-  
filteringmethodes wordt een bundelsturing uitgevoerd m.b.v. luidspreker- en/of  
microfoonroosters om de directe terugkoppeling tussen de luidsprekers en de  
microfoons te reduceren. methodes gebaseerd op kamermodellering beogen de  
modellering en identificatie van de kamerakoestiek voor het ontwerp van een  
feedbackonderdrukkingsfilter of een egalisatiefilter.

In **Paragraaf 2.4** wordt fasemodulatiegebaseerde feedbackbeheersing (PFC) in  
detail besproken, d.i. de oudste techniek voor akoestische-feedbackbeheersing  
die recent veel belangstelling kreeg bij het ontwerp van galmverbeteringssys-  
temen. Het concept van PFC wordt geïllustreerd a.h.v. de theorie van line-  
aire periodiek tijdsvariante filters en er wordt een onderscheid gemaakt tussen

vier types van fasemodulatie. Verder wordt de realisatie van dergelijke filters in detail beschreven, worden suggesties gemaakt voor een gepaste keuze van de parameterwaarden en worden de voor- en nadelen van de PFC-methode toegelicht.

**Paragraaf 2.5** handelt over inkepingsfiltergebaseerde fluittoononderdrukking (NHS), wat zonder twijfel de meest gebruikte techniek is voor akoestische-feedbackbeheersing in geluidsversterkingssystemen. De nadruk ligt op tweeledige NHS-methodes, waarvan achtereenvolgens het gedeelte voor fluittoondectie en het gedeelte voor inkepingsfilterontwerp in detail worden besproken. Deze paragraaf wordt afgesloten met enkele praktische overwegingen voor het initialiseren van de NHS-methode in een geluidversterkingssysteem en met een overzicht van de pro's en contra's voor deze methode.

In **Paragraaf 2.6** wordt een meer recente techniek voor akoestische-feedbackbeheersing besproken, meer bepaald adaptieve feedbackonderdrukking (AFC). Deze techniek wordt gekaderd in de theorie van geslotenlusidentificatie, van waaruit de combinatie van het adaptief filter met een decorrelatiemethode noodzakelijk blijkt. De bestaande decorrelatiemethodes worden in detail beschreven, alsook enkele recente technieken om de performantie van de AFC-methode verder te verbeteren, zoals postfiltering en regularisatie. Tenslotte worden de sterktes en zwaktes van de AFC-methode toegelicht.

In **Paragraaf 2.7** worden de uitvoering en de resultaten besproken van een vergelijkende evaluatie van de drie methodes die in de voorgaande paragrafen in detail aan bod kwamen. Voor elk van deze drie methodes worden drie algoritmen uit de literatuur geselecteerd die representatief zijn voor de state of the art. Die negen algoritmen worden vervolgens geëvalueerd op basis van de haalbare versterking, de geluidskwaliteit en de betrouwbaarheid. Zowel voor spraak- als audiosignalen blijkt de AFC-methode superieur te zijn inzake haalbare versterking en geluidskwaliteit, en biedt deze methode een relatief hoge betrouwbaarheid. Van de drie geëvalueerde AFC-algoritmen scoort AFC op basis van prefiltering (AFC-PF) over het algemeen het best, d.i. een algoritme dat in dit proefschrift wordt ontwikkeld (zie **Hoofdstuk 9** en **12**).

Tot slot worden in **Paragraaf 2.8** de belangrijkste uitdagingen voor toekomstig onderzoek in akoestische-feedbackbeheersing op een rijtje gezet. Gezien de relatief superieure performantie van de AFC-methode, ligt het voor de hand dat voornamelijk die methode het onderwerp van toekomstig onderzoek zal uitmaken. De grootste hindernis die de AFC-methode met zich meebrengt is de hoge rekencomplexiteit. Aansluitend daarbij is de uitbreiding van de AFC-methode naar meerkanaalssystemen hoegenaamd niet triviaal. Enkele suggesties om die twee hindernissen in de toekomst te overwinnen worden aangehaald, waarbij de nadruk ligt op het gebruik van een alternatief model voor het akoestische-feedbackpad.



## Hoofdstuk 3: Dubbelspraakrobuuste AEC

In dit hoofdstuk wordt een nieuwe oplossing aangereikt voor het dubbelspraakprobleem in akoestische-echo-onderdrukking (AEC). In **Paragraaf 3.1** wordt het dubbelspraakprobleem uiteengezet en wordt een literatuurstudie rond dit onderwerp gepresenteerd. De meest gebruikte manier om divergentie van het adaptief filter tegen te gaan tijdens dubbelspraak bestaat in het inschakelen van een dubbelspraakdetector (DTD) die de adaptatie desgevallend bevriest. Toch is er nood aan dubbelspraakrobuuste adaptieve-filteralgoritmen om de tekortkomingen van de DTD op te vangen en om adaptatie tijdens continue dubbelspraak mogelijk te maken.

In **Paragraaf 3.2** wordt een theoretisch kader geschetst van waaruit de voorgestelde dubbelspraakrobuuste AEC-algoritmen kunnen worden afgeleid. Het principe van minimale-variantieschatting suggereert dat het dubbelspraakprobleem groter is wanneer het bronsignaal een grotere spectrale kleuring vertoont. Bovendien blijkt dat de variantie van de traditionele kleinste-kwadratenschatter kan worden verlaagd door de karakteristieken van het bronsignaal in rekening te brengen bij de schatting.

In **Paragraaf 3.3** worden enkele algoritmen voor dubbelspraakrobuuste AEC afgeleid. In de eerste plaats wordt er een verband gelegd tussen de theorie van minimale-variantieschatting en de predictiefoutmethode (PEM) voor systeemidentificatie. Er wordt een predictiefoutcriterium opgesteld op basis waarvan een algoritme voor recursieve predictiefoutidentificatie (RPE) van het akoestische echopad en een autoregressief bronsignaalmodel wordt afgeleid. Het RPE-algoritme blijkt een veralgemening te zijn van drie algoritmen die eerder werden voorgesteld in het kader van adaptieve feedbackonderdrukking.

**Paragraaf 3.4** handelt over de implementatie en complexiteit van de voorgestelde algoritmen. Zowel een Gauss-Newton-implementatie als een stochastische-gradientimplementatie worden besproken, en er worden richtlijnen gegeven omtrent de keuze van de orde van het bronsignaalmodel. Tenslotte wordt de complexiteit van de voorgestelde algoritmen vergeleken met de complexiteit van enkele bestaande adaptieve-filteralgoritmen.

In **Paragraaf 3.5** worden simulatieresultaten besproken voor vier verschillende scenario's: continue dubbelspraak, abrupte dubbelspraak, echopadwijziging tijdens enkelspraak en echopadwijziging tijdens dubbelspraak. De voornaamste vaststellingen zijn dat het bestaande recursieve kleinste-kwadratenalgoritme (RLS) goed presteert tijdens continue dubbelspraak maar veel minder robuust is tegen abrupte dubbelspraak in vergelijking met de voorgestelde algoritmen (Gauss-Newton-implementatie). In het geval van een stochastische-gradientimplementatie vertonen de voorgestelde algoritmen een opmerkelijk beter convergentiegedrag dan het bestaande genormaliseerde kleinste-gemiddelde-kwadratenalgoritme (NLMS).

## Hoofdstuk 4: Ondermodellering van het akoestische-echopad

Dit hoofdstuk handelt over een nieuwe aanpak voor het probleem van ondermodellering in akoestische-echo-onderdrukking (AEC). Dit probleem doet zich voor wanneer de orde van het adaptief filter in AEC aanzienlijk kleiner is dan de lengte van de impulsresponsie van het akoestische-echopad. In **Paragraaf 4.2** wordt het ondermodellingsprobleem geanalyseerd: het residuele echosignaal ten gevolge van ondermodellering blijkt zowel een fout als een variantie in de schatting van het akoestische-echopad te veroorzaken. Indien echter het luidsprekersignaal een ongecorrleerd signaal is, dan is de fout op de schatting gelijk aan nul.

In **Paragraaf 4.3** wordt een PEM-gebaseerde aanpak van het ondermodellingsprobleem voorgesteld. Door in de identificatie van het akoestische-echopad een prefiltering uit te voeren met het predictiefoutfilter van een autoregressief model van het echo-gecompenseerde signaal, kan zowel de fout als de variantie in de resulterende schatting gecontroleerd worden. De variantie kan bijgevolg worden verlaagd, terwijl de fout kan worden gelokaliseerd in de laatste coëfficiënten van het adaptief filter.

In **Paragraaf 4.4** wordt de performantie van de voorgestelde aanpak geverifieerd op basis van simulatieresultaten. De voornaamste vaststelling is dat bij een directe identificatie van het akoestische-echopad (zonder prefiltering) de variantie ten gevolge van ondermodellering ontoelaatbaar hoog is. Met de voorgestelde aanpak blijkt het inderdaad mogelijk te zijn om de variantie drastisch te verlagen. De fout op de schatting blijft -zoals voorspeld- na prefiltering geconcentreerd in de laatste adaptieve-filtercoëfficiënten en kan bijgevolg worden verwijderd door de impulsresponsie van het adaptief filter af te kappen.

## Hoofdstuk 5: Optimale regularisatie voor AEC/AFC

In dit hoofdstuk wordt het probleem van onvoldoende excitatie in adaptieve-filtertoepassingen zoals akoestische-echo-onderdrukking (AEC) en adaptieve feedbackonderdrukking (AFC) behandeld. Dat probleem doet zich vaak voor in kamerakoestische toepassingen ten gevolge van de lengte van de kamerimpulsresponsie (RIR) en de spectrale kleuring van het luidsprekersignaal, en kan leiden tot een trage convergentie en numerieke problemen in het adaptief filter.

In **Paragraaf 5.1** wordt het probleem van onvoldoende excitatie voorgesteld, alsook de manier waarop het probleem doorgaans wordt aangepakt. Bestaande adaptieve-filteralgoritmen omvatten meestal een ad-hocoplossing voor het excitatieprobleem die bestaat in het optellen van een gescaleerde eenheidsmatrix bij de autocorrelatiematrix van het luidsprekersignaal voordat die laatste geïnverteerd wordt. Die zogenaamde regularisatie vinden we in de literatuur

terug in twee varianten (Tikhonov- en Levenberg-Marquardtregularisatie) die schijnbaar niet gerelateerd zijn. Het doel van dit hoofdstuk bestaat erin om een theoretisch kader te scheppen waarin beide regularisatievarianten kunnen worden geïnterpreteerd als speciale gevallen van een meer algemene regularisatie-aanpak, alsook om de optimaliteit van de bestaande technieken te onderzoeken en te verbeteren.

In **Paragraaf 5.2** wordt de regularisatie-aanpak gekaderd binnen de theorie van Bayesiaanse minimale-gemiddelde-kwadratisc-foutschatting (MMSE). Daaruit blijkt dat een optimale benadering van de regularisatie-aanpak mogelijk is door zowel het bronsignaal als de RIR te beschouwen als stochastische vectorprocessen, en zo mogelijk a-priorikennis over deze processen te integreren in het schattingsproces. Die benadering leidt tot het gebruik van een regularisatiematrix die niet noodzakelijk een gescaleerde eenheidsmatrix is.

In **Paragraaf 5.3** wordt gesuggereerd op welke manier a-priorikennis over het bronsignaal en de RIR kan worden geconstrueerd. De covariantiematrix van het bronsignaal kan worden herleid tot een bovendriehoeksbandmatrix indien een autoregressief signaalmodel wordt verondersteld. De coëfficiënten van die matrix kunnen worden geschat m.b.v. de PEM-gebaseerde aanpak uit **Hoofdstuk 3**. De verwachte waarde van de RIR blijkt een cruciale rol te spelen in de regularisatie-aanpak: indien de verwachte waarde wordt gelijkgesteld aan nul bekomt men een Tikhonovregularisatie, en indien de verwachte waarde wordt gelijkgesteld aan de adaptieve RIR-schatting op het vorige tijdstip bekomt men een Levenberg-Marquardtregularisatie. De covariantiematrix van de RIR kan worden geconstrueerd op basis van een initiële RIR-meting of van een 3-parameter-RIR-model waarin kennis over de akoestische opstelling vervat zit.

In **Paragraaf 5.4** wordt de voorgestelde regularisatie-aanpak toegepast op het exponentieel gewogen recursieve kleinste-kwadratenalgoritme (RLS). De twee regularisatievarianten (Tikhonov- en Levenberg-Marquardtregularisatie) leiden tot twee nieuwe RLS-algoritmen waarin a-priorikennis over het bronsignaal en de RIR kan worden opgenomen. Er worden bovendien twee methodes aangehaald om een expliciete matrixinversie in de voorgestelde RLS-algoritmen te omzeilen, waardoor de nodige rekenkracht kan worden gereduceerd.

Vervolgens wordt in **Paragraaf 5.5** de voorgestelde regularisatie-aanpak toegepast op zogenaamde ondergedetermineerde RLS-algoritmen, in het bijzonder het NLMS-algoritme en het affiene-projectie-algoritme (APA). In dit geval blijkt de voorgestelde regularisatie-aanpak te resulteren in een algoritme dat twee bijzondere eigenschappen vertoont: adaptatielek en proportionele adaptatie. Aangezien adaptatielek een eerder negatieve invloed lijkt te hebben in kamerakoestische toepassingen, ligt de nadruk in deze paragraaf op de Levenberg-Marquardt-regularisatievariant aangezien daarmee enkel een proportionele adaptatie wordt bekomen. Dit leidt tot nieuwe NLMS- en APA-algoritmen, en bovendien tot een nieuwe interpretatie van de bestaande proportionele-

adaptatie-algoritmen.

Tenslotte wordt in **Paragraaf 5.6** de performantie van de voorgestelde geregulariseerde algoritmen vergeleken met state-of-the-art-adaptieve-filteralgoritmen in drie scenario's: RLS-gebaseerde AEC in een stationaire akoestische omgeving, NLMS-gebaseerde AEC in een niet-stationaire akoestische omgeving en NLMS-gebaseerde AFC in een stationaire akoestische omgeving. Telkens blijkt dat hoe meer a-priorikennis over het bronsignaal en de RIR wordt gebruikt, des te beter het convergentiegedrag van het adaptief algoritme is.

## Hoofdstuk 6: Duale regularisatie voor AEC/AFC

Dit hoofdstuk handelt over een numeriek probleem dat zich voordoet in het RPE-algoritme dat werd afgeleid in **Hoofdstuk 3**. De voorgestelde oplossing bestaat in een regularisatie van de schatting van het bronsignaalmodel, waarbij eventueel a-priorikennis over het bronsignaalmodel kan worden geïntegreerd.

In **Paragraaf 6.2** wordt een schalingsambiguïteit in het RPE-algoritme blootgelegd die aan de basis ligt van het geobserveerde numerieke probleem. Als de hulpvariabelen in het RPE-algoritme worden geschaald met een factor (of een functie daarvan) die aanzienlijk groter is dan de inverse norm van de impulsresponsie van het bronsignaalmodelpredictiefoutfilter, dan kan de resulterende schatting van de kamerimpulsresponsie (RIR) toch identiek zijn aan de schatting in het geval er geen schaling optreedt. De schalingsambiguïteit kan bijgevolg pas worden waargenomen in het convergentiegedrag van de adaptieve RIR-schatting wanneer er numerieke overloop optreedt.

In **Paragraaf 6.3** wordt een oplossing voor dit probleem voorgesteld die steunt op het concept van optimale regularisatie (zie **Hoofdstuk 5**). Naast de (prmale) regularisatieterm waarmee het probleem van onvoldoende excitatie in het luidsprekersignaal wordt aangepakt, kan een tweede (duale) regularisatieterm aan het predictiefoutcriterium worden toegevoegd om de ongewenste schaling van de coëfficiënten van het bronsignaalmodel te voorkomen. Bovendien kan in die duale regularisatieterm a-priorikennis over het bronsignaalmodel worden geïntegreerd.

In **Paragraaf 6.4** worden simulatieresultaten voorgesteld waaruit blijkt dat het numerieke probleem in het RPE-algoritme effectief wordt opgelost wanneer de duale regularisatie wordt toegepast. Daarnaast wordt een werkwijze voorgesteld en geëvalueerd om a-priorikennis over het bronsignaalmodel te construeren indien het bronsignaal een spraaksignaal is.

## Hoofdstuk 7: Bikwadratisch inkepingsfilterontwerp

In dit hoofdstuk wordt een nieuwe ontwerpmethodologie voor bikwadratische parametrische-egaliseringsfilters voorgesteld, i.e., tweede-ordefilters met een oneindige impulsresponsie (IIR) die het signaalspectrum slechts wijzigen in een smalle frequentieband rondom een centrale frequentie. De voorgestelde ontwerpmethodologie wordt in **Hoofdstuk 2** gebruikt voor het ontwerp van inkepingsfilters voor akoestische-feedbackbeheersing.

In **Paragraaf 7.1** wordt het concept van parametrische-egaliseringsfilterontwerp uiteengezet a.h.v. een literatuurstudie en enkele toepassingen, en worden de ontwerpdoelstellingen geformuleerd in een stelsel van vijf ontwerpvergelijkingen. Terwijl nagenoeg alle bestaande ontwerpmethododes vertrekken van een analoog filterontwerp gevolgd door een bilineaire transformatie, werkt de voorgestelde methode volledig in het digitale domein.

De nieuwe ontwerpmethodologie wordt vervolgens ontwikkeld in **Paragraaf 7.2**, waar een onderscheid wordt gemaakt tussen inkepingsfilter- en resonantiefilterontwerp. De methode is gebaseerd op de techniek van pool- en nulplaatsing en wordt afgeleid door gebruik te maken van de grafische methode om de frequentieresponsie van een lineair en tijdsinvariant filter te berekenen. De resulterende ontwerpvergelijkingen kunnen aanzienlijk vereenvoudigd worden door veronderstellingen te maken over de onderlinge ligging van de polen en de nullen, en over hun ligging ten opzichte van de eenheidsdriehoek in het complexe vlak.

Tenslotte wordt de performantie van de voorgestelde ontwerpmethodologie geïllustreerd a.h.v. vijf ontwerpvoorbeelden in **Paragraaf 7.3**. In de eerste vier voorbeelden wordt de magnituderesponsie van het resulterend parametrische-egaliseringsfilter uitgezet voor verschillende waarden van de bandbreedte, de resonantieversteviging en de centrale frequentie. In het vijfde voorbeeld wordt de voorgestelde methode vergeleken met drie state-of-the-art-ontwerpmethododes, waaruit blijkt dat de nieuwe methode even accuraat is als de bestaande methododes en tegelijkertijd een meer intuïtief filterontwerp toelaat.

## Hoofdstuk 8: Efficiënte parametrische frequentieschatting

In dit hoofdstuk wordt een goedkope methode voorgesteld om de frequenties van een som van sinusoidale signalen in additieve ruis te schatten. In **Paragraaf 8.1** wordt een beknopt overzicht gegeven van de vele oplossingen die in de literatuur voorhanden zijn voor dit probleem. Een eerste belangrijke vaststelling is dat de relatief goedkope niet-parametrische frequentieschattingmethododes een grote hoeveelheid signaalbemonsteringen nodig hebben om een goede frequentieresolutie te bekomen, waardoor ze uiteindelijk relatief duur uitvallen. Een tweede vaststelling is dat de parametrische frequentieschattingmethododes over het algemeen een hogere nauwkeurigheid leveren ten koste van een hoge reken-

complexiteit.

In **Paragraaf 8.2** wordt de nieuwe methode afgeleid, op basis van het zogenaamde beperkte-polen-en-nullensignaalmodel dat in de literatuur werd voorgesteld in het kader van adaptieve inkepingfilteralgoritmen (ANF). In de voorgestelde beperkte-polen-en-nullen-lineairepredictiemethode (CPZLP) worden de parameters van dat signaalmodel geschat d.m.v. een numeriek optimalisatie-algoritme. Door het signaalmodel te ontbinden in een product van bikwadratische deelmodellen kan het optimalisatieprobleem ontkoppeld worden, waardoor elke sinusoidale frequentie apart kan worden geschat. In de optimalisatie wordt een iteratieve lijnzoekmethode gebruikt waarbij de zoekrichting wordt berekend d.m.v. een steilste-helling-aanpak, een Gauss-Newton-aanpak, of een quasi-Newton-aanpak.

**Paragraaf 8.3** handelt over de rekencomplexiteit van de voorgestelde CPZLP-methode. Ten gevolge van de ont koppeling van het optimalisatieprobleem is de rekencomplexiteit een lineaire functie van het aantal gebruikte signaalbeemonsteringen en van het aantal sinusoidale componenten in het geobserveerde signaal. Omwille van de lage complexiteit kunnen relatief lange signaalsegmenten worden gebruikt waardoor de ruisgevoeligheid gereduceerd wordt.

Tenslotte worden in **Paragraaf 8.4** simulatieresultaten getoond die de nauwkeurigheid van de CPZLP-methode illustreren. Wanneer een quasi-Newton-aanpak wordt gevolgd om de zoekrichting te berekenen, blijkt de resulterende frequentievariantie dicht bij de Cramér-Rao-ondergrens te liggen voor een signaal-ruisverhouding groter dan of gelijk aan 5 dB.

## Hoofdstuk 9: AFC voor spraaktoepassingen

Dit hoofdstuk is het eerste van drie hoofdstukken die handelen over adaptieve feedbackonderdrukking (AFC), een methode voor akoestische-feedbackbeheersing die in eerste instantie voor spraaktoepassingen wordt voorgesteld. In **Paragraaf 9.1** wordt het akoestische-feedbackprobleem besproken, alsook een overzicht van de voornaamste oplossingen uit de literatuur. In kamertoepassingen wordt voornamelijk gebruik gemaakt van inkepingfiltergebaseerde fluittoononderdrukkingmethoden (NHS), terwijl in hoorapparaten vaak de AFC-methode wordt toegepast. Die laatste methode levert doorgaans een hogere performantie, maar de algoritmen die werden afgeleid voor hoorapparaten kunnen niet als dusdanig worden gebruikt in kamertoepassingen.

In **Paragraaf 9.2** wordt een recente methode voor AFC in hoorapparaten besproken, die is gebaseerd op de predictiefoutmethode (PEM) voor gesloten-lussysteemidentificatie. Die zogenaamde PEM-AF-methode kan een foutloze schatting van het akoestische-feedbackpad aanleveren door tegelijkertijd een model van het bronsignaal te schatten. De methode is echter gebaseerd op de

veronderstelling dat het bronsignaal stationair is gedurende een tijdsinterval dat even lang is als de impulsresponsie van het akoestische-feedbackpad. Die veronderstelling is geldig voor de korte akoestische-feedbackpaden in hoorapparaten maar niet voor langere feedbackpaden zoals in kamertoepassingen.

In **Paragraaf 9.3** wordt een nieuw PEM-gebaseerd AFC-algoritme afgeleid waarin geen verband wordt verondersteld tussen de stationariteit van het bronsignaal en de lengte van de akoestische-feedbackpadimpulsresponsie. Het resulterende algoritme voert enkel rij-operaties uit op de luidsprekersignaalmatrix in het predictiefoutcriterium, en wordt daarom aangeduid met de naam PEM-AFROW.

Een tweede verbetering van het PEM-AFROW-algoritme t.o.v. het PEM-AF-algoritme wordt voorgesteld in **Paragraaf 9.4** en bestaat in het gebruik van een lange-termijnpredictiefilter om de periodiciteit in het bronsignaal te modelleren. Het gebruik van een dergelijk filter blijkt noodzakelijk te zijn wanneer de fundamentele periode van de stemhebbende spraakklanken in het bronsignaal korter is dan de akoestische-feedbackpadimpulsresponsie, wat doorgaans enkel het geval is in kamertoepassingen.

Het resulterende algoritme wordt samengevat in **Paragraaf 9.5** en de rekencomplexiteit alsook de keuze van de parameterwaarden wordt besproken in **Paragraaf 9.6**. Tenslotte worden in **Paragraaf 9.7** simulatieresultaten getoond waaruit blijkt dat voor lange akoestische-feedbackpaden een aanzienlijke verbetering van de performantie kan worden gerealiseerd met het PEM-AFROW-algoritme t.o.v. het NLMS- en het PEM-AF-algoritme.

## Hoofdstuk 10: Robuuste en efficiënte AFC-implementatie

In dit hoofdstuk komt de praktische implementatie aan bod van het PEM-AFROW-algoritme dat in **Hoofdstuk 9** werd voorgesteld voor AFC in spraaktoepassingen. In **Paragraaf 10.1** worden het akoestische-feedbackprobleem en de bestaande oplossingen toegelicht, waarna het concept en de werking van het PEM-AFROW-algoritme wordt uitgelegd. De nadruk ligt daarbij op het proactieve karakter van de AFC-methode in vergelijking met de reactieve werking van de nochtans populaire NHS-methode.

In **Paragraaf 10.2** worden zes technieken voorgesteld om de robuustheid van het PEM-AFROW-algoritme in een realistische situatie te verbeteren. De eerste techniek wordt aangeduid als adaptatiecontrole en bestaat in het afschakelen van de adaptatie gedurende tijdsintervallen waarin een lage feedback-tot-bronsignaalvermogenverhouding wordt gedetecteerd (i.e., bij spraakaanzetten in het bronsignaal). In de tweede techniek wordt een regularisatieterm op basis van a-priorikennis toegevoegd om de initiële convergentie van het adaptief filter te verbeteren. In de derde techniek wordt een tweelingfilterstructuur toe-

gepast waarbij het zogenaamde achtergrondfilter sneller wordt geadapteerd dan het voorgrondfilter, vanuit de bedoeling om een snelle convergentie te verzoenen met een kleine stationaire adaptatiefout. De vierde techniek vertrekt vanuit de idee om de voordelen van de AFC-methode en de NHS-methode te combineren in een hybride aanpak. Concreet worden inkepingsfilters geplaatst rond frequentiewaarden waarop het geslotenlussysteem zich dicht bij de stabiliteitsgrens bevindt, waardoor de robuustheid tegen abrupte veranderingen in het akoestische-feedbackpad toeneemt. De vijfde techniek bestaat in de toevoeging van een niet-lineair element in het elektro-akoestisch voorwaarts signaalpad om te voorkomen dat de luidspreker van het geluidsversterkingssysteem een niet-lineair gedrag vertoont. In de zesde techniek wordt het microfoonsignaal door een hoogdoorlaatfilter gestuurd om het verschil in afsnijfrequentie tussen de luidspreker en de microfoon te compenseren.

**Paragraaf 10.3** handelt over subband- en frequentiedomeinimplementaties van het PEM-AFROW-algoritme. In de subbandimplementatie worden zowel het microfoonsignaal als het luidsprekersignaal in subbanden gesplitst en worden de feedback-gecompenseerde subbandsignalen opnieuw samengevoegd in een discrete-Fouriertransformatiegemoduleerde perfectereconstructiefilterbank. Op die manier kan de adaptatie in het PEM-AFROW-algoritme in smallere frequentiebanden gebeuren, met bijgevolg een lagere adaptieve-filterorde en een lagere bemonsteringsfrequentie. In de frequentiedomeinimplementatie wordt gekozen voor een blokgebaseerde adaptatie a.h.v. het bestaande FDAF- of PBFDAF-algoritme.

Tenslotte worden in **Paragraaf 10.4** enkele van de voorgestelde technieken voor het verhogen van de robuustheid uitgetest d.m.v. computersimulaties en realistische experimenten. Uit experimenten in reële tijd blijkt dat voornamelijk de adaptatiecontrole, de tweelingfilterstructuur en het gebruik van een hoogdoorlaatfilter een positieve invloed hebben op de robuustheid van het PEM-AFROW-algoritme. Een toename van de stabiele versterking met 14 dB blijkt haalbaar te zijn in een sterk tijdsvariante akoestische omgeving.

## Hoofdstuk 11: Lineaire predictie van audiosignalen

De PEM-gebaseerde AFC-methode die werd voorgesteld en geëvalueerd in **Hoofdstuk 9** en **10** blijkt niet goed te presteren voor audiotoeepassingen omdat het autoregressief bronsignaalmodel niet geschikt is voor audiosignalen. In dit hoofdstuk wordt het resultaat van een onderzoek naar lineaire predictie (LP) van audiosignalen beschreven, als voorbereiding op de ontwikkeling van AFC-algoritmen voor audiotoeepassingen in **Hoofdstuk 12**.

In **Paragraaf 11.1** wordt de state of the art inzake de modellering en codering van audiosignalen beschreven, waarbij de nadruk ligt op LP-gebaseerde methodes. In een meer algemene context wordt de literatuur omtrent de invloed



van ruis op de modellering en identificatie van sinusoidale signalen bestudeerd. Tenslotte worden een aantal LP-modellen aangehaald die in het verleden werden voorgesteld als alternatief voor het autoregressief model.

In **Paragraaf 11.2** wordt het tonale signaalmodel voorgesteld waarop de analyse en de voorbeelden in de rest van het hoofdstuk gebaseerd zijn. Dat model kan zowel voor monofone als voor polyfone signalen gebruikt worden. Daarnaast wordt het concept van LP uitgelegd, met de nadruk op de interpretatie van LP als een spectrale-schattingsmethode. Er wordt een kleinste-kwadraten criterium voorgesteld van waaruit de performantie van de verschillende LP-modellen kan worden geanalyseerd.

In **Paragraaf 11.3** worden de eigenschappen en meer bepaald de tekortkomingen van het autoregressief LP-model geanalyseerd. De belangrijkste vaststelling is dat de aanwezigheid van de ruiscomponent in het tonale signaalmodel de positie van de polen in het geschatte LP-model op twee manieren beïnvloedt. Het eerste effect van de ruis werd voordien in de literatuur beschreven en bestaat in een verschuiving van de polen in de richting van de oorsprong van het complexe vlak, waardoor de magnituderesponsie van het LP-model wordt afgevlakt. Een tweede effect, dat nog niet eerder werd beschreven, bestaat in een rotatie van de polen die neigt naar een uniforme polenverdeling rondom de eenheidscirkel. Het gevolg daarvan is dat het autoregressief LP-model voornamelijk geschikt is voor de modellering van signalen waarvan de tonale componenten uniform verdeeld zijn in het Nyquistinterval, wat meestal niet het geval is voor audiosignalen.

Op basis van de voorgaande vaststelling worden in **Paragraaf 11.4** een aantal alternatieve LP-modellen voor audiosignalen voorgesteld waarvan de performantie niet of in mindere mate afhangt van de verdeling van de tonale componenten in het Nyquistinterval. Het beperkte-polen-en-nulmodel (PZLP) wordt geanalyseerd en blijkt ongevoelig te zijn voor de aanwezigheid van ruis in het tonale signaalmodel. Het hoge-orde autoregressief (HOLP) model kan beschouwd worden als een benadering van het PZLP-model aangezien dat laatste een oneindige impulsresponsie heeft. Het toonhoogtepredictiemodel (PLP) is enkel geschikt voor monofone signalen en vertoont een kamfiltergedrag waarbij de polen per definitie uniform verdeeld zijn rondom de eenheidscirkel. Het zogenaamde verdraaid LP-model (WLP) wordt bekomen door een bilineaire transformatie uit te voeren op het autoregressief LP-model en wordt gekenmerkt door een spectrale modellering volgens een niet-uniforme frequentieresolutie. Het selectief LP-model is een autoregressief model dat het geobserveerde signaal aan een lagere bemonsteringsfrequentie voorstelt waardoor enkel de laagfrequente tonale componenten worden gemodelleerd.

In **Paragraaf 11.5** worden de verschillende LP-modellen vergeleken op basis van uiteenlopende performantiecriteriën en zowel voor synthetische als echte monofone en polyfone audiosignalen. Over het algemeen wordt de beste per-

formantie bereikt met het HOLP-model, maar dat model heeft te veel vrije parameters om in de praktijk zinvol te kunnen worden aangewend. Het PZLP-model is een mogelijk alternatief voor het HOLP-model, maar de schatting van de PZLP-modelparameters is niet triviaal, in het bijzonder voor polyfone signalen. Het PLP-model presteert dan weer bijzonder goed voor monofone signalen, terwijl het WLP-model eerder voor polyfone signalen geschikt blijkt te zijn.

## Hoofdstuk 12: AFC voor auditoepassingen

In dit hoofdstuk wordt op basis van de PEM-gebaseerde AFC-aanpak van **Hoofdstuk 9** en de alternatieve LP-modellen van **Hoofdstuk 11** een aantal nieuwe AFC-algoritmen ontwikkeld die in het bijzonder geschikt zijn voor toepassingen waarin het bronsignaal een audiosignaal is.

In **Paragraaf 12.1** wordt een overzicht gegeven van de bestaande methodes voor akoestische-feedbackbeheersing. De meeste van die methodes blijken niet geschikt te zijn voor auditoepassingen, ofwel omdat ze een onvoldoende hoge geluidskwaliteit garanderen, ofwel omdat ze gebaseerd zijn op een signaalmodel waarin expliciet een spraaksignaal verondersteld wordt. De meest voor de hand liggende richting om een akoestische-feedbackbeheersingsmethode voor auditoepassingen te ontwikkelen is een AFC-aanpak waarbij de decorrelatie in het adaptieve-filtercircuit wordt uitgevoerd i.p.v. in de gesloten signaallus.

In **Paragraaf 12.2** wordt het akoestische-feedbackprobleem uiteengezet en wordt het AFC-concept geïntroduceerd. Vervolgens wordt in **Paragraaf 12.3** een PEM-gebaseerde AFC-aanpak voorgesteld waarbij het bronsignaal wordt voorgesteld d.m.v. een serieschakeling van twee LP-modellen: een alternatief LP-model voor de tonale componenten en een autoregressief LP-model voor de ruiscomponenten. Het alternatief LP-model kan een PZLP-, PLP-, WLP- of SLP-model zijn, overeenkomstig de resultaten van het onderzoek in **Hoofdstuk 11**.

In **Paragraaf 12.4** worden de identificeerbaarheidsvoorwaarden voor gelijktijdige identificatie van het akoestische-feedbackpad en de seriegeschakelde bronsignaalmodellen afgeleid. De identificeerbaarheid kan worden gegarandeerd op voorwaarde dat er vertragingselementen worden ingevoerd in het elektroakoestisch voorwaarts signaalpad en/of in het adaptieve-filtercircuit.

In **Paragraaf 12.5** worden de nieuwe AFC-algoritmen in detail voorgesteld. De volgorde van de verschillende signaalbewerkingen is daarbij van groot belang om te voorkomen dat de rekencomplexiteit de hoogte in gaat. Daarnaast wordt de keuze van verscheidene cruciale algoritmeparameters besproken en gemotiveerd. In **Paragraaf 12.6** wordt de complexiteit van de voorgestelde algoritmen vergeleken met die van de NLMS- en PEM-AFROW-algoritmen.

Bovendien wordt een aantal modelbenaderingen voorgesteld die gebaseerd zijn op veronderstellingen inzake de stationariteit van het akoestische-feedbackpad en het bronsignaal, waarmee het surplus in complexiteit t.o.v. het NLMS-algoritme kan worden beperkt tot 25 à 40 %, afhankelijk van de beoogde toepassing.

Tenslotte worden de nieuwe algoritmen in **Paragraaf 12.7** geëvalueerd voor zowel kamerakoestische als hoorapparatentoepassingen. De resultaten bevestigen enigszins de experimentele vaststellingen die in **Hoofdstuk 11** werden gemaakt. Het PLP-model blijkt vooral goed te scoren in het geval van monofone bronsignalen, terwijl het WLP-model eerder geschikt lijkt voor polyfone bronsignalen. Zowel voor monofone als voor polyfone signalen blijkt ook het PZLP-model in een goede AFC-performantie te resulteren. Het effect van de modelbenaderingen op het gedrag van de algoritmen blijkt overigens in de meeste gevallen verwaarloosbaar te zijn.

## **Hoofdstuk 13: Conclusies en suggesties voor toekomstig onderzoek**

In **Paragraaf 13.1** worden de voornaamste bijdragen in het doctoraatsproefschrift op een rijtje gezet. Een eerste grote bijdrage bestaat in de ontwikkeling van PEM-gebaseerde adaptieve-filteralgoritmen en de toepassing daarvan in verschillende open problemen in akoestische signaalverwerking. Een tweede grote bijdrage is de verwezenlijking van een ruim theoretisch kader waarin de techniek van regularisatie in lineaire adaptieve filtering kan worden geïnterpreteerd en verbeterd. Een derde grote bijdrage ligt in de beschrijving van een uitgebreid overzicht en een vergelijkende evaluatie van de state of the art in akoestische-feedbackbeheersing, aangezien rond dit onderwerp tot op heden geen overzichtsliteratuur of vergelijkende studie beschikbaar is. Naast die drie grote bijdragen bevat dit proefschrift nog drie kleinere bijdragen, meer bepaald met betrekking tot parametrische-egalisatiefilterontwerp, efficiënte parametrische frequentieschatting en lineaire predictie van audiosignalen.

Tenslotte worden in **Paragraaf 13.2** enkele suggesties voor toekomstig onderzoek besproken. De twee belangrijkste uitdagingen, die bovendien nauw met elkaar verbonden zijn, bestaan in de reductie van de nodige rekenkracht voor akoestische-echo-onderdrukking en adaptieve feedback-onderdrukking en in de uitbreiding van die twee methodes naar meerkanaalssystemen. De sleutel om die uitdagingen aan te pakken ligt vermoedelijk in het gebruik van andere modellen dan de impulsresponsie om de kamerakoestiek voor te stellen.



## Part I

# Introduction



# Chapter 1

## Introduction and Overview

This thesis addresses two topics in signal processing dealing with the enhancement of sound signals in room acoustics, namely *acoustic echo cancellation* and *acoustic feedback control*. In this introduction, we briefly set forth the concept of room acoustic signal enhancement, and subsequently provide an overview of open problems in the areas of acoustic echo cancellation and acoustic feedback control. In this overview, we concisely review the state of the art, referring to the publications collected in this thesis for a more thorough literature review of each of the topics covered. More specifically, a literature review on acoustic echo cancellation can be found in **Chapter 3**, while an extensive description of the state of the art in acoustic feedback control is provided in **Chapter 2**. In the introductory overview, we also explain how the work presented in the thesis fits into the state of the art, and point out the main contributions. The introduction ends with an overview of the publications included in the different chapters.

### 1.1 Room Acoustic Signal Enhancement

The importance of sound signals in human communication and interaction can hardly be overrated. Speech can be viewed as one of the fundamental communication tools between human beings, carrying not only a linguistic message but also an emotional expression [1]. Music has become a daily source of empathy and entertainment for many people, either through sound recordings or live performances. Furthermore, we are surrounded by other kinds of –more or less informative– sound signals, such as traffic noise, alarm signals, weather sounds, etc.

Sound quality is of major importance for being able to properly perceive and interpret sound signals. However, sound signals are often distorted on their way from the source (e.g., the speaker) to the receiver (e.g., the listener). We may think of car noise interfering with a speech conversation between car passengers, low-fidelity loudspeakers distorting audio reproduction in portable music players, speech echoes disrupting a mobile telephone conversation, etc. One particular source of distortion is due to room acoustics. When a sound is produced in a closed acoustic environment, it is partially reflected by the physical boundaries of the environment, i.e., by the walls, floor, and ceiling of the room. As a result, not only the source signal is received (denoted as the “direct” sound component), but also a multitude of delayed and attenuated replicas of the source signal (denoted as the “indirect” sound component). This effect is known as *reverberation*, and depending on the particular application, this is viewed either as a desired or undesired effect. In fact, both the artificial enhancement of the reverberation effect (e.g., for improving concert hall acoustics [2]-[4]) as well as the elimination of reverberation (i.e., dereverberation, e.g., for improving speech intelligibility [5]-[8]) have been active research topics in signal processing for many years. The aim of dereverberation is to retain the direct sound component while eliminating the indirect sound component. In a typical dereverberation scenario, the source signal is captured using one or more microphones that are positioned relatively far from the source. In this case, the source signal is not separately accessible as a reference signal, hence most dereverberation algorithms perform a blind identification of the room acoustics, operating only on the available microphone signals.

A related problem occurs when loudspeakers are used to reproduce speech or music signals in an acoustic environment, while at the same time microphones are present in the same environment to capture local sound signals. In such a scenario, it is usually undesired yet unavoidable that the loudspeaker sound is captured by the microphones, in addition to the local sound signals. This problem is known as the *acoustic echo* problem, and it differs from the dereverberation problem in two ways. First of all, both the direct and indirect sound component of the echo signal have to be eliminated from the microphone signals. Second, the loudspeaker signal is typically available as a reference signal, such that the room acoustics can be identified using non-blind system identification techniques. The acoustic echo problem becomes more complicated when the loudspeakers are used to reproduce the local sound signals captured by the microphones. Since in this case the echo signals are highly correlated with the local sound signals, constructive interference of these signals at the microphones may lead to oscillations that are perceived as ringing and howling effects. This problem is known as the *acoustic feedback* problem, and can be considered to be one of the most long-standing problems in acoustic signal processing. The acoustic echo and feedback problems have several implications on sound quality. While acoustic echoes may severely degrade speech intelligibility and disturb the normal course of speech conversations, acoustic feedback



may distort speech and audio signals through howling, ringing, echoes, and excessive reverberation.

A number of applications in which the acoustic echo and feedback problems are encountered, are the following [9],

- *Hands-free telephony* is probably the most exemplary application of acoustic echo control, in which a telephone conversation is performed with freestanding loudspeakers and microphones instead of the traditional handheld telephone devices. A similar yet more advanced application is the so-called *teleconferencing* application, which usually involves multiple microphones and loudspeakers and in which more than two participants may take part.
- *In-vehicle communications* is an emerging application for acoustic signal processing, that receives a lot of interest from the automotive industry. The aim is to improve the comfort of speech conversations as well as the quality of audio playback within the vehicle, despite the high levels of, e.g., road, wind, and traffic noise. Since an in-vehicle communications system includes loudspeakers that reproduce the sound signals captured by nearby microphones, acoustic feedback is usually unavoidable. In addition, acoustic echoes may result from audio playback being picked up by the microphones. Some example applications are in-car communications systems, cockpit communications systems in aircrafts, and on-board passenger information systems in trains.
- *Public address* systems represent a very challenging application of acoustic feedback control. Since extremely high model orders are required for modeling the acoustics of large venues, acoustic feedback control in public address applications is often restricted to howling suppression (without modeling the room acoustics). Sound reinforcement systems for public address are used in many different areas, e.g., sound reinforcement in concert halls and auditoria, information broadcasting in airports, train stations, and other public environments, and conference systems for large-scale meetings.
- *Hearing aids* have been one of the most widely studied applications of acoustic feedback control. Due to the closeness of hearing aid microphones and loudspeakers, these devices have been found to be very prone to acoustic feedback and howling. The success of acoustic feedback cancellation in hearing aids has paved the way for this technique being extrapolated to other applications, particularly room acoustic applications.

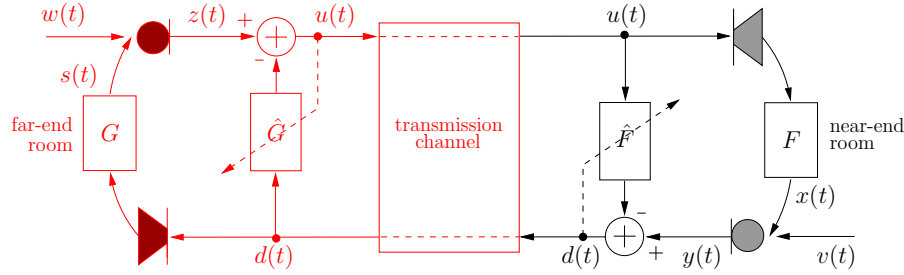


Figure 1.1: Acoustic echo cancellation in a hands-free telephony application.

## 1.2 Acoustic Echo Cancellation

### 1.2.1 Concept

A conceptual example of the acoustic echo cancellation (AEC) approach in a hands-free telephony application is shown in Fig. 1.1. We focus on the right-hand part of the figure, denoted as the “near-end room”. In this room, a microphone is present for capturing the speech signal  $v(t)$  of the so-called “near-end speaker”, which is to be sent over a transmission channel to the receiver in the far-end room. Similarly, the far-end speech is transmitted from the far-end room to the near-end room, and is reproduced by the loudspeaker in the near-end room. Due to the acoustic coupling between the loudspeaker and microphone in the near-end room, an echo  $x(t)$  of the far-end signal  $u(t)$ , but instead consists of a multitude of delayed and attenuated replicas of the far-end signal. This effect is due to the room reverberation, and substantially increases the complexity of the acoustic echo problem.

The AEC approach consists in modeling and identifying the acoustic feedback path  $F$ , subsequently filtering the far-end signal  $u(t)$  with the echo path estimate  $\hat{F}$ , and finally subtracting the predicted echo signal from the microphone signal  $y(t)$ . The echo-compensated signal  $d(t)$  obtained after the subtraction is then sent over the transmission channel to the far-end room. Since the acoustic echo path may vary with time, e.g., due to people moving around the room or changes in the microphone and loudspeaker positions, the echo path estimate  $\hat{F}$  should be regularly updated. For this reason, the identification of the acoustic echo path is preferably carried out using an adaptive filtering algorithm, which updates the previous echo path estimate according to the most recent far-end and microphone data. A similar process concurrently takes place at the far-end side to compensate for the acoustic echo path in the far-end room, see Fig. 1.1.

Despite the fact that the AEC approach is conceptually simple and that an

astonishing number of research results on AEC has been published (see [10]-[13] and references therein), several issues still remain unsolved. Three of these open issues are treated in this thesis and will be introduced below, i.e., double-talk robustness, acoustic echo path undermodeling, and poor excitation.

### 1.2.2 Double-Talk Robustness

It is well known that least-squares(LS)-based adaptive filtering algorithms which are traditionally used in AEC, behave in a non-robust way in a double-talk situation. Double-talk is defined as the situation in which the near-end speaker and the far-end speaker are concurrently active, and has been shown to occur during approximately 20 % of the time in a normal telephone conversation [14]. It has been observed that during double-talk, the convergence speed of the recursive least squares (RLS) adaptive filtering algorithm is drastically reduced, while underdetermined RLS algorithms such as the normalized least mean squares (NLMS) and affine projection algorithm (APA) may even diverge. For this reason, the state-of-the-art acoustic echo canceller is usually equipped with a double-talk detector (DTD) that freezes the adaptation during double-talk periods. However, there are several reasons why the use of a DTD is discouraged. First, due to the detection lag and other DTD imperfections, the adaptive filter coefficients are often seen to diverge even before the DTD freezes the adaptation. Second, the AEC is not able to track echo path changes that occur during double-talk, hence a large residual echo may result from such a situation. Finally, if the AEC is applied in a noisy environment, there is a continuous near-end activity so that the use of a DTD becomes futile. For these reasons, there is a trend towards the development of double-talk-robust adaptive filtering algorithms, that can either cope with an imperfect DTD behavior or even operate without a DTD. Referring to **Chapter 3** for a detailed literature review, we should mention that the most widely accepted double-talk-robust AEC solution consists in introducing a scaled nonlinearity in the AEC criterion [11, Ch. 6],[15]. This solution still requires the use of a DTD, and results in an improved robustness at the onset of double-talk periods, at the expense of a decreased convergence rate during single-talk periods (i.e., when the near-end speaker is not active).

Our contribution to the double-talk robustness problem consists in the development of a family of double-talk-robust adaptive filtering algorithms that are designed to operate without a DTD, and deliver a convergence improvement during double-talk without sacrificing convergence speed during single-talk periods, see **Chapter 3**. This result is obtained by searching for the optimal AEC solution in a minimum-variance linear estimation framework, rather than in a traditional LS framework. It can be understood that the variance of the LS echo path estimate during double-talk increases with the degree of coloration in the near-end signal. The minimum-variance echo path estimate (also known as the “best linear unbiased estimate”) thus depends on the near-end signal

characteristics, which are in practice unknown and time-varying. The proposed algorithms aim to whiten the near-end signal component in the microphone signal by using adaptive decorrelation filters that are estimated concurrently with the acoustic echo path using a prediction-error-method (PEM)-based approach [16, Ch. 3],[17, Ch. 7]. It appears that, particularly in the underdetermined case, the adaptive filter convergence can be significantly improved at the cost of a moderate increase in computational complexity.

### 1.2.3 Acoustic Echo Path Undermodeling

While the acoustic echo path effectively has an infinite impulse response (IIR), it is usually modeled as a finite impulse response (FIR) filter, since adaptive FIR filtering techniques are generally recommended over adaptive IIR filtering techniques due to their guaranteed stability and global convergence. As a consequence, the adaptive filter undermodels the acoustic echo path, hence the echo path estimate may be biased and have a high variance. The undermodeling issue is often bypassed in the AEC literature by performing computer simulations using a truncated acoustic echo path together with a sufficient-order adaptive filter. Only a few methods have been proposed that attempt to improve the adaptive filter behavior in the undermodeled AEC situation, which are based on appropriately updating the NLMS step size parameter [18],[19].

Our contribution to the problem of undermodeled acoustic echo path estimation is described in **Chapter 4**. Due to the echo path undermodeling, part of the echo signal captured by the microphone cannot be predicted by the AEC. This unpredictable echo signal can be decomposed into two components: a component that lies in the column space of the far-end signal Hankel matrix, and a component that is orthogonal to this column space. The former component results in a bias in the resulting echo path estimate, while the latter component increases the variance of the echo path estimate. Similarly to the double-talk-robust AEC algorithms proposed in **Chapter 3**, adaptive decorrelation filters can be used to (partially) whiten the unpredictable echo signal. The resulting effect is that the variance of the echo path estimate is heavily reduced, while the bias is concentrated in only a few adaptive filter taps which can easily be discarded from the echo path estimate. This approach is implemented using one of the PEM-based algorithms described in **Chapter 3**, which does not require significantly more computations than the traditional adaptive filtering algorithms.

### 1.2.4 Poor Excitation and Regularization

An important result in system identification states that the identifiability of the acoustic echo path using an echo path model of order  $n_F$  is guaranteed only if the far-end signal is persistently exciting of order  $n_F$  [17, Ch. 14]. This

essentially means that the  $n_F \times n_F$  far-end signal autocorrelation matrix should be nonsingular. Since  $n_F$  may be very large in room acoustic applications, this condition is not always satisfied, and if it is, the far-end signal autocorrelation matrix often turns out to be ill-conditioned. This may lead to a high variance of the echo path estimate, and moreover, numerical problems may further reduce the AEC performance. The poor excitation problem is traditionally approached in a pragmatic way, by either freezing the adaptation if the far-end signal power decreases below a certain threshold or by adding a scaled identity matrix to the far-end signal autocorrelation matrix before it is inverted in the LS estimation. The latter approach is known as regularization, and is generally considered to offer a powerful solution to the poor excitation problem. In AEC, the regularization parameter (i.e., the identity matrix scaling factor) is usually chosen to be an a priori estimate of the long-term near-end signal power, as suggested in [20].

A first contribution in the context of poor excitation and regularization is described in **Chapter 5**. Instead of applying regularization as an ad hoc method for improving the conditioning of the far-end signal autocorrelation matrix, we attempt at finding an optimal approach to regularization in which we also allow the regularization matrix to be different from the identity matrix. This is possible by estimating the echo path parameters in a Bayesian minimum mean square error (MMSE) framework instead of in an LS framework, hence creating the opportunity to take into account prior knowledge that may be available on the echo path parameters. This approach leads to a general framework for regularization in linear adaptive filtering, in which the existing scaled-identity-matrix-based Tikhonov and Levenberg-Marquardt regularization methods appear to be special cases that do not fully exploit the available prior knowledge. Interestingly enough, the optimally regularized APA and NLMS algorithms exhibit features known as leakage and proportionate adaptation, which have not previously been related to regularization. In particular, the proposed regularization framework has led to a new interpretation of the proportionate APA [21] and proportionate NLMS [22],[23] algorithms that have recently gained a lot of interest.

A second contribution relating to regularization is described in **Chapter 6**, and consists in the development of a so-called “dual regularization” approach that is aimed at improving the numerical properties of the double-talk-robust PEM-based AEC algorithms proposed in **Chapter 3**. It has been found that some of the PEM-based algorithms are prone to numerical problems due to a scaling ambiguity that is inherent in the PEM-based approach. The proposed solution is based on the extension of the Bayesian MMSE criterion obtained in **Chapter 5** with an additional term that regularizes the estimation of the adaptive decorrelation filters in the PEM-based algorithms. This dual regularization term may additionally be used to incorporate prior knowledge on the near-end signal, e.g., using a model of long-term speech characteristics.

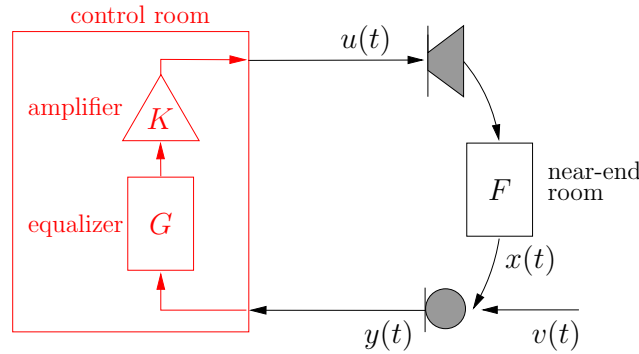


Figure 1.2: The acoustic feedback problem in a sound reinforcement system.

## 1.3 Acoustic Feedback Control

### 1.3.1 Concept

The acoustic feedback problem is depicted in Fig. 1.2 for a sound reinforcement system having one microphone and one loudspeaker. The similarity between the acoustic feedback problem and the acoustic echo problem is apparent from comparing Figs. 1.1 and 1.2. However, the crucial difference is that the transmission channel and far-end side have been replaced with a control room, in which the microphone signal is processed (e.g., equalized and amplified) before being reproduced by the loudspeaker in the near-end room. The sound reinforcement system in Fig. 1.2 thus comprises a closed signal loop, which may result in an unstable behavior that is observed through audible ringing and howling effects.

Similarly to the acoustic echo problem, the acoustic feedback problem has been the subject of extensive research in acoustic signal processing. However, the available research results have hardly been compared and the different approaches to acoustic feedback control have not yet been synthesized in a unifying framework (which contrasts with the availability of survey literature on acoustic echo control [10]-[13]). Consequently, while the AEC approach is generally accepted to be the prototype solution to the acoustic echo problem, there is no such consensus on how the acoustic feedback problem should be tackled. A first contribution of this thesis in the framework of acoustic feedback control, consists in providing a survey report that covers the relevant literature produced during the past five decades, see **Chapter 2**. In this report, we unambiguously define the acoustic feedback problem and provide a comprehensive literature review of existing acoustic feedback control methods (as proposed in research reports and patents), with an emphasis on public address and hands-free communication applications. Moreover, an in-depth

treatment of the three most widely used acoustic feedback control methods, i.e., phase-modulating feedback control (PFC), notch-filter-based howling suppression (NHS), and adaptive feedback cancellation (AFC), is given, thereby discussing conceptual as well as realization issues. Finally, a comparative evaluation of these three methods is performed and a number of future challenges in acoustic feedback control research are formulated.

Additionally, several contributions to the improvement of NHS and AFC methods for acoustic feedback control are described in this thesis, which are briefly outlined below.

### 1.3.2 Notch-Filter-Based Howling Suppression

Notch-filter-based howling suppression (NHS) is a widely used technique for acoustic feedback control, that is aimed at removing the ringing and howling artifacts resulting from acoustic feedback, thereby stabilizing the sound reinforcement system. A prototype NHS method essentially consists of the combination of a howling detection method and a notch filter design method. Howling detection and notch filter design can either be performed jointly or separately, resulting in one-stage and two-stage NHS methods, respectively.

#### Howling Detection

The NHS notch filters are activated only when an unstable behavior or a tendency towards instability is detected in the sound reinforcement system. Most of the available NHS howling detection methods are reactive, in the sense that howling can usually be perceived before being detected. In these methods, howling detection is typically based on a combined spectral and temporal analysis of the microphone signal. In one-stage NHS methods, the spectral analysis is achieved by identifying a suitably parametrized adaptive notch filter (ANF), which is subsequently used for howling suppression. In two-stage NHS methods, the spectral analysis is most often performed using a non-parametric estimate of the microphone signal spectrum, such as the short-time discrete Fourier transform (DFT), from which the appropriate howling detection features are then identified.

Our first contribution to NHS howling detection consists in the development of a so-called “proactive” howling detection technique for two-stage NHS methods, see **Chapter 10**. In contrast to the state-of-the-art howling detection methods, the proactive method identifies critical frequencies of the sound reinforcement system (i.e., frequencies at which the system is close to instability) before howling effectively occurs. The system can hence be stabilized by activating notch filters at the identified critical frequencies, thereby avoiding ringing and howling artifacts and hence retaining sound quality. This method is based on an estimate of the acoustic feedback path frequency response, which is regularly updated using an AFC algorithm.

A second contribution is related to one-stage NHS methods and is described in **Chapter 8**. Both FIR and IIR notch filters have been used in state-of-the-art one-stage NHS methods, the parameters of which are typically identified using an ANF algorithm. While the IIR-ANF approach has the advantage of being capable of producing extremely narrowband notch filters, its main difficulty lies in the fact that the LS criterion associated with the IIR-ANF frequency estimation problem can be highly non-convex. As a consequence, the performance of most IIR-ANF algorithms relies heavily on the initial conditions, and finding the global IIR-ANF solution appears to be hard to achieve. In **Chapter 8**, we propose a novel parametric frequency estimation algorithm that is based on a constrained IIR model similar to the model used in many IIR-ANF algorithms [24]. In contrast to the ANF approach, the proposed constrained pole-zero linear prediction (CPZLP) method is non-adaptive, but instead operates on a batch of available data. A key difference between the CPZLP and ANF approaches, is that the gradient information used in the CPZLP approach is recalculated using the entire data record in each iteration, thereby reducing the sensitivity to initial conditions and hence increasing the probability of global convergence. Unlike other non-adaptive parametric frequency estimation methods (e.g., maximum-likelihood [25] or subspace methods [26]), the CPZLP method has a computational complexity that depends only linearly on the size of the data record.

### Notch Filter Design

In two-stage NHS methods, biquadratic (i.e., second-order IIR) notch filters are typically used for howling suppression. These filters are usually designed using a procedure for the design of biquadratic parametric equalizer filters, of which the biquadratic notch filter is a special case. Nearly all existing design procedures start from the design of an analog parametric equalizer filter, followed by a bilinear transform that maps the analog frequency axis onto the digital frequency axis, see, e.g., [27]. To this end, the digital design variables should be “prewarped” to analog variables, which can however not be done in an exact way. Alternatively, biquadratic parametric equalizer filters can be designed directly in the digital domain, either by bilinearly transforming a  $\pi/2$ -centered filter into an arbitrarily centered filter [28], or by a design procedure known as pole-zero placement.

Our contribution to notch filter design consists in the development of a pole-zero placement technique for designing biquadratic parametric equalizer filters, see **Chapter 7**. The main strength of the pole-zero placement technique is its intuitiveness. However, existing pole-zero placement design procedures are based on certain approximations for which a quantitative foundation is lacking. In **Chapter 7**, we provide a quantitative treatment of the pole-zero placement design procedure, resulting in a set of novel design equations that are directly applicable to the NHS notch filter design problem.



### 1.3.3 Adaptive Feedback Cancellation

Adaptive feedback cancellation (AFC) is a promising approach to acoustic feedback control, which is conceptually similar to the AEC approach for acoustic echo control. However, state-of-the-art AEC algorithms cannot be applied as such for acoustic feedback control due to the existence of a closed signal loop. A first difficulty arising from the closed-loop situation, is that the system continuously operates in a double-talk situation. A second and more fundamental problem is that the closed signal loop results in a considerable correlation between the loudspeaker and near-end signals, and as a consequence, traditional LS-based adaptive filtering algorithms converge to a biased estimate of the acoustic feedback path. While the double-talk problem has been disregarded in most state-of-the-art AFC solutions, the correlation problem has been partially solved by inserting a decorrelating signal operation in the closed signal loop, e.g., a noise injection [29] or frequency shifting [30]. However, these decorrelation techniques provide a bias reduction rather than a complete bias removal, and moreover result in a considerable distortion of the loudspeaker signal. As a consequence, the sound quality provided by these state-of-the-art AFC solutions is relatively poor.

A first contribution in the context of AFC consists in the development of a novel AFC algorithm for speech applications, see **Chapter 9**. Inspired by recent AFC research in hearing aid applications [31],[32], we propose a PEM-based AFC approach in which both the double-talk problem and the correlation problem are tackled. The resulting AFC algorithm is similar to the double-talk-robust AEC algorithms proposed in **Chapter 3**, and does not require the insertion of a decorrelating signal operation in the closed signal loop. Decorrelation is instead performed in the adaptive filtering circuit, using adaptive decorrelation filters that whiten both the formant and pitch components in the near-end speech signal. The so-called “PEM-AFROW” algorithm differs from the PEM-based AFC algorithms proposed for hearing aids [32], in that no modeling approximations are applied when calculating the output signals of the adaptive decorrelation filters, which appears to be an essential condition in room acoustic speech applications.

A second contribution covers several issues related to the practical implementation of AFC algorithms, see **Chapter 10**. The motivation for this work has been to facilitate a real-time implementation of the PEM-AFROW algorithm that can be used for acoustic feedback control in real-life speech applications. To this end, the robustness and efficiency of the PEM-AFROW algorithm as proposed in **Chapter 9** are considerably improved. In terms of robustness, we propose the following improvements: an adaptation control strategy for dealing with convergence problems at near-end speech onsets, the use of a foreground/background twin adaptive filter structure (known from AEC [33]) for further improving the double-talk robustness, the insertion of a limiter in the

closed signal loop for modeling the nonlinear amplifier and loudspeaker behavior, and a regularization technique incorporating prior knowledge on the acoustic feedback path (similar to the regularization approach described in **Chapter 5**). Additionally, we propose a hybrid approach to acoustic feedback control by combining the use of the PEM-AFROW algorithm with an NHS method that features a proactive howling detection method. Finally, in terms of efficiency, subband and frequency domain implementations of the PEM-AFROW algorithm are proposed with the aim of achieving a real-time operation.

A final contribution consists in the development of AFC algorithms for audio applications, see **Chapter 11** and **12**. The PEM-AFROW algorithm developed in **Chapter 9** is particularly suited for speech applications, as it relies on the efficient all-pole modeling of speech signals. However, the use of all-pole models for audio signal modeling is not widespread. This is somewhat surprising, as it is well known that a signal consisting of a sum of  $N$  sinusoids can be perfectly modeled using a  $2N$ th order all-pole model [34]. In **Chapter 11**, we point out that the main cause for all-pole models to perform poorly in audio applications is related to the fact that the dominating tonal components in an audio signal lie in a frequency range that is much smaller than the Nyquist interval. We also review some alternative linear prediction models that do not suffer from the aforementioned problem, namely pole-zero models, high-order all-pole models, pitch prediction models, frequency-warped all-pole models, and selective all-pole models. These models are subsequently used to derive a set of PEM-based AFC algorithms in **Chapter 12**, which are better suited for AFC in audio applications. It appears that in particular the use of a pole-zero model or pitch prediction model, cascaded with a conventional all-pole model, is beneficial to the AFC performance. The estimation of the pole-zero model coefficients in the PEM-based AFC algorithm can be performed using the CPZLP algorithm proposed in **Chapter 8**. By introducing some useful model approximations related to the stationarity of the acoustic feedback path and the near-end signal, the computational overhead of the PEM-based AFC algorithms can be reduced to 25% of the NLMS complexity without significantly influencing the AFC performance.

## 1.4 Chapters and Publications Overview

The following publications are included in this thesis:

### Part I: Introduction

#### Chapter 2: State of the Art in Acoustic Feedback Control

T. van Waterschoot and M. Moonen, “50 years of acoustic feedback control: state of the art and future challenges,” submitted for publication in *Proc. IEEE*, Feb. 2009.

**Part II: Acoustic Echo Cancellation****Chapter 3: Double-Talk-Robust AEC**

T. van Waterschoot, G. Rombouts, P. Verhoeve, and M. Moonen, “Double-talk-robust prediction error identification algorithms for acoustic echo cancellation,” *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 846–858, Mar. 2007.

**Chapter 4: Acoustic Echo Path Undermodeling**

G. Rombouts, T. van Waterschoot, K. Struyve, P. Verhoeve, and M. Moonen, “Identification of undermodeled room impulse responses,” in *Proc. 2005 Int. Workshop Acoustic Echo Noise Control (IWAENC '05)*, Eindhoven, The Netherlands, Sep. 2005, pp. 153–156.

**Chapter 5: Optimal Regularization for AEC/AFC**

T. van Waterschoot, G. Rombouts, and M. Moonen, “Optimally regularized adaptive filtering algorithms for room acoustic signal enhancement,” *Signal Processing*, vol. 88, no. 3, pp. 594–611, Mar. 2008.

**Chapter 6: Dual Regularization for AEC/AFC**

T. van Waterschoot, G. Rombouts, and M. Moonen, “Dually regularized recursive prediction error identification for acoustic feedback and echo cancellation,” in *Proc. 15th European Signal Process. Conf. (EUSIPCO '07)*, Poznań, Poland, Sep. 2007, pp. 1610–1614.

**Part III: Acoustic Feedback Control****Chapter 7: Biquadratic Notch Filter Design**

T. van Waterschoot and M. Moonen, “A pole-zero placement technique for designing second-order IIR parametric equalizer filters,” *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 8, pp. 2561–2565, Nov. 2007.

**Chapter 8: Efficient Parametric Frequency Estimation**

T. van Waterschoot and M. Moonen, “Constrained pole-zero linear prediction: an efficient and near-optimal method for multi-tone frequency estimation,” in *Proc. 16th European Signal Process. Conf. (EUSIPCO '08)*, Lausanne, Switzerland, Aug. 2008.

**Chapter 9: AFC for Speech Applications**

G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, “Acoustic feedback suppression for long acoustic paths using a nonstationary source model,” *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.

**Chapter 10: Robust and Efficient AFC Implementation**

G. Rombouts, T. van Waterschoot, and M. Moonen, “Robust and efficient implementation of the PEM-AFROW algorithm for acoustic feedback cancellation,” *J. Audio Eng. Soc.*, vol. 55, no. 11, pp. 955–966, Nov. 2007.

**Chapter 11: Linear Prediction of Audio Signals**

T. van Waterschoot and M. Moonen, “Comparison of linear prediction models for audio signals,” *EURASIP J. Audio, Speech, Music Process.*, vol. 2008, Article ID 706935, 24 pages, 2008. doi:10.1155/2008/706935

**Chapter 12: AFC for Audio Applications**

T. van Waterschoot and M. Moonen, “Adaptive feedback cancellation for audio applications,” conditionally accepted for publication in *Signal Processing*, Mar. 2009.

**Bibliography**

- [1] K. R. Scherer, “Vocal communication of emotion: a review of research paradigms,” *Speech Commun.*, vol. 40, no. 1–2, pp. 227–256, Apr. 2003.
- [2] R. W. Guelke and A. D. Broadhurst, “Reverberation time control by direct feedback,” *Acustica*, vol. 24, pp. 33–41, 1971.
- [3] D. Griesinger, “Improving room acoustics through time-variant synthetic reverberation,” in *Preprints AES 90th Convention*, Paris, France, Feb. 1991, AES Preprint 3014.
- [4] T. Lokki and J. Hiipakka, “A time-variant reverberation algorithm for reverberation enhancement systems,” in *Proc. COST G-6 Conf. Digital Audio Effects (DAFX '01)*, Limerick, Ireland, Dec. 2001, pp. 28–33.
- [5] M. Miyoshi and Y. Kaneda, “Inverse filtering of room acoustics,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-36, no. 2, pp. 145–152, Feb. 1988.
- [6] S. Gannot and M. Moonen, “Subspace methods for multimicrophone speech dereverberation,” *EURASIP Journal Appl. Signal Process.*, vol. 2003, no. 11, pp. 1074–1090, 2003, special Issue on Signal Processing for Acoustic Communication Systems.
- [7] M. Delcroix, T. Hikichi, and M. Miyoshi, “Precise dereverberation using multichannel linear prediction,” *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 2, pp. 430–440, Feb. 2007.
- [8] E. A. P. Habets, S. Gannot, I. Cohen, and P. C. W. Sommen, “Joint dereverberation and residual echo suppression of speech signals in noisy

- environments,” *IEEE Trans. Audio Speech Lang. Process.*, vol. 16, no. 8, pp. 1433–1451, Nov. 2008.
- [9] G. Schmidt, “Applications of acoustic echo control – an overview,” in *Proc. 12th European Signal Process. Conf. (EUSIPCO '04)*, Vienna, Austria, Sep. 2004, pp. 9–16.
- [10] C. Breining, P. Dreiseitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, and J. Tilp, “Acoustic echo control. an application of very-high-order adaptive filters,” *IEEE Signal Process. Mag.*, vol. 16, no. 4, pp. 42–69, Jul. 1999.
- [11] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*. Berlin: Springer-Verlag, 2001.
- [12] E. Hänsler and G. Schmidt, *Acoustic echo and noise control: a practical approach*. New York: Wiley-IEEE, 2004.
- [13] E. Hänsler and G. Schmidt, Eds., *Topics in Acoustic Echo and Noise Control. Selected Methods for the Cancellation of Acoustical Echoes, the Reduction of Background Noise, and Speech Processing*. Berlin: Springer-Verlag, 2006.
- [14] M. M. Sondhi and D. A. Berkley, “Silencing echoes on the telephone network,” *Proc. IEEE*, vol. 68, pp. 948–963, Aug. 1980.
- [15] H. Buchner, J. Benesty, T. Gänsler, and W. Kellermann, “An outlier-robust extended multidelay filter with application to acoustic echo cancellation,” in *Proc. 2003 Int. Workshop Acoustic Echo Noise Control (IWAENC '03)*, Kyoto, Japan, Sep. 2003, pp. 19–22.
- [16] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. Cambridge, Massachusetts: MIT Press, 1986.
- [17] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
- [18] R. D. Poltmann, “Stochastic gradient algorithm for system identification using adaptive FIR-filters with too low number of coefficients,” *IEEE Trans. Circuits Syst.*, vol. 35, no. 2, pp. 247–250, Feb. 1988.
- [19] C. Paleologu, S. Ciochina, and J. Benesty, “Variable step-size NLMS algorithm for under-modeling acoustic echo cancellation,” *IEEE Signal Process. Lett.*, vol. 15, pp. 5–8, 2008.
- [20] S. L. Gay, “Fast projection algorithms with application to voice echo cancellation,” Ph.D. dissertation, Rutgers, The State University of New Jersey, New Brunswick, New Jersey, USA, Oct. 1994.

- [21] T. Gänsler, J. Benesty, S. L. Gay, and M. M. Sondhi, "A robust proportionate affine projection algorithm for network echo cancellation," in *Proc. 2000 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '00)*, vol. 2, Istanbul, Turkey, Jun. 2000, pp. 793–796.
- [22] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 5, pp. 508–518, Sep. 2000.
- [23] Z. Chen, S. Haykin, and S. L. Gay, "Proportionate adaptation: New paradigms in adaptive filtering," in *Advances in LMS filters*, S. Haykin and B. Widrow, Eds. New York: Wiley, 2003, ch. 8.
- [24] A. Nehorai, "A minimal parameter adaptive notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 4, pp. 983–996, Aug. 1985.
- [25] P. Stoica, R. L. Moses, B. Friedlander, and T. Söderström, "Maximum likelihood estimation of the parameters of multiple sinusoids from noisy measurements," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-37, no. 3, pp. 378–392, Mar. 1989.
- [26] P. Stoica and T. Söderström, "Statistical analysis of MUSIC and subspace rotation estimates of sinusoidal frequencies," *IEEE Trans. Signal Process.*, vol. 39, no. 8, pp. 1836–1847, Aug. 1991.
- [27] P. A. Regalia and S. K. Mitra, "Tunable digital frequency response equalization filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 1, pp. 118–120, Jan. 1987.
- [28] J. A. Moorer, "The manifold joys of conformal mapping: Applications of digital filtering in the studio," *J. Audio Eng. Soc.*, vol. 31, no. 11, pp. 826–841, Nov. 1983.
- [29] A. Goertz, "An adaptive subtraction filter for feedback cancellation in public address sound systems," in *Proc. 15th Int. Congr. Acoust. (ICA '95)*, Trondheim, Norway, Jun. 1995, pp. 69–72.
- [30] S. Kamerling, K. Janse, and F. van der Meulen, "A new way of acoustic feedback suppression," in *Preprints AES 104th Convention*, Amsterdam, The Netherlands, May 1998, AES Preprint 4735.
- [31] J. Hellgren, "Analysis of feedback cancellation in hearing aids with Filtered-X LMS and the direct method of closed-loop identification," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 2, pp. 119–131, Feb. 2002.
- [32] A. Spriet, I. Proudler, M. Moonen, and J. Wouters, "Adaptive feedback cancellation in hearing aids with linear prediction of the desired signal," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3749–3763, Oct. 2005.

- [33] K. Ochiai, T. Araseki, and T. Ogihara, "Echo canceler with two echo path models," *IEEE Trans. Commun.*, vol. 25, no. 6, pp. 589–595, Jun. 1977.
- [34] Y. T. Chan, J. M. M. Lavoie, and J. B. Plant, "A parameter estimation approach to estimation of frequencies of sinusoids," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 2, pp. 214–219, Apr. 1981.





## Chapter 2

# State of the Art in Acoustic Feedback Control

50 years of acoustic feedback control: state of the  
art and future challenges

Toon van Waterschoot and Marc Moonen

Submitted for publication in *Proc. IEEE*, Feb. 2009.

### **Contributions of first author**

- co-development of acoustic feedback problem statement
- literature review
- co-analysis of three main acoustic feedback control methods (PFC, NHS, and AFC)
- co-design of evaluation experiments
- software implementation and computer simulations
- co-interpretation of simulation results
- co-formulation of conclusion and future challenges
- text redaction and editing

## Abstract

The acoustic feedback problem has intrigued researchers over the past five decades, and a multitude of solutions has been proposed. In this survey paper, we aim to provide an overview of the state of the art in acoustic feedback control, to report results of a comparative evaluation with a selection of existing methods, and to cast a glance at the challenges for future research.

## 2.1 Introduction

Ever since sound reinforcement and public address (PA) systems have been in use, their performance has been troubled by the problem of acoustic feedback. Whenever a microphone captures a desired sound signal which is then processed (e.g., amplified) and played back by a loudspeaker in the same environment, as it is the case in a PA system, the loudspeaker signal is unavoidably fed back into the microphone. In this way, a closed signal loop is created which affects the system performance, deteriorating the sound quality and limiting the achievable amplification. Among the different artifacts that are produced by this acoustic coupling between loudspeaker and microphone, the howling effect is without any doubt the most characteristic one.

Historically, some ambiguity has arisen in the terminology associated with the acoustic feedback problem. The term *acoustic feedback* has been used to refer to the undesired acoustic coupling between a loudspeaker and a microphone as well as to the howling effect that results from the coupling. We will use the term acoustic feedback in the first sense. Both the acoustic coupling and the howling effect are sometimes also referred to as the *Larsen effect*, after the Danish physicist Søren Larsen, who is said to have been one of the first researchers to investigate the acoustic feedback problem [1].

*Acoustic feedback control* refers to the process of attempting to solve the acoustic feedback problem either completely (i.e., to remove the acoustic coupling) or partially (e.g., to remove the howling artifacts from the loudspeaker signal). This paper only deals with *automatic* methods for acoustic feedback control, i.e., methods that do not require the interaction of an operator. Moreover, the emphasis will be on discrete-time methods that can be implemented on a digital signal processor. Surprisingly enough, despite 50 years of research on automatic acoustic feedback control, many PA system technicians still prefer to prosecute *manual* control of acoustic feedback. The main reason for this is the lack of reliability in the available automatic acoustic feedback control solutions, i.e., howling may still occur and may even take more time to be eliminated than in case of manual control.

The goal of this paper is threefold. First of all, we aim to provide an overview

of the state of the art in acoustic feedback control, by reviewing relevant scientific papers and patents that have been published over the past five decades. The focus of the literature review is on acoustic feedback control for room acoustic sound reinforcement systems, a PA system being the most exemplary application. We should point out, however, that during the past two decades a considerable amount of research results has been published related to feedback control in hearing aids (HA), where the feedback is due to a combined acoustic and mechanical coupling. Some of these publications will also be referenced here as they have provided solutions that have successfully been extrapolated to room acoustic applications. Our second goal is to report results of a comparative evaluation of the three most widely used methods for acoustic feedback control, namely phase-modulating feedback control (PFC), notch-filter-based howling suppression (NHS), and adaptive feedback cancellation (AFC). Such a comparative evaluation has to our knowledge not been reported earlier. This is presumably due to the fact that these three methods in fact attempt to solve different problems (i.e., smoothing the system loop gain vs. howling suppression vs. removal of the acoustic loudspeaker-microphone coupling) and hence different measures have previously been used to quantify the performance of each of these methods individually. We observe, however, that even though the PFC, NHS, and AFC problem formulations are different, the ultimate objectives of these methods are the same: to improve sound quality, to increase the amount of achievable amplification, and to operate in a reliable way. Hence the reported comparative evaluation is carried out with these three objectives in mind. Finally, our third goal is to formulate the challenges that we believe are most prevalent to steer future research in acoustic feedback control towards the development of reliable and affordable solutions.

The paper is organized as follows. In Section 2.2, the acoustic feedback problem is formulated in a formal way, where the key result is the Nyquist stability criterion, based on which all the acoustic feedback control methods can be derived. Section 2.3 deals with the state of the art in acoustic feedback control: the existing feedback control solutions are divided into four categories (phase modulation methods, gain reduction methods, spatial filtering methods, and room modeling methods), and a detailed literature review of each category is given. In the next Sections, the three most popular acoustic feedback control methods are outlined in more detail: Section 2.4 deals with the PFC method, Section 2.5 with the NHS method, and Section 2.6 with the AFC method. In Section 2.7, these three methods are then evaluated in terms of the resulting sound quality, the achievable amplification, as well as their reliability. The evaluation is based on computer simulation results using realistic room acoustic models and for both speech and audio signals. Finally, in the concluding Section 2.8 we summarize the results obtained with the state-of-the-art methods and formulate future research challenges in acoustic feedback control.

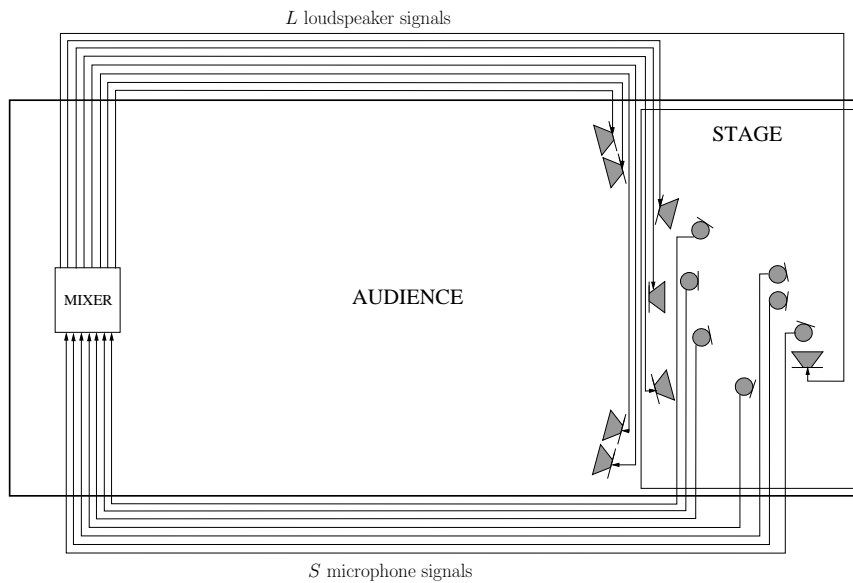


Figure 2.1: A typical public address (PA) system scenario, featuring 7 microphones, 4 on-stage loudspeakers, 4 loudspeakers directed towards the audience, and a mixing/signal processing/amplification console.

## 2.2 The Acoustic Feedback Problem

A typical PA system scenario is shown in Fig. 2.1: a number of microphones are positioned such as to pick up the sound of possibly multiple sound sources that are of interest. The microphone signals are then routed to the mixing console and may be subject to additional processing, such as dynamic range processing, artificial reverberation, etc., which is usually performed in the digital domain. The mixed signals are then amplified and sent to the loudspeakers, which are often arranged in a group-wise fashion (i.e., all the loudspeakers in the same loudspeaker group broadcast the same signal). Usually, the microphones and loudspeakers are positioned in such a way that, taking into account their directivity, the loudspeaker sound does not directly hit the microphones, i.e., no direct acoustic coupling between the loudspeakers and microphones exists. However, in nearly every sound reinforcement application it is unavoidable that the loudspeaker sound is reflected by the boundaries (walls, floor, and ceiling) of the acoustic environment (denoted as the “room”) and by subjects and objects within the environment. These reflections constitute an indirect acoustic coupling between the loudspeakers and microphones of the system.

The PA scenario can be modeled in a discrete-time context as shown in Fig. 2.2. All continuous-time signals involved are assumed to be bandlimited in

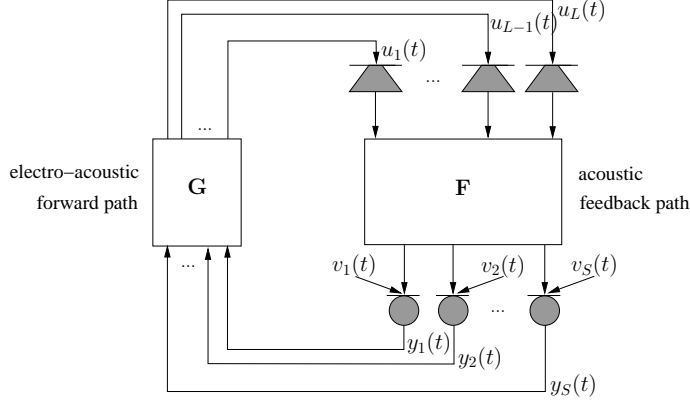


Figure 2.2: Discrete-time model of a PA system with  $S$  microphones and  $L$  loudspeakers.

such a way that they can be sampled at a standard sampling frequency (e.g.,  $f_s = 16$  kHz in speech applications,  $f_s = 44.1$  kHz in audio applications) and represented by their discrete-time counterparts<sup>1</sup>. If we represent the  $S$  source signals by  $v_i(t)$ ,  $i = 1, \dots, S$ , the corresponding  $S$  microphone signals as  $y_i(t)$ ,  $i = 1, \dots, S$ , and the  $L$  loudspeaker signals as  $u_j(t)$ ,  $j = 1, \dots, L$ , then the discrete-time closed-loop system model in Fig. 2.2 can be described by the following relations:

$$\bar{\mathbf{y}}(t) = \mathbf{F}(q, t)\bar{\mathbf{u}}(t) + \bar{\mathbf{v}}(t) \quad (2.1)$$

$$\bar{\mathbf{u}}(t) = \mathbf{G}[\bar{\mathbf{y}}(t), t]. \quad (2.2)$$

Here, the source signal, microphone signal, and loudspeaker signal vectors are defined as

$$\bar{\mathbf{v}}(t) = [v_1(t) \quad \dots \quad v_S(t)]^T \quad (2.3)$$

$$\bar{\mathbf{y}}(t) = [y_1(t) \quad \dots \quad y_S(t)]^T \quad (2.4)$$

$$\bar{\mathbf{u}}(t) = [u_1(t) \quad \dots \quad u_L(t)]^T \quad (2.5)$$

and the multi-channel acoustic feedback path  $\mathbf{F}(q, t)$  and electro-acoustic forward path characteristics  $\mathbf{G}[\cdot, t]$  are defined below.

Between each loudspeaker-microphone pair  $(j, i)$ ,  $j = 1, \dots, L$ ,  $i = 1, \dots, S$  there exists an acoustic coupling, which can be modeled by the acoustic feed-

<sup>1</sup>In our notation, we discriminate between continuous-time and discrete-time signals by using curly brackets for the former and round brackets for the latter. E.g.,  $x\{\tau\}$  is a continuous-time signal and  $x(t)$  is the corresponding discrete-time signal. The discrete time index  $t$  is related to the continuous time index  $\tau$  as  $t = \tau/T_s$  with the sampling interval defined as  $T_s = 1/f_s$ .

back path transfer function

$$F_{ij}(q, t) = f_{ij}^{(0)}(t) + f_{ij}^{(1)}(t)q^{-1} + \dots + f_{ij}^{(n_F)}(t)q^{-n_F} \quad (2.6)$$

where  $q$  denotes the discrete-time shift operator, i.e.,  $q^{-k}u_j(t) = u_j(t - k)$ . The multi-channel feedback path matrix in (2.1) is then defined as an  $S \times L$  polynomial matrix

$$\mathbf{F}(q, t) = \begin{bmatrix} F_{11}(q, t) & \dots & F_{1L}(q, t) \\ \vdots & \ddots & \vdots \\ F_{S1}(q, t) & \dots & F_{SL}(q, t) \end{bmatrix}. \quad (2.7)$$

The acoustic feedback path model is linear, time-varying, and of finite order<sup>2</sup>  $n_F$ . The linearity assumption is generally considered to be a reasonable one, since the effects of sound propagation and reflections in the acoustic environment (i.e., signal attenuations and time delays) are quasi level-independent. The finite-order assumption, which contrasts with the infinite impulse response (IIR) nature of room acoustics, can be justified by the observation that a typical room impulse response (RIR) has an exponentially decaying envelope such that it can be truncated to have  $n_F + 1 < \infty$  coefficients.

An example RIR, which was measured at  $f_s = 44.1$  kHz and truncated at a length of  $n_F + 1 = 4410$  coefficients (corresponding to 100 ms), is shown in Fig. 2.3. The frequency response of this RIR is displayed in Fig. 2.4. It can be seen that the magnitude response has an overall lowpass behavior as well as many local magnitude peaks and dips. This irregular behavior was explained and quantified by Schroeder in [2], under the assumption that the acoustic coupling is mainly due to reflections and not due to a direct acoustic path between the loudspeaker and microphone. The average frequency distance between two magnitude peaks is then about 10 Hz, and the peak magnitude can be up to 10 dB larger than the average magnitude in the frequency response [2].

In the electro-acoustic forward path, the  $S$  microphone signals are mixed and amplified to obtain  $L$  loudspeaker signals, and moreover, some additional signal processing is performed. Since usually nonlinear dynamics processing (e.g., compression, limiting, etc.) is involved here, the forward path mapping  $G_{ji}[\cdot, t]$  between the  $(i, j)$ th microphone-loudspeaker pair should be modeled as a non-linear, time-varying filter. However, to be able to perform a stability analysis of the closed-loop system, we will mostly assume that the forward path can be modeled by a linear, time-varying transfer function,  $\forall i, j$ ,

$$G_{ji}[\cdot, t] = G_{ji}(q, t) = g_{ji}^{(0)}(t) + g_{ji}^{(1)}(t)q^{-1} + \dots + g_{ji}^{(n_G)}(t)q^{-n_G} \quad (2.8)$$

<sup>2</sup>For ease of notation, we assume that all the acoustic feedback path transfer functions  $F_{ij}(q, t)$ ,  $i = 1, \dots, S$ ,  $j = 1, \dots, L$  have the same order  $n_F$ . This is also reasonable from a physical point of view, since the reverberation time in a room does not depend on the loudspeaker and microphone positions. We will make the same assumption further on for the electro-acoustic forward path transfer functions  $G_{ji}(q, t)$ .

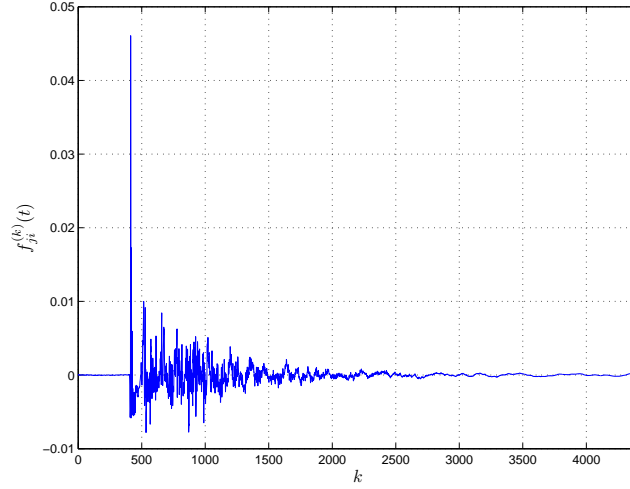


Figure 2.3: A typical room impulse response, measured at  $f_s = 44.1$  kHz and truncated at a length of  $n_F + 1 = 4410$  coefficients.

and

$$\mathbf{G}[\cdot, t] = \mathbf{G}(q, t) = \begin{bmatrix} G_{11}(q, t) & \dots & G_{1S}(q, t) \\ \vdots & \ddots & \vdots \\ G_{L1}(q, t) & \dots & G_{LS}(q, t) \end{bmatrix}. \quad (2.9)$$

If the forward path includes IIR components, such as IIR equalization filters, we have that  $n_G = \infty$ . We further assume that the sound sources have sufficient directivity and are close enough to the respective microphones, such that the acoustic transfer function matrix from the sources to the microphones is an identity matrix. These assumptions can be justified since these do not relate directly to the feedback problem.

While many sound reinforcement systems comprise multiple loudspeakers and microphones, most acoustic feedback control methods have been proposed in a single-channel context (i.e., for one loudspeaker and one microphone), without a framework for an extension to multi-channel systems being explicitly provided. For this reason, we will analyze the acoustic feedback problem and explain the acoustic feedback control methods in a single-channel context, and drop the subscripts  $i$  and  $j$ . We will however comment on the implications of extending a particular method to a multi-channel system whenever appropriate.

In a single-channel sound reinforcement system, the closed-loop frequency response from the source signal to the loudspeaker signal can be expressed as follows:

$$\frac{U(\omega, t)}{V(\omega, t)} = \frac{G(\omega, t)}{1 - G(\omega, t)F(\omega, t)}. \quad (2.10)$$



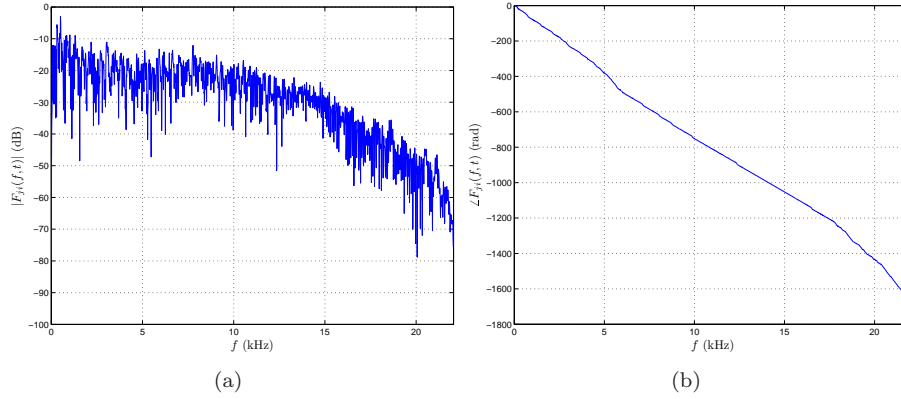


Figure 2.4: Frequency response of the RIR shown in Fig. 2.3: (a) Magnitude response, (b) Phase response.

Here,  $\omega \in [0, 2\pi]$  represents the radial frequency variable,  $U(\omega, t)$  and  $V(\omega, t)$  denote the short-term frequency spectra of the loudspeaker and source signal, and  $G(\omega, t)$  and  $F(\omega, t)$  are the short-term frequency responses of the forward and feedback path, which can be calculated using the short-time discrete Fourier transform (DFT). The frequency function  $G(\omega, t)F(\omega, t)$  appearing in the denominator of (2.10) is often referred to as the “loop response” of the system, and plays a crucial role in acoustic feedback control (the corresponding magnitude response  $|G(\omega, t)F(\omega, t)|$  is then referred to as the “loop gain” and the phase response  $\angle G(\omega, t)F(\omega, t)$  as the “loop phase”). It is well known that a closed-loop system can exhibit instability, which may lead to oscillations that, in an acoustic system, are perceived as howling. Stability analysis of linear closed-loop systems is by now a well-understood topic in control systems theory, which originated from early studies on feedback amplifiers. The current approach to closed-loop system stability analysis is based on a classical paper by Nyquist [3]. The *Nyquist stability criterion* can be formulated as follows<sup>3</sup>: if there exists a radial frequency  $\omega = 2\pi(f/f_s)$  for which

$$\begin{cases} |G(\omega, t)F(\omega, t)| \geq 1 & (2.11) \\ \angle G(\omega, t)F(\omega, t) = n2\pi, \quad n \in \mathbb{Z} & (2.12) \end{cases}$$

then the closed-loop system is unstable. If the unstable system is moreover excited at the critical frequency  $f$ , i.e., if the source signal contains a non-zero frequency component at  $f$ , then an oscillation at this frequency will occur. The

<sup>3</sup>We should note that the Nyquist stability criterion is defined in [3] for linear time-invariant systems. The stability of linear time-varying systems should be analyzed by the so-called circle criterion instead [4, Ch. 5]. However, to achieve consistency with the literature on acoustic feedback control, we will still define stability using the Nyquist criterion, under the assumption that the electro-acoustic forward path and the feedback path characteristics are slowly time-varying.

criterion in (2.11)-(2.12) is essential in the remainder of this paper, since any acoustic feedback control method effectively attempts at preventing either one or both of these conditions from being met.

With the aim of quantifying the achievable amplification in a sound reinforcement system with and without acoustic feedback control, it is customary to define a broadband gain factor  $K(t)$  as the average magnitude of the forward path frequency response  $G(\omega, t)$  and extract it from the forward path transfer function  $G(q, t)$ , i.e.,

$$G(q, t) = K(t)J(q, t) \quad (2.13)$$

with

$$K(t) = \frac{1}{2\pi} \int_0^{2\pi} |G(\omega, t)| d\omega. \quad (2.14)$$

Assuming now that  $J(q, t)$  is given, and that  $K(t)$  can be varied, the *maximum stable gain (MSG)* can be defined as follows:

$$\text{MSG}(t) \text{ [dB]} \triangleq 20 \log_{10} K(t) \quad \text{such that} \quad \max_{\omega \in \mathcal{P}} |G(\omega, t)F(\omega, t)| = 1 \quad (2.15)$$

$$= -20 \log_{10} \left[ \max_{\omega \in \mathcal{P}} |J(\omega, t)F(\omega, t)| \right] \quad (2.16)$$

where  $\mathcal{P}$  denotes the set of frequencies at which the phase condition (2.12) is fulfilled, i.e.,

$$\mathcal{P} = \{\omega | \angle G(\omega, t)F(\omega, t) = n2\pi\}. \quad (2.17)$$

From a statistical analysis of room acoustics, Schroeder concluded that in a sound reinforcement system without feedback control and having a reverberation time of  $T_{60}$  s and a bandwidth of  $B$  Hz, the average MSG can be calculated as [2]

$$\text{MSG}(t) \text{ [dB]} = -10 \log_{10} [\log_{10}(BT_{60}/22)] - 3.8. \quad (2.18)$$

The gain margin is defined as the difference between the MSG and the actual gain of the system. From a sound quality point of view, a gain margin of 2 to 3 dB is recommended to avoid audible ringing effects [2],[5].

## 2.3 State of the Art in Acoustic Feedback Control

As already mentioned, we will only deal with automatic methods for acoustic feedback control. A review of manual feedback control methods is given in [6]. These methods are based on a proper microphone and loudspeaker selection and positioning, suppression of discrete room modes using notch filters, and equalization of the entire room response using 1/3 octave graphic equalizer filters, and may result in an MSG increase of 5 to 8 dB [6].

Automatic feedback control methods may be categorized into four classes: phase modulation methods, gain reduction methods, spatial filtering methods, and room modeling methods.

### 2.3.1 Phase Modulation Methods

One of the earliest approaches to acoustic feedback control consists in *frequency shifting (FS)* the microphone signals before these are amplified and sent to the loudspeakers. The FS approach can largely be attributed to Schroeder, who published a number of papers on this topic in the early 1960s [2],[7]-[9]. By applying FS, the loop gain can be smoothed, such that ideally, the MSG is determined by the average magnitude response rather than the peak magnitude response [9]. Since the average frequency distance between two magnitude peaks in a room response was found to be around 10 Hz, the optimal FS value is expected to be around 5 Hz [2]. An MSG increase up to 14 dB was reported [7], however, the subjectively acceptable MSG increase is limited to 6 dB if audible beating effects due to the FS operation are to be avoided [2],[9]. It is claimed in [9] that a frequency shift of 5 Hz is inaudible both for speech and music signals. The earliest FS implementations were based on analog single-sideband modulation [10] or phase modulation [11]. More recently, a digital FS implementation using a truncated FIR Hilbert filter has been proposed [12]. A drawback of the FS approach is that it does not preserve the harmonic relations between tonal components in voiced speech and music signals. It was shown in [13] that a bandwidth compression does preserve harmonic relations and results in a feedback stability improvement similar to the FS approach.

Another early feedback control method employs *phase modulation (PM)* in the electro-acoustic forward path, with the aim of bypassing the phase condition (2.12) in the Nyquist criterion. In 1958, Mishin [14] described a sinusoidal PM approach in which the choice of the modulation parameter relates to the zeros of Bessel functions of the first kind. In a 1968 paper by Nishinomiya [15], an MSG increase up to 7 dB is reported using sinusoidal *frequency modulation (FM)*, which is conceptually equivalent to sinusoidal PM. Guelke and Broadhurst [5] applied the sinusoidal PM technique in the context of reverberation enhancement (RE) systems, using a very low modulation frequency ( $\sim 1$  Hz), and resulting in a 4 dB MSG increase. The apparent suitability of PM, FM, and other periodic modulations for feedback control in digital RE systems resulted in a renewed interest in these methods in the 1990s. Svensson [16] and Nielsen and Svensson [17] provided a unifying approach to phase-modulating feedback control (PFC) in which the modulators, including sinusoidal PM, FM, *amplitude modulation (AM)*, and *delay modulation (DM)*, are viewed as linear periodically time-varying filters. Moreover, they showed that the FS approach also fits into this framework, hence labeling FS-based feedback control as a special case of PFC. Svensson [16] reported an average 4 dB MSG increase with a synthetic acoustic feedback path, while Nielsen and Svensson [17] obtained

MSG increases up to 8 dB in real room acoustic feedback scenarios. Poletti [18] was the first to study the performance of PFC (in particular using a FS approach) in multi-channel sound systems. His somewhat discouraging conclusion was that the stability improvement due to FS reduces as the number of channels increases. Finally, while the impact of the PFC approach on sound quality may be considerable, to our knowledge a perceptual evaluation has not yet been reported on.

In Section 2.4, a more extensive treatment of the PFC approach is provided.

### 2.3.2 Gain Reduction Methods

The most straightforward approach to acoustic feedback control, is to automate the actions that a human operator would undertake for preventing or eliminating howling in a sound reinforcement system. These actions usually consist in reducing the electro-acoustic forward path gain, such that the system moves away from magnitude condition (2.11) in the Nyquist criterion. Depending on the width of the frequency band in which the gain is actually reduced, we can discriminate between three gain reduction methods:

1. in *automatic gain control (AGC)* methods [19]-[21], the gain is reduced equally in the entire frequency range by decreasing the broadband gain factor  $K(t)$  defined in (2.14),
2. in *automatic equalization (AEQ)* [21]-[29], the gain reduction is applied in critical subbands of the entire frequency range, namely those subbands in which the loop gain is close to unity,
3. in *notch-filter-based howling suppression (NHS)* [30]-[55], the gain is reduced in narrow frequency bands around critical frequencies, i.e., frequencies at which the loop gain is close to unity.

Every gain reduction method has to be activated in some way, when a closed-loop instability or a tendency towards instability is detected. Only a few gain reduction methods have been proposed which are based on a proactive instability detection: these are either based on an online measurement of the feedback path magnitude response [50],[56] or on an early detection of the spectral accumulation effect that can be observed at critical frequency components in the microphone signal [27]-[29],[57],[58]. Most gain reduction methods are reactive, in the sense that howling can usually be perceived before it is actually detected. In these methods, howling detection is typically based on a combined spectral and temporal analysis of the microphone signal. Due to the sinusoidal nature of howling, the microphone signal frequency components having the largest magnitude are considered to be candidate howling components. The true howling components within this set of candidates can then be discriminated from the source signal tonal components (originating from voiced speech or musical tones) using several criteria. Spectral criteria for discriminating be-

tween howling and tonal components are based on one or more of the following features: the power ratio of the candidate howling component and the entire spectrum [24]-[26],[34]-[36],[42],[43],[47]-[49],[53], the power ratio of the candidate howling component and its (sub-)harmonics [31],[33]-[35],[41], and the power ratio of the candidate howling component and its neighboring frequency components [21],[27]-[29]. On the other hand, temporal criteria for howling detection rely on the observation that howling components typically persist for a longer time than tonal components [19],[20],[30],[31],[33],[36],[41]-[43],[47] and exhibit an exponentially increasing magnitude until the sound reinforcement system saturates [27]-[29]. A comparative evaluation of these spectral and temporal howling detection criteria is reported in [59].

The AGC method is the earliest gain reduction method, which was proposed by Patronis in 1978 [19],[20]. If howling is detected, the broadband gain is immediately reduced, and after a specified time interval the gain is restored to the initial value. Candidate howling frequencies are discriminated from tonal source signal components by assuming that howling components persist for several seconds. A subband implementation of this method was proposed by Ando [21], featuring a spectral approach to howling detection by evaluating power ratios between adjacent subbands. Obviously, AGC methods do not increase the MSG since the spectral shape of the loop gain is not altered. The main strength of AGC methods is their reliability: if the gain is sufficiently reduced, an unstable system is guaranteed to be stabilized. Therefore, many other acoustic feedback control methods include an AGC method as a “rescue procedure” that is activated if all else fails, see, e.g., [21],[34],[35],[53],[60].

The AEQ method follows directly from the subband approach to AGC, as proposed by Ando [21]. If howling detection is performed in frequency subbands, then the gain reduction can be limited to those subbands in which howling is detected. Hanajima *et al.* [24],[25] further improved the subband howling detection, by first performing a howling detection in relatively wide subbands, and subsequently dividing the most critical subband in narrower subbands in which the howling detection is then repeated. They use 10 logarithmically spaced wide subbands in the 10–10000 Hz range, which are then divided into 10 linearly spaced narrower subbands to obtain a more accurate howling detection. An even more advanced howling detection can be found in the AEQ method of Osmanovic *et al.* [27]-[29]. The detection criterion consists of a linear combination of two features that are calculated for all candidate howling components: the “slopesness” is a temporal feature that models the exponential buildup of a howling component, while the “peakness” is a spectral feature that estimates the power ratio of a candidate howling component and its neighboring frequency components. For the equalization, Osmanovic *et al.* use 14 logarithmically spaced 8th order IIR bandstop filters in the speech range 300–6000 Hz [27]-[29].

The NHS methods can be divided into two categories, i.e., one-stage and two-stage NHS methods, depending on whether the howling detection and notch filtering are performed jointly or separately. The earliest NHS methods are one-stage methods, which are usually implemented using adaptive notch filters (ANF). In 1989, Foley proposed the adaptive periodic noise canceller [30] for speech applications, which is an FIR-ANF that is able to track and cancel a narrowband component in the microphone signal. Since the FIR-ANF in [30] is adapted using the least mean squares (LMS) algorithm, it is expected to be too slow to cancel tonal speech components, which vary more quickly in time than howling components. Also, the FIR-ANF is preceded by a delay of 8 samples such that it cannot cancel the short-term-correlated speech formants. Foley's FIR-ANF was shown to be  $H_\infty$ -robust for a first-order feedback path (i.e.,  $n_F = 1$ ), provided that the LMS stepsize is properly chosen [39],[45]. Staudacher [38] proposed an extension to Foley's FIR-ANF, by using a variable LMS stepsize that increases as the FIR-ANF input signal power increases, such that the convergence is accelerated when howling occurs. To reduce the impact of the ANF on sound quality, the notch filter bandwidth should be as small as possible. A disadvantage of the FIR-ANF implementation is that a large filter order is required to obtain a narrowband notch characteristic, e.g., Foley [30] and Staudacher [38] use 32nd order filters to cancel a single narrowband component. If multiple howling components are to be cancelled, the required FIR-ANF filter order may become unpractically large. Following this observation, several IIR-ANF implementations have been proposed, which only require a biquadratic (i.e., second-order) filter structure to cancel one narrowband component. The main difficulty with IIR-ANF implementations is that the least squares (LS) cost function associated with the howling component frequency estimation is typically non-convex. Kuo and Chen [32] proposed a constrained biquadratic IIR-ANF in which the global minimum of the LS cost function can be found with high probability by increasing the notch bandwidth during the howling detection process. Once howling has been detected, the notch filter is activated in the electro-acoustic forward path with a reduced bandwidth, to avoid a loss of sound quality. Another approach to bypass local minima in the LS cost function associated with the IIR-ANF implementation, is to only adapt the FIR part of the filter, and subsequently copy the numerator coefficients to the denominator [41], perhaps after including some scaling factor [37],[52]. A biquadratic IIR-ANF implementation featuring an advanced howling detection method was proposed by Porayath and Mapes-Riordan [41]: a howling frequency is detected when it has a power that is 30 dB larger than its first harmonic and when this power difference persists for at least 50–100 ms. Since the power spectral density is however hard to estimate when using the ANF approach, a different howling detection method was recently proposed by Gil-Cacho *et al.* [55], which is based on running multiple regularized biquadratic IIR-ANFs in parallel with different regularization factors. Yet another second-order ANF implementation was proposed by Wei *et al.* [46], in which the input samples to the ANF consist of phase-shifted instead of time-shifted microphone signal samples.

The two-stage NHS method, which is by now probably the most popular gain reduction method for acoustic feedback control, originates from the work of Lewis *et al.* [31],[33] and Er *et al.* [34],[35] in the early 1990s. A non-parametric frequency analysis of the microphone signal is computed using a fast Fourier transform (FFT) algorithm, from which the candidate howling components are determined using a peak picking algorithm. The power of the candidate howling components is then compared to an absolute power threshold [34],[35], to the average signal power [34],[35], and to the (sub-)harmonics power [31],[33]-[35] to determine if howling occurs. This spectral criterion is combined with a temporal criterion for howling detection by Lewis *et al.* [31],[33]. Whenever howling is detected, biquadratic notch filters are inserted in the electro-acoustic forward path. Several improvements to the methods by Lewis *et al.* and Er *et al.* have been reported. Kawamura *et al.* [36],[42] propose an online modification of the thresholds used in the spectral and temporal howling detection criteria, steered by estimates of the background noise spectrum, the source signal spectrum, the reverberation time, and the acoustic feedback path response. Lane *et al.* [40] apply a parametric frequency analysis instead of the non-parametric analysis proposed earlier, using a set of adjustable bandpass filters having relatively wide passbands as compared to the stopbands of the notch filters. An alternative way of determining the set of candidate howling components was proposed by Williams [43]: instead of executing a peak picking algorithm on the FFT magnitude spectrum estimate, a so-called “ballistics procedure” is applied to model the temporal build-up of narrowband components such that components with an increasing power can be identified. Rocha and Ferreira [47] and Börsch [48],[49] replace the FFT algorithm in the non-parametric frequency analysis by an odd FFT algorithm and a frequency-warped FFT algorithm, respectively. Moreover, the frequency analysis described by Börsch [48],[49] is the only non-parametric method which includes a compensation for the estimation errors due to the limited FFT resolution. In [50], Rombouts *et al.* propose a proactive howling detection method applied to NHS, based on the estimation of critical closed-loop system frequencies from an adaptive estimate of the feedback path response. Abe [51] was the first to consider NHS in a multi-channel sound reinforcement system, and succeeded in reducing the computational and memory requirements by frequency-analyzing the individual microphone signals with a low-resolution FFT algorithm and the mixed signal with a high-resolution FFT algorithm. Finally, Somasundaram [53] proposes an advanced spectral howling detection criterion, in which the power of the candidate howling component is compared to a threshold that is calculated using the mean and standard deviation of the entire FFT spectrum estimate. Furthermore, the notch filters used in [53] are gradually enabled and disabled using a so-called leaky integrator, to avoid artifacts in the loudspeaker signal.

Since the majority of the available gain reduction methods are described in patents, not many experimental results are available and no MSG increase values have been reported. However, from Schroeder’s statistical analysis of a



feedback path frequency response [2] it can be expected that if the loop gain could be perfectly smoothed using an AEQ or NHS approach, a maximal MSG increase of about 10 dB may be achieved. The two-stage NHS method, being the most popular of all gain reduction methods, will be described in more detail in Section 2.5.

### 2.3.3 Spatial Filtering Methods

Spatial filtering methods for acoustic feedback control aim at altering the loop response  $G(\omega, t)F(\omega, t)$  of the closed-loop system by using microphone and/or loudspeaker arrays of which the received/transmitted signals are processed by beamforming filters. The general objective is then to design a microphone array beamformer that has its main lobe (i.e., its maximal spatial response) in the direction of the source while having a null (i.e., zero spatial response) in the direction of the loudspeaker, and/or a loudspeaker array with the main lobe directed towards the audience and a null in the direction of the microphone. The first spatial filtering approach to acoustic feedback control was proposed by Duong *et al.* in 1984 for hands-free telephony applications [61], focusing on the combined use of a microphone and loudspeaker array for a single-channel scenario with fixed microphone and loudspeaker positions. The stringent spatial constraints (i.e., the microphone and loudspeaker array are to have the same center and lie orthogonal to each other) make this method rather impractical for many sound reinforcement applications. A more flexible approach, which allows for scenarios with arbitrary microphone and loudspeaker array positions, consists in adapting the beamformer coefficients based on the available sound signals. Obviously, an adaptive microphone array is more straightforward to implement than an adaptive loudspeaker array, since the latter does not collect any information on the acoustic environment. A fundamental problem that occurs when computing the coefficients of an adaptive microphone array beamformer in a closed-loop system, is the fact that the source signal is correlated with the loudspeaker signal (i.e., the loudspeaker signal can be calculated by filtering the source signal with the closed-loop response, see (2.10)). Due to this correlation, a conventional adaptive beamforming algorithm will not converge to the desired solution, and consequently, part of the source signal will eventually be attenuated while part of the feedback signal will still appear in the output of the microphone array. Several solutions to this correlation problem have been proposed. Janse and Belt [62] propose the combined use of an adaptive feedback canceller (AFC) and a microphone array beamformer. By feeding the feedback-compensated signal from the AFC to the microphone array, the influence of the feedback signal on the beamforming algorithm can be decreased. In this case, however, it is not possible to create a beamformer null directed towards the loudspeaker, since the feedback-compensated signal (ideally) does not provide any information on the loudspeaker position. Another solution was proposed by Kobayashi *et al.* [63],[64], in which the coefficients



of an adaptive microphone array beamformer outside the closed signal loop are computed by cancelling the source signal using a null beamformer (NBF) and inserting an artificial source signal. The adaptive beamformer coefficients are then copied to a microphone array beamformer in the closed signal loop, resulting in an MSG increase up to 15 dB [63],[64]. Due to the source signal cancellation, the adaptive beamformer can unambiguously identify the loudspeaker direction, however, the direction of the source w.r.t. the microphone array needs to be known a priori [63] or estimated by an adaptive NBF [64]. The artificial source signal, of which the design is not specified in [63],[64], serves to constrain the adaptive beamformer response to unity in the source direction. A more recent solution to the correlation problem in adaptive microphone array beamforming was proposed by Rombouts *et al.* [65],[66], and consists in prewhitening the source signal component in the adaptive beamformer desired signal using an adaptive decorrelation filter that is estimated concurrently with the beamformer coefficients. This approach was shown to result in an MSG increase between 7 and 14 dB (depending on the reverberation time of the room), while it does not require a priori information on the source position and is considerably cheaper than the approach in which an AFC is also used. Finally, a fundamentally different approach to spatial filtering for acoustic feedback control was proposed by Goodwin and Elko [67],[68]. In the so-called “beam dithering” approach, a loudspeaker array is steered by a beamformer of which the coefficients are varied periodically with time, by time-stepping through a discrete sequence of approximate Chebyshev coefficients. In this way, a spatial modulation is obtained that provides a smoothing of the loop gain, comparable to the smoothing effect obtained with the phase modulation methods for acoustic feedback control. An MSG increase up to 6 dB has been obtained [68], however, the spatial constraints of the beam dithering approach are rather stringent (in that the audience should always be in the main beamformer lobe, while the microphones should be in the sidelobes) and a perceptual calibration of the system is required [67].

### 2.3.4 Room Modeling Methods

In room modeling methods for acoustic feedback control, a model of the acoustic feedback path is identified either off-line (during the initialization of the sound reinforcement system) or on-line (during the operation of the sound reinforcement system). We can distinguish between two room modeling methods, depending on how the model is subsequently applied for acoustic feedback control. In *adaptive feedback cancellation (AFC)*, the acoustic feedback path model is used to predict the feedback signal component in the microphone signal (i.e., the part of the microphone signal that stems from the loudspeaker signal through the acoustic coupling). The predicted feedback signal is then subtracted from the microphone signal, hence resulting in a feedback-compensated signal, which is in fact an estimate of the source signal component in the mi-

crophone signal. If an accurate model of the acoustic feedback path can be identified, then the AFC method achieves a nearly complete elimination of the acoustic coupling (i.e., the loop gain comes close to zero for all frequencies), and consequently very large MSG increases may be obtained. Alternatively, the inverse of the acoustic feedback path can be modeled and identified, and this inverse model can then be inserted in the closed signal loop to optimally equalize the microphone signal. This approach is referred to as *adaptive inverse filtering (AIF)*, and ideally results in a perfect smoothing of the loop gain, for which the MSG increase can be expected to be around 10 dB [2].

The AIF approach has received only little attention in the context of acoustic feedback control. In 1994, Ushiyama *et al.* [69] proposed an inverse filtering approach in which an inverse model of the minimum-phase components in the acoustic feedback path is identified off-line. It is observed that a smoothing of the inverse model frequency response increases the robustness of the (time-invariant) inverse model w.r.t. time variations in the acoustic feedback path response. Another off-line approach to inverse filtering was proposed by Nagata *et al.* [22],[23], and consists in automatically adjusting a large number of equalizers in the electro-acoustic forward path, based on an off-line measurement of the acoustic feedback path response using a noise probe signal. Finally, a hybrid AIF-AFC approach was proposed by Janse and Belt [62] and Schmidt and Haulick [70], in which the inverse model coefficients are adjusted based on the acoustic feedback model that is identified in the AFC algorithm. More results on the AIF approach can be found in the literature on acoustic dereverberation and equalization, see, e.g., [71]-[74].

In the AFC approach, which is conceptually similar to the well-known acoustic echo cancellation (AEC) approach, an adaptive filter is used to model, identify, and track the impulse response of the acoustic feedback path. Analogously to the correlation problem found in adaptive microphone array beamforming (see Section 2.3.3), the fundamental problem encountered in AFC lies in the fact that, unlike in the AEC case, the adaptive filter's input signal (i.e., the loudspeaker signal) and disturbance signal (i.e., the source signal) are now correlated, see (2.10). Applying a standard adaptive filtering algorithm to the AFC problem hence results in a biased estimate of the acoustic feedback path impulse response [75]-[77], and consequently, the source signal component in the microphone signal ends up being partially cancelled. For this reason, a decorrelation method is generally incorporated in the AFC scheme which is either included in the closed signal loop or in the adaptive filtering circuit [77], see [78] for an overview and comparative evaluation.

Decorrelation in the closed signal loop can be accomplished by injecting a noise signal, by including a nonlinear or time-varying signal operation, or by inserting a processing delay in the electro-acoustic forward path. The earliest AFC reference appears to be a 1988 patent by Ibaraki *et al.* [79], in which a white noise signal is injected in the closed signal loop non-continuously (e.g., dur-

ing source signal pauses) to identify the low-frequency response of the acoustic feedback path. Goertz [80] proposes to inject a white noise signal continuously and reports a 5 dB MSG increase in a severely undermodeled AFC scenario (i.e., the adaptive filter length being only 1/15 of the feedback path length). Decorrelation by continuous white noise injection was also applied by Stott and Wells [60], van Waterschoot [81], and Schmidt and Haulick [70]. With the aim of reducing the sound quality deterioration due to noise injection, several attempts have been made to shape the spectrum of the injected noise signal such that it becomes less perceptible. Goertz [80] proposes to use A-weighted noise instead of white noise, while van Waterschoot [81] and Janse and Tchang [82] apply a time-varying noise shaping based on a psychoacoustic model. However, to obtain an AFC performance comparable to the methods using white noise injection, the psychoacoustically shaped noise has to be amplified to a level at which it is found to be even more disturbing than white noise [81]. Decorrelation in the closed signal loop can also be achieved by including a nonlinear or time-varying signal operation in the electro-acoustic forward path. Janse *et al.* [83]-[85] propose to use a frequency shifter or a periodic phase or delay modulator. The AFC robustness can then be increased since these decorrelating operations also have a stabilizing effect on the closed-loop system (see Section 2.3.1). Another nonlinear decorrelation technique, which was adopted from the stereo AEC literature [86] by van Waterschoot *et al.* [87] and Schmidt and Haulick [70], consists in adding a half-wave rectified version of the loudspeaker signal to the original loudspeaker signal, yet was found to improve the AFC performance only marginally [87]. Finally, in the context of hearing aid AFC applications, inserting a processing delay in the electro-acoustic forward path has been proposed for reducing the correlation between the source and loudspeaker signals [75],[88]. The motivation for this approach is that the source and loudspeaker signal cross-correlation function is expected to decrease for increasing time lags, which is particularly the case for voiceless speech signals.

While most of the above decorrelation techniques are rather effective when applied in the closed signal loop, their effect on the sound quality may be detrimental. For this reason, there has been an increased interest in the application of decorrelating signal operations in the adaptive filtering circuit, such that the closed loop signals remain unaffected. A first approach, that was proposed by Ortega *et al.* [89],[90], consists in having the adaptive filter preceded by a processing delay. The resulting decorrelation effect is similar to when a processing delay is inserted in the electro-acoustic forward path. However, the delay length in the adaptive filtering circuit should not exceed the initial delay (i.e., the “dead time”) in the acoustic feedback path impulse response (e.g., with the acoustic feedback path impulse response shown in Fig. 2.3, the maximum allowable processing delay would be 405 samples). A second approach consists in the use of decorrelating prefilters, that are designed to whiten the source signal component in the microphone signal. This approach was adopted from hearing aid AFC research [76],[91],[92], and was applied to PA systems by

van Waterschoot *et al.* [77], [87] and to in-car communication systems by Ortega *et al.* [93]. A fundamental difficulty lies in the concurrent identification of the optimal prefilter and the acoustic feedback path model from the closed-loop signals. This identification problem was tackled following a prediction-error-method (PEM)-based approach [94, Ch. 3], [95, Ch. 7] by Rombouts *et al.* [96]-[101]. The PEM-based AFC approach developed in [96]-[98] is based on a nonstationary all-pole source signal model, the inverse of which is then used as a time-varying FIR decorrelating prefilter in the AFC scheme. The robustness of the PEM-based AFC approach was further improved in [99] by including some additional features such as adaptation control and the joint use of a foreground and background adaptive filter. Also, efficient subband and frequency domain implementations of the PEM-based AFC method were proposed in [99]. It was shown by van Waterschoot *et al.* [100], [101] how the convergence of the PEM-based AFC scheme can be improved even further by incorporating prior knowledge on the source signal and the acoustic feedback path through regularization.

In recent years, several remaining issues concerning the AFC approach have been analyzed and further improvements have been reported. The overall performance of the AFC approach may be improved by combining AFC with other acoustic feedback control methods and signal enhancement techniques, leading to so-called hybrid AFC methods. Ortega *et al.* [89], [90] propose the combination of AFC with a residual feedback and noise suppression postfilter, and this hybrid AFC scheme was further expanded by Janse and Belt [62] with an adaptive microphone array beamformer and an AIF. The combination of AFC with an NHS method is of particular interest due to the robustness of the NHS methods to system instability: Schmidt *et al.* [70], [102] use an ANF that operates on the AFC feedback-compensated signal, while Rombouts *et al.* [50], [99] apply a two-stage NHS method in which the howling detection is based on a frequency analysis of the AFC feedback path estimate. The considerable computational complexity of the AFC approach in room acoustic applications is another issue that has recently been addressed. An interesting approach towards AFC complexity reduction was proposed by Okumura and Fujita [103] and consists in applying two or more parallel adaptive filters, preceded by a processing delay in the adaptive filtering circuit, to model a single acoustic feedback path. The first filter (which can be understood to model the late reverberation in the acoustic feedback path impulse response) has many coefficients that are adapted not very frequently using a transform domain approach, while the second filter (which then models the early reflections) is a short filter that is adapted at each time instant using a time domain adaptive filtering algorithm. A final issue is related to AFC in audio applications: none of the above-mentioned AFC methods has been designed to operate in a high-fidelity audio environment. When applying decorrelation in the closed signal loop, introducing signal distortion is unavoidable, while decorrelation techniques in the adaptive filtering circuit are typically based on the assumption that the

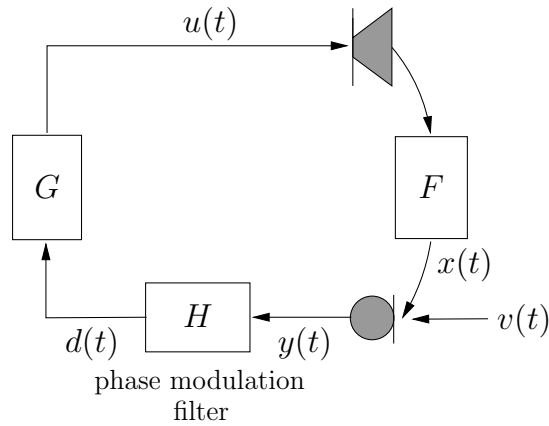


Figure 2.5: Phase-modulating feedback control (PFC) by inserting a phase modulation filter in the electro-acoustic forward path.

source signal is a speech signal. Van Waterschoot and Moonen [104],[105] have recently proposed a novel PEM-based AFC method that is designed particularly for audio signals, but performs equally well in speech applications. The method is based on a cascade of two source signal models, where one models the tonal components in the source signal and the other one models the source signal noise components.

## 2.4 Phase-Modulating Feedback Control

### 2.4.1 Concept

The goal of PFC is to control the phase of the microphone signal in such a way that every frequency component in the feedback signal has a different phase each time it arrives at the microphone after having traveled one cycle around the closed signal loop [17]. In this way, the phase condition in the Nyquist criterion (2.12) can be guaranteed not to hold for the same frequency at two successive time instants, hence the closed-loop system stability can be improved, regardless of the magnitude condition (2.11). The PFC goal can be achieved by inserting a phase modulation (PM) filter in the electro-acoustic forward path, which operates directly on the microphone signal  $y(t)$  and delivers an output signal  $d(t)$  to the forward path processing unit  $G(q, t)$ , see Fig. 2.5.

The behavior of a PM filter can be analyzed elegantly using the theory of linear time-varying (LTV) filters [16],[17]. A discrete-time<sup>4</sup> LTV filter can be

<sup>4</sup>Note that in [16],[17], the theory of LTV filters is described for the continuous-time case.

described in the time domain using the input-output relationship [106]

$$d(t) = \sum_{\tau=-\infty}^{\infty} h(\tau, t)y(t - \tau) \quad (2.19)$$

with  $h(\tau, t)$  the LTV filter's impulse response, which depends on both the observation time instant  $t$  and the time difference  $\tau$  between excitation and observation. If the LTV filter is moreover periodically time-varying (LPTV) with a period  $T_m$  that corresponds to an integer number of sampling periods, i.e.,  $T_m = NT_s$ , then the periodic LTV frequency response

$$H(\omega, t) = \sum_{\tau=-\infty}^{\infty} h(\tau, t)e^{-j\omega\tau} \quad (2.20)$$

admits an  $N$ -point discrete Fourier transform (DFT) representation with coefficients

$$\mathcal{H}(\omega, n) = \sum_{t=0}^{N-1} H(\omega, t)e^{-jn(2\pi/N)t} \quad (2.21)$$

and the input-output relationship in (2.19) can be written in the frequency domain as follows [106],

$$D(\omega) = \frac{1}{N} \sum_{n=0}^{N-1} \mathcal{H}(\omega - n\omega_m, n)Y(\omega - n\omega_m) \quad (2.22)$$

with  $\omega_m = 2\pi/N$  the LPTV filter fundamental frequency. In other words, the LPTV filter output spectrum is a sum of  $N$  frequency-weighted and frequency-shifted versions of the input spectrum. The LPTV filter frequency response DFT coefficients  $\mathcal{H}(\omega, n)$  are usually referred to as the carrier response (for  $n = 0$ ) and the sideband responses (for  $n \neq 0$ ).

It can be seen from (2.22) that the output spectrum also contains a non-frequency-shifted version of the input spectrum (for  $n = 0$ ), which is undesirable in view of the acoustic feedback control performance [17]. The contribution of the non-frequency-shifted version of the input spectrum to the total output spectrum is quantified using the so-called carrier suppression<sup>5</sup> [17],

$$\text{CS [dB]} = -10 \log_{10} \left[ \int_0^{2\pi} |\mathcal{H}(\omega, 0)|^2 d\omega \right] \quad (2.23)$$

and it has been hypothesized that the CS corresponds to an upper bound for the increase in MSG that can be obtained using the PFC approach [17].

---

<sup>5</sup>Note that our definition of the carrier suppression as given in (2.23) is somewhat more general than the definition in [17], because we do not restrict the sideband responses  $\mathcal{H}(\omega, n)$  to be independent of  $\omega$ .

Another hypothesis stated in [17] is that a modulation scheme having a larger number of sideband responses with a relatively large power  $|\mathcal{H}(\omega, n)|^2$  provides a better acoustic feedback control performance, since in this case, more input signal energy is shifted away from the original (carrier) frequency. However, this hypothesis is based on a continuous-time analysis and may not hold in a discrete-time context, since aliasing will fold all the input signal energy that has been shifted above the Nyquist frequency back to lower frequencies.

The following four PM techniques have been studied in the context of acoustic feedback control [17]:

1. Sinusoidal PM [5],[14],[16],[17]: A sinusoidal PM filter has a frequency response

$$H(\omega, t) = e^{j\beta \sin \omega_m t} \quad (2.24)$$

which is characterized by frequency-independent carrier and sideband responses  $\mathcal{H}(n)$  that correspond to the Bessel functions of the first kind and order  $n$ ,

$$\mathcal{H}(n) = J_n(\beta), \quad n = 0, \dots, N - 1. \quad (2.25)$$

These functions are plotted as a function of the so-called modulation index  $\beta$  in Fig. 2.6.

2. Sinusoidal frequency modulation (FM) [15],[17]: The effect of a sinusoidal FM filter with a modulation frequency  $f_m = \omega_m(f_s/2\pi)$  and a modulation depth  $\Delta_f$  can be shown to be identical to the effect of a sinusoidal PM filter with the same modulation frequency  $f_m$  and a modulation index  $\beta = \Delta_f/f_m$  [17].
3. Frequency shifting (FS) [2],[7],[11],[12],[17],[18]: An FS device can either be viewed as a nonlinear time-invariant system or as an LPTV system. From the latter interpretation, it can be shown that an FS operation with a frequency shift of  $f_m = \omega_m(f_s/2\pi)$  Hz corresponds to a PM operation with a phase function that increases linearly with time [11],[17], i.e.,

$$H(\omega, t) = e^{j\omega_m t} \quad (2.26)$$

and, as a consequence,

$$\mathcal{H}(n) \begin{cases} = 1, & n = 1 \\ = 0, & n = 0, 2, \dots, N - 1. \end{cases} \quad (2.27)$$

$$(2.28)$$

In other words, an FS device can be described as an LPTV filter with zero carrier response (i.e.,  $CS = \infty$ ) and only one non-zero sideband response (for  $n = 1$ ).

4. Sinusoidal delay modulation (DM) [14],[16],[17]: A sinusoidal DM filter varies the input signal's time delay sinusoidally around a time delay offset  $\tau_0$  with a maximum time delay deviation  $\Delta_\tau$  and a modulation frequency  $\omega_m$ , as can be seen in its frequency response

$$H(\omega, t) = e^{-j\omega(\tau_0 + \Delta_\tau \sin \omega_m t)}. \quad (2.29)$$

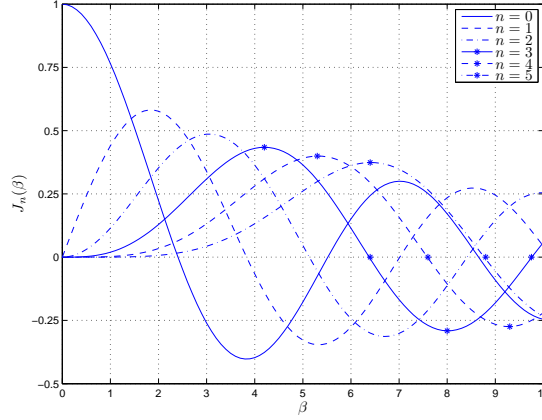


Figure 2.6: Bessel functions of the first kind for different orders  $n = 0, \dots, 5$ .

This can be interpreted as a sinusoidal PM filter with the same modulation frequency  $\omega_m$  and a modulation index  $\beta = \omega\Delta_\tau$  that is proportional to the original (carrier) frequency  $\omega$ . As a consequence, the corresponding carrier and sideband responses are frequency-selective (as opposed to the frequency-independent PM, FM, and FS responses),

$$\mathcal{H}(\omega, n) = J_n(\omega\Delta_\tau), \quad n = 0, \dots, N - 1. \quad (2.30)$$

From the above expression, it can be understood that a sinusoidal DM filter performs poorly in the low frequency range since in this case, the carrier response has a much larger magnitude than the sideband responses (see Fig. 2.6).

## 2.4.2 Realization

The sinusoidal PM, sinusoidal FM, and FS filters are usually realized by operating on the so-called analytical representation of the microphone signal  $y(t)$ . In continuous time, the analytical signal  $y_a\{\tau\}$  is defined as follows [107],

$$y_a\{\tau\} = y\{\tau\} + j\hat{y}\{\tau\} \quad (2.31)$$

where  $\hat{y}\{\tau\}$  represents the Hilbert transform of  $y\{\tau\}$ . The corresponding discrete-time analytical signal  $y_a(t)$  can be calculated in several ways. A first approach is to design an FIR filter  $L(q)$  approximating the Hilbert transform such that an approximation to  $\hat{y}(t)$  can be calculated as  $L(q)y(t)$  [108], and then the discrete-time analytical signal can be obtained as  $y_a(t) = y(t) + jL(q)y(t)$ . Since the so-called Hilbert filter  $L(q)$  is non-causal, a processing delay of half the filter length of  $L(q)$  has to be introduced in the signal path [12]. Another



drawback is that this approach does not preserve the orthogonality between  $y(t)$  and  $\hat{y}(t)$  which can be obtained in the continuous-time case [109]. A second approach is to design two complex FIR filters, so-called dual quadrature FIR filters, that produce orthogonal approximations to  $y(t)$  and  $\hat{y}(t)$ , respectively, which are then added according to (2.31) [110]. Unfortunately, this approach does not preserve the original data since the real part of the discrete-time analytical signal is not exactly equal to  $y(t)$  [109]. In a third approach, which combines the desirable properties of original data preservation in the real part and orthogonality between the real and imaginary part of the discrete-time analytical signal,  $y_a(t)$  is approximated as the inverse DFT of a one-sided discrete spectrum (with zero negative frequency content) that is calculated using the DFT of the original signal  $y(t)$  [109]. This approach is frame-based, hence a processing delay equal to the frame size minus the frame overlap is required. We will use this latter approach for the PFC evaluation in Section 2.7.

Given the discrete-time analytical signal  $y_a(t) = y(t) + j\hat{y}(t)$ , the output signal of the PM, FM, and FS filters can be calculated by modulating  $y_a(t)$  with the LPTV frequency response  $H(\omega, t)$ , and then taking the real part (denoted with  $\text{Re}\{\cdot\}$ ) [18], i.e.,

$$d(t) = \text{Re}\{y_a(t)H(\omega, t)\}. \quad (2.32)$$

Using (2.24) and (2.26), this leads to

$$d(t) = y(t) \cos \phi(t) - \hat{y}(t) \sin \phi(t) \quad (2.33)$$

with

$$\phi(t) = \begin{cases} \beta \sin \omega_m t & \text{for sinusoidal PM} \\ \frac{\Delta_f}{f_m} \sin \omega_m t & \text{for sinusoidal FM} \\ \omega_m t & \text{for FS.} \end{cases} \quad (2.34)$$

$$\quad (2.35)$$

$$\quad (2.36)$$

A sinusoidal DM filter can be realized by directly operating on the microphone signal  $y(t)$ , which is then fed to a variable-length delay line. Such delay lines have also been used for realizing DM-based digital audio effects such as vibrato, flanging, and chorus, see, e.g., [111]-[113]. The sinusoidal DM variable-length delay line has an LPTV transfer function that can be approximated as the cascade of an integer delay of  $K$  samples and a fractional delay of  $l/D$  samples [111]-[113], where  $D$  is denoted as the interpolation ratio and  $l = 0, \dots, D - 1$  is the fractional phase,

$$H(q, t) = q^{-(\tau_0 + \Delta_\tau \sin \omega_m t)} \quad (2.37)$$

$$\approx q^{-K} q^{-l/D}. \quad (2.38)$$

The fractional part of the transfer function in (2.38) can be realized using any of the available methods for fractional delay filter design [114], e.g., using linear

[111]-[113], all-pass [111]-[113], or spline [112],[113] interpolation filters. We will use a linear FIR interpolation filter that is a Hamming-windowed, truncated (length- $2I$ ) approximation of the ideal sinc-like interpolation filter [114],

$$H(q, t) = q^{-K} \sum_{i=-I}^{I-1} w_h(i + l/D) \text{sinc}(i + l/D) q^i \quad (2.39)$$

where  $w_h(t)$  denotes the Hamming window, centered at  $t = 0$ , and the integer delay and fractional phase are chosen as  $K = \lfloor \tau_0 + \Delta_\tau \sin \omega_m t \rfloor$  and  $l = \lceil (\tau_0 + \Delta_\tau \sin \omega_m t - K)/D \rceil$ , with  $\lfloor \cdot \rfloor$  the floor function and  $\lceil \cdot \rceil$  the nearest integer function, respectively. Note that  $\tau_0$ ,  $\Delta_\tau$ , and  $I$  should be chosen such that  $\tau_0 - \Delta_\tau \geq I - 1$  to guarantee causality of the sinusoidal DM filter.

### 2.4.3 Discussion

The main strength of the PFC approach is its simplicity, both conceptually and computationally. The design of a PFC system requires little effort, since only the modulation technique (PM, FM, FS, or DM) and a few parameter values have to be decided on. The main computational load lies in the calculation of the analytical microphone signal (for PM, FM, and FS) and the fractional delay interpolation filtering (for DM), which should not be a barrier for real-time implementation. Moreover, the PFC approach does not involve any form of learning or adaptivity, such that it behaves in a completely deterministic way, which is beneficial in terms of robustness.

The choice of the modulation technique depends on the envisaged application. The FS technique is known to generally deliver a larger MSG increase than the other modulation techniques, but is perceptually inappropriate for music applications [18],[82]. The MSG increase obtained with modulation techniques that have a larger number of sideband responses (PM, FM, and DM, with a sufficiently large  $\beta$ ) appears to be more or less independent of the modulation frequency  $\omega_m$ , such that these techniques can operate at a lower value of  $\omega_m$  as compared to FS, which is perceptually advantageous [17]. DM is known to perform poorly at low signal frequencies, such that it should preferably be combined with another modulation technique or even with a non-phase-modulation-based acoustic feedback control method [16].

For a given modulation technique, the main parameters determining the PFC performance are the modulation frequency  $\omega_m$  and the modulation index  $\beta$ . It has been theoretically shown and experimentally verified that in the case of FS, an optimal value of the frequency shift  $f_m = \omega_m (f_s / 2\pi)$  is around  $4/T_{60}$  Hz, with  $T_{60}$  the room reverberation time in seconds [2]. The optimal value for  $f_m$  is less related to the reverberation time in the case of PM, FM, and DM, and values as low as 0.5 Hz may provide a satisfactory MSG increase, especially at high modulation index values [17]. The influence of the modulation index  $\beta$

in the case of PM, FM, and DM is governed by two effects [17]: as the value of  $\beta$  approaches the zeros of  $J_0(\beta)$  (e.g., see Fig. 2.6), the CS and hence the maximum achievable MSG increase become larger, and on the other hand, a larger value of  $\beta$  leads to a larger number of influential sideband responses which (at least in the continuous-time case) can be expected to improve the acoustic feedback control performance [17]. The former effect provides an explanation for the value of  $\beta = 2.4$  having been suggested as an optimal choice in early studies on PFC using sinusoidal PM [5],[14].

Finally, the PFC method has three major drawbacks. First of all, the achievable MSG increase is limited. An MSG increase of 12 dB has been found to be the theoretical maximum using FS in a typical room acoustic sound reinforcement system, and moreover, to avoid the FS effect to be clearly audible, a system equipped with a FS filter should operate 6 dB below the MSG, reducing the practically realizable MSG increase to 6 dB [2]. Similar MSG increase values (around 6 dB) were found in experiments using the other modulation techniques (PM, FM, and DM), as reported in several studies [14]-[17],[115]. A second drawback is that inserting a PM filter in the electro-acoustic forward path unavoidably leads to signal distortion, the perceptual consequences of which have not been objectively quantified in any of the reports dealing with PFC. A third disadvantage is the fact that in multi-channel systems, the stability improvement obtained with PFC has been shown to decrease as the number of channels increases [18], hence the practical use of PFC in large-scale sound reinforcement systems (e.g., PA or RE systems) is expected to be limited.

## 2.5 Notch-Filter-Based Howling Suppression

### 2.5.1 Concept

The objective of the NHS method can be either to prevent the closed-loop system from becoming unstable by reducing the loop gain  $|G(\omega, t)F(\omega, t)|$  in the neighborhood of critical frequencies, or to stabilize the system and suppress howling after oscillations have occurred. The former objective requires a proactive approach to instability detection, while the latter approach is reactive in the sense that notch filters are activated only after the detection of howling. We will mainly focus on the reactive approach to NHS, which is much more widespread than the proactive approach. Also, the emphasis is on two-stage NHS methods, since these are much more popular as compared to the ANF-based one-stage NHS methods. In a two-stage NHS method, the microphone signal  $y(t)$  is first processed by a howling detection algorithm, which forwards a set of design parameters  $\mathcal{D}_H(t)$  to a bank of adjustable notch filters  $H(q, t)$  that is inserted in the electro-acoustic forward path, see Fig. 2.7.

The howling detection algorithm is the most critical part of the two-stage NHS

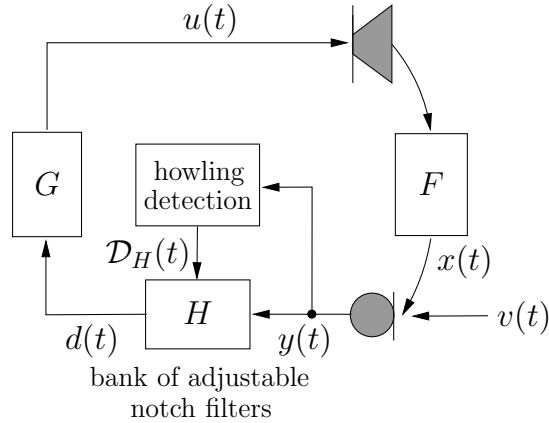


Figure 2.7: Two-stage notch-filter-based howling suppression (NHS) by feeding the microphone signal to a howling detection algorithm, which forwards a set of design parameters  $\mathcal{D}_H(t)$  to a bank of adjustable notch filters  $H(q, t)$  that is inserted in the electro-acoustic forward path.

method. Since howling is known to consist of sinusoidal signal components, the detection of howling is based on a frequency analysis of the microphone signal. It can be understood that howling components can be recognized as signal components having a large magnitude in the frequency domain. However, voiced speech components and tonal music components also have this property, hence it is crucial to discriminate howling components from tonal source signal components. We will use an example to illustrate the signal attributes that can be used to discriminate between howling and tonal components. Let us consider a single-channel closed-loop system defined by the acoustic feedback path shown in Figs. 2.3-2.4, and an electro-acoustic forward path consisting of a cascade of a unit delay and a broadband gain factor  $K = 5.53$  dB. The loop gain of this system is shown in Fig. 2.8(a) for  $f \in [0, 3]$  kHz. It can be observed that the Nyquist magnitude condition (2.11) is fulfilled for a frequency value just above 500 Hz, such that an oscillation at this frequency can be expected. When an audio signal fragment, more specifically a 10 s excerpt from the Partita No. 2 in D minor (Allemande) for solo violin by J. S. Bach, is applied as a source signal in the closed-loop system, the corresponding microphone signal has a spectrogram as shown in Fig. 2.8(b) (zooming in on the frequency region  $f \in [0, 3]$  kHz). The build-up of a howling component at a frequency slightly above 500 Hz is clearly visible from the spectrogram. Moreover, it can be observed that the howling component has some distinct features that may be used to discriminate it from the tonal source signal components. Spectral features include the fact that the howling component has a relatively large magnitude, and does not have any harmonic or subharmonic frequency components. Temporal features typical to the howling component are its long duration and its increasing magnitude with

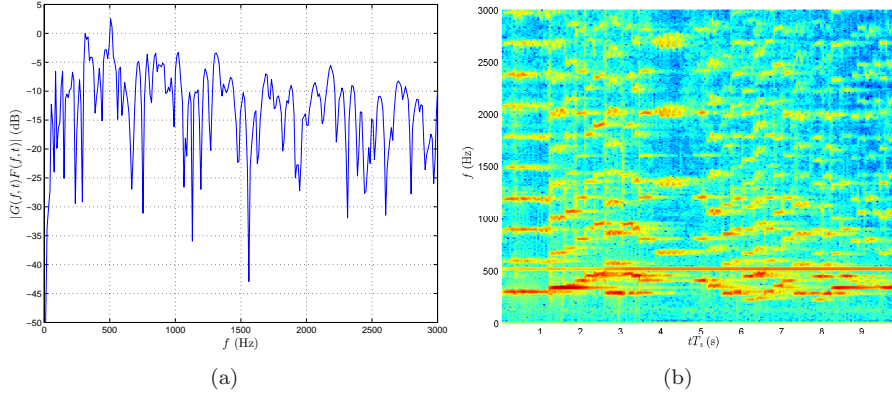


Figure 2.8: Example to recognize discriminating features between howling and tonal components (zooming in on the frequency region  $f \in [0, 3]$  kHz): (a) Loop gain of the unstable closed-loop system defined by the acoustic feedback path response shown in Fig. 2.4 and a flat electro-acoustic forward path response with gain factor  $K = 5.53$  dB, (b) Microphone signal spectrogram after feeding an audio source signal to the unstable closed-loop system.

time.

Apart from detecting howling components in the microphone signal spectrum, the howling detection algorithm in the two-stage NHS method shown in Fig. 2.7 also calculates some features of the detected howling components that are subsequently used to design appropriate notch filters. The set of design parameters  $\mathcal{D}_H(t)$  typically includes the howling components' frequency and magnitude values. The notch filters are then designed to have center frequencies corresponding to the howling component frequencies and notch depth values depending on the howling component magnitude values. The notch filters' 3 dB bandwidth is usually fixed to a value in the range of 1/10–1/60 octave. A more narrowband notch filter has the advantage of removing less of the desired source signal components, but requires a more accurate howling component frequency estimation.

## 2.5.2 Realization

### Howling Detection

We assume that the howling detection algorithm operates in a frame-based manner, on microphone signal frames with a frame length of  $M$  samples and a frame hop size of  $P$  samples (i.e., a frame overlap of  $M - P$  samples). At time  $t$ , the data in the microphone signal frame can then be represented by the vector (which is not to be confused with the multi-channel microphone signal

vector  $\bar{\mathbf{y}}(t)$  defined in (2.4))

$$\mathbf{y}(t) = [y(t+P-M) \quad \dots \quad y(t+P-1)]^T \quad (2.40)$$

and the short-term microphone signal spectrum can be obtained as the DFT of the data in  $\mathbf{y}(t)$ , i.e.,

$$Y(\omega_k, t) = \sum_{n=0}^{M-1} w(t_n)y(t_n)e^{-j\omega_k t_n}, \quad k = 0, \dots, M-1 \quad (2.41)$$

with  $\omega_k \triangleq 2\pi k/M$  and  $t_n \triangleq t+P-M+n$ . The microphone signal DFT in (2.41) is generally calculated using the FFT algorithm, and includes a window function  $w(t_n)$  to reduce the spectral leakage [116] (e.g., a Blackman window has successfully been applied to audio signal processing [117]<sup>6</sup>). Alternatively, a parametric frequency estimation method may be applied instead of the non-parametric (DFT-based) approach to obtain a good frequency resolution with relatively short signal frames [40]. Also, a frequency-warped DFT [118] may be used to improve the frequency resolution in the low-frequency region [48],[49]. The choice of the signal framing parameters  $M$  and  $P$  has a rather profound influence on the performance of the howling detection. Small values for the frame length  $M$  have been proposed to allow for very quick howling detection (e.g.,  $M = 128$ , corresponding to 4 ms at  $f_s = 32$  kHz [27]-[29]), such that howling may potentially be detected before it is actually perceived [27]-[29]. On the other hand, larger values for  $M$  provide a better frequency resolution in the microphone signal DFT spectrum estimate (e.g.,  $M = 4096$ , corresponding to 92.9 ms at  $f_s = 44.1$  kHz [31],[33] or to 85.3 ms at  $f_s = 48$  kHz [43]), which is necessary when working with very narrowband notch filters such as the 1/60 octave filters used in [48],[49]. A large frame hop size  $P$  may result in a large time lag between howling detection and notch filtering, unless a  $P$ -sample delay is inserted in the electro-acoustic forward path. On the other hand, a small value for  $P$  leads to an increase in computational complexity since the howling detection algorithm is then executed more often. Generally, a 25–50 % frame overlap ( $P = 3M/4, \dots, M/2$ ) is found to be a good compromise.

Based on the DFT-based microphone signal spectrum estimation, a pre-defined number  $N$  of spectral peaks is identified from the spectrum estimate, with  $N$  typically chosen in the range 1–10. These  $N$  frequency components are termed “candidate howling components” and their radial frequency values are collected in the set  $\mathcal{D}_{\tilde{\omega}}(t) = \{\tilde{\omega}_i\}_{i=1}^N$ . A spectral peak picking algorithm is usually applied to find the candidate howling components. A more advanced approach consists in selecting frequency components that have a consistently increasing magnitude in successive signal frames. This is possible by applying a so-called “ballistics” procedure [43] before executing the peak picking algorithm. The following spectral and temporal features of the microphone signal have been

<sup>6</sup>[http://ccrma.stanford.edu/~jos/mdft/Use\\_Blackman\\_Window.html](http://ccrma.stanford.edu/~jos/mdft/Use_Blackman_Window.html)

proposed to determine whether a candidate howling component indeed corresponds to a howling component or rather to a source signal tonal component:

- The Peak-to-Threshold Power Ratio (PTPR) [34],[35],[43] is a spectral feature that determines the ratio of the candidate howling component power  $|Y(\check{\omega}_i, t)|^2$  and a fixed absolute power threshold  $P_0$ , i.e.,

$$\text{PTPR}(\check{\omega}_i, t) \text{ [dB]} = 10 \log_{10} \frac{|Y(\check{\omega}_i, t)|^2}{P_0}. \quad (2.42)$$

Howling is detected at the frequency  $\check{\omega}_i$  if  $\text{PTPR}(\check{\omega}_i, t) \geq 0$  dB. The rationale behind using the PTPR for howling detection is that howling should only be suppressed when it appears with a minimum loudness [43]. The absolute power threshold  $P_0$  depends on the particular sound reinforcement scenario at hand, e.g., a value of  $10 \log_{10} P_0 = 85$  dB SPL was suggested in [43] for a loudspeaker-microphone distance of 1 m.

- The Peak-to-Average Power Ratio (PAPR) [24],[25],[34]-[36],[42],[43],[47]-[49],[53] is a spectral feature that determines the ratio of the candidate howling component power  $|Y(\check{\omega}_i, t)|^2$  and the average microphone signal power  $\hat{P}_y(t)$ , i.e.,

$$\text{PAPR}(\check{\omega}_i, t) \text{ [dB]} = 10 \log_{10} \frac{|Y(\check{\omega}_i, t)|^2}{\hat{P}_y(t)} \quad (2.43)$$

with

$$\hat{P}_y(t) = \frac{1}{M} \sum_{k=0}^{M-1} |Y(\omega_k, t)|^2. \quad (2.44)$$

The  $i$ th candidate howling component is identified as a howling component if the PAPR exceeds a predetermined threshold, i.e.,  $\text{PAPR}(\check{\omega}_i, t) \geq T_{\text{PAPR}}$ . The PAPR feature is probably the most widely used feature for howling detection, and different values for the threshold have been proposed, e.g.,  $T_{\text{PAPR}} = 6$  dB [24],[25],  $T_{\text{PAPR}} = 10 \log_{10}(M/150)^2$  dB [43], and  $T_{\text{PAPR}} = 10$  dB [47]. Kawamura *et al.* [36],[42] propose the use of a variable threshold  $T_{\text{PAPR}}(t)$  that is adapted online, based on estimates of the background noise spectrum, the source signal spectrum, the reverberation time, and the acoustic feedback path response. It is also suggested in [36],[42] to remove the  $Q_P$  largest frequency components from the spectrum  $Y(\omega_k, t)$  before estimating the average signal power  $\hat{P}_y(t)$  in (2.44), the value of  $Q_P$  depending on the bandwidth of the frequency analysis. Yet another way of estimating the average microphone signal power  $\hat{P}_y(t)$  was suggested in [53], i.e.,

$$\hat{P}_y(t) = 10 \log_{10} \left[ \left( \frac{1}{M} \sum_{k=0}^{M-1} |Y(\omega_k, t)|^2 \right) + 2 \sqrt{\frac{1}{M} \sum_{k=0}^{M-1} \left( |Y(\omega_k, t)|^2 - \frac{1}{M} \sum_{m=0}^{M-1} |Y(\omega_m, t)|^2 \right)^2} \right] \quad (2.45)$$

which should be particularly useful when the source signal has a Gaussian probability density function (PDF).

- The Peak-to-Harmonic Power Ratio (PHPR)[31],[33]-[35] is a spectral feature that determines the ratio of the candidate howling component power  $|Y(\check{\omega}_i, t)|^2$  and its  $m$ th (sub-)harmonic component power  $|Y(m\check{\omega}_i, t)|^2$ , i.e.,

$$\text{PHPR}(\check{\omega}_i, t, m) \text{ [dB]} = 10 \log_{10} \frac{|Y(\check{\omega}_i, t)|^2}{|Y(m\check{\omega}_i, t)|^2}. \quad (2.46)$$

In [31],[33], howling is detected at the frequency  $\check{\omega}_i$  if the PHPR exceeds a predetermined threshold for the 2nd, 3rd, and 4th harmonics and the 0.5th and 1.5th subharmonics, i.e., if

$$\bigcap_{m \in \{0.5, 1.5, 2, 3, 4\}} [\text{PHPR}(\check{\omega}_i, t, m) \geq T_{\text{PHPR}}] = 1 \quad (2.47)$$

with  $T_{\text{PHPR}} = 33$  dB. In [34],[35], a simpler howling detection criterion  $\text{PHPR}(\check{\omega}_i, t, 2) \geq T_{\text{PHPR}}$  is used.

- The Peak-to-Neighboring Power Ratio (PNPR) [21],[27]-[29] is a spectral feature that determines the ratio of the candidate howling component power  $|Y(\check{\omega}_i, t)|^2$  and its  $m$ th neighboring frequency component power  $|Y(\check{\omega}_i + 2\pi m/M, t)|^2$ , i.e.,

$$\text{PNPR}(\check{\omega}_i, t, m) \text{ [dB]} = 10 \log_{10} \frac{|Y(\check{\omega}_i, t)|^2}{|Y(\check{\omega}_i + 2\pi m/M, t)|^2}. \quad (2.48)$$

In [21],  $\check{\omega}_i$  is determined to be a howling frequency if the PNPR in two adjacent frequency bins on either side of the candidate howling component is consistently above two predetermined thresholds and the PTPR is above 0 dB, i.e., if

$$\left\{ [\text{PTPR}(\check{\omega}_i, t) \geq 0 \text{ dB}] \wedge \bigcap_{m \in \{\pm 1, \pm 2\}} [\text{PNPR}(\check{\omega}_i, t, m) \geq T_{\text{PNPR}}(|m|)] \right\} = 1. \quad (2.49)$$

In [27]-[29], howling is detected based on a so-called “peakness” feature, which reflects the time-averaged probability (over 8 signal frames) that



the PNPR, averaged over 6 neighboring frequency bins on either side of  $\check{\omega}_i$  (excluding the closest neighbor on either side), exceeds a 15 dB threshold, i.e.,

$$\text{peakness}(\check{\omega}_i, t) = \sum_{j=0}^7 \frac{1}{16} \left\{ \left[ \frac{1}{6} \sum_{m=2}^7 \text{PNPR}(\check{\omega}_i, t - jP, m) \geq 15 \text{ dB} \right] + \left[ \frac{1}{6} \sum_{m=-7}^{-2} \text{PNPR}(\check{\omega}_i, t - jP, m) \geq 15 \text{ dB} \right] \right\}. \quad (2.50)$$

- The Interframe Peak Magnitude Persistence (IPMP) [19],[20],[31],[33],[36],[42],[47] is a temporal feature based on counting in how many frames out of  $Q_M$  past signal frames the frequency  $\check{\omega}_i$  is in the set of candidate howling frequencies, i.e.,

$$\text{IPMP}(\check{\omega}_i, t) = \frac{\sum_{j=0}^{Q_M-1} [\check{\omega}_i \in \mathcal{D}_{\check{\omega}}(t - jP)]}{Q_M}. \quad (2.51)$$

Howling is usually detected if  $\text{IPMP}(\check{\omega}_i, t) = 1$  [19],[20],[36],[42],[47], with, e.g.,  $Q_M = 3$  [47]. In [31],[33], a howling detection criterion  $\text{IPMP}(\check{\omega}_i, t) \geq 3/5$  is proposed with  $Q_M = 5$ .

- The Interframe Magnitude Slope Deviation (IMSD) [27]-[29] is a temporal feature that determines the deviation (over  $Q_M$  successive signal frames) of the slope, which is defined by averaging magnitude difference values of a candidate howling component, where the differentiation is carried out between an old signal frame and more recent signal frames, i.e.,

$$\text{IMSD}(\check{\omega}_i, t) = \frac{1}{Q_M - 1} \sum_{m=1}^{Q_M-1} \left[ \frac{1}{Q_M} \sum_{j=0}^{Q_M-1} \frac{1}{Q_M - j} (20 \log_{10} |Y(\check{\omega}_i, t - jP)| - 20 \log_{10} |Y(\check{\omega}_i, t - Q_M P)|) - \frac{1}{m} \sum_{j=0}^{m-1} \frac{1}{m - j} (20 \log_{10} |Y(\check{\omega}_i, t - jP)| - 20 \log_{10} |Y(\check{\omega}_i, t - mP)|) \right]. \quad (2.52)$$

Small values for the IMSD are characteristic of howling components since these exhibit a nearly linear (dB-scale) magnitude increase in time, hence a nearly constant slope can be expected. A detection threshold of 0.05 has been proposed in [29], such that howling is detected when  $|\text{IMSD}(\check{\omega}_i, t)| \leq 0.05$ , with  $Q_M = 7$ .

The complete howling detection algorithm is summarized in Fig. 2.9. Obviously, any combination of the above spectral and temporal features may be

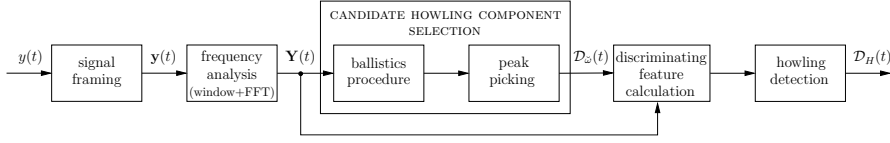


Figure 2.9: Howling detection algorithm for two-stage NHS method: from the microphone signal  $y(t)$ , a set of notch filter design parameters  $\mathcal{D}_H(t)$  is calculated.

used to discriminate between howling and tonal components. In most of the existing NHS methods, at least one spectral and one temporal feature are taken into account for detecting howling.

### Notch Filtering

When howling has been detected, a notch filter has to be activated to suppress the howling component and stabilize the closed-loop system. The most commonly used notch filter structure in NHS is the second-order IIR (i.e., bi-quadratic) filter structure,

$$H_l(q, t) = \frac{b_l^{(0)}(t) + b_l^{(1)}(t)q^{-1} + b_l^{(2)}(t)q^{-2}}{1 + a_l^{(1)}(t)q^{-1} + a_l^{(2)}(t)q^{-2}}. \quad (2.53)$$

The bank of adjustable notch filters that is inserted in the electro-acoustic forward path, as shown in Fig. 2.7, then consists of a cascade of  $n_H/2$  such filters, i.e.,

$$H(q, t) = \prod_{l=1}^{n_H/2} H_l(q, t) \quad (2.54)$$

with  $n_H$  the resulting order of the cascade filter.

The notch filter design procedure consists of two parts: first the set of design parameters  $\mathcal{D}_H(t)$  delivered by the howling detection algorithm has to be mapped to a set of filter specifications, which are then translated into filter coefficient values. A biquadratic notch filter has five coefficients, which depend on a set of six filter specifications [119]: the (radial) center frequency  $\omega_{c,l}$ , the (radial) bandwidth  $B_l$ , the notch gain  $G_{c,l}$ , the gain at the band edges  $G_{B,l}$ , the gain at DC  $G_{0,l}$ , and the gain at the Nyquist frequency  $G_{\pi,l}$ . If we fix the latter two variables to  $G_{0,l} = G_{\pi,l} = 0$  dB and the gain at the band edges to  $G_{B,l} = G_{c,l} + 3$  dB in case  $G_{c,l} \leq -6$  dB, or to  $G_{B,l} = G_{c,l}/2$  in case  $G_{c,l} \geq -6$  dB (thereby adopting Moorers bandwidth definition [120]), then only the first three filter specifications remain.

The set of design parameters  $\mathcal{D}_H(t)$  should always contain the radial frequencies  $\{\check{\omega}_i\}_{i \in \mathcal{I}_H(t)}$  of the howling components that have been identified in the howling

detection algorithm, where  $\mathcal{I}_H(t) \subseteq \{1, \dots, N\}$  denotes the set of indices for which howling has been detected. For each howling component, a notch filter should be activated, with a center frequency corresponding to the howling frequency. It is desirable to compensate for the limited frequency resolution of the microphone signal DFT by linearly interpolating the notch filter center frequency, using the DFT information from frequency bins adjacent to the identified howling component [43], e.g.,

$$\omega_{c,l} = \check{\omega}_i + \frac{2\pi}{M} \left( \frac{|Y(\check{\omega}_i + 2\pi/M)| - |Y(\check{\omega}_i - 2\pi/M)|}{|Y(\check{\omega}_i - 2\pi/M)| + |Y(\check{\omega}_i)| + |Y(\check{\omega}_i + 2\pi/M)|} \right). \quad (2.55)$$

In this case, the DFT magnitude values  $|Y(\check{\omega}_i - 2\pi/M)|$ ,  $|Y(\check{\omega}_i)|$ , and  $|Y(\check{\omega}_i + 2\pi/M)|$  should also appear in the set of design parameters  $\mathcal{D}_H(t)$ . The DFT magnitude information may also be used to determine the notch gain  $G_{c,l}$ , however, it is common practice to work with fixed notch gain values that are independent of the howling component magnitude. Typically, when a new howling component has been detected (i.e., a howling component at a frequency that has not occurred before), the notch gain is set to an initial value  $G_{c,l}^{(0)}$ , e.g.,  $G_{c,l}^{(0)} = -3$  dB [31],[33] or  $G_{c,l}^{(0)} = -6$  dB [43]. If howling persists or reoccurs at a frequency close to a previously identified howling frequency, then the gain is decreased with  $\Delta G_{c,l}$  dB, e.g.,  $\Delta G_{c,l} = -3$  dB [31],[33] or  $\Delta G_{c,l} = -6$  dB [43]. Finally, the radial notch filter bandwidth  $B_l$  is usually chosen proportional to the center frequency, such that the filter has a constant Q factor. The octave bandwidth is then also constant and is typically chosen in the range 1/10–1/60 octave, e.g., 1/10 octave [31],[33],[50],[99], 1/20 octave [50],[99], or 1/60 octave [48],[49].

Finally, the filter specifications  $\mathcal{S}_{H_l}(t) = \{\omega_{c,l}, B_l, G_{c,l}\}$  have to be translated to a set of filter coefficients  $\mathcal{C}_{H_l}(t) = \{b_l^{(0)}(t), b_l^{(1)}(t), b_l^{(2)}(t), a_l^{(1)}(t), a_l^{(2)}(t)\}$ . Most notch filter design methods are based on a bilinear transform of either an analog notch filter transfer function [121]–[127], or a digital notch filter transfer function centered at  $\omega_c = \pi/2$  [120]. A novel design procedure for biquadratic notch filters was recently proposed, that operates directly in the digital domain using a technique known as pole-zero placement [119]. This design procedure, which is equally accurate yet more intuitive than the bilinear-transform-based design methods, will be applied in the evaluation of the NHS method in Section 2.7. The complete notch filter design procedure for the two-stage NHS method is shown schematically in Fig. 2.10.

### 2.5.3 Initialization

In the PFC method for acoustic feedback control, the optimal values for the algorithm parameters (i.e., the modulation frequency  $\omega_m$  and modulation index  $\beta$ ) were found to be independent of the specific acoustic feedback path

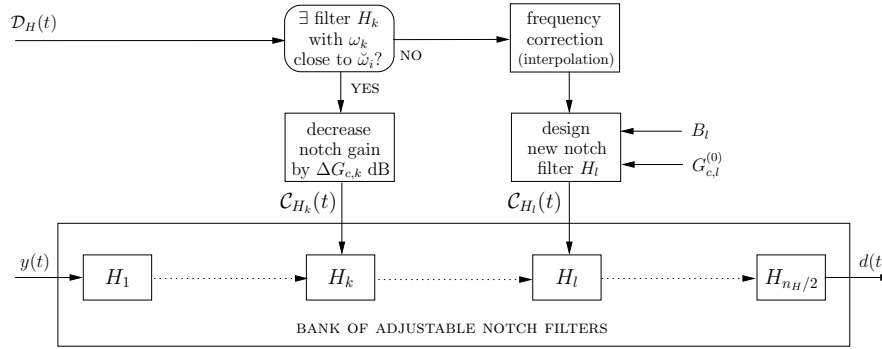


Figure 2.10: Notch filter algorithm for two-stage NHS method: the microphone signal  $y(t)$  is filtered in a bank of adjustable notch filters, designed using the design parameters in  $\mathcal{D}_H(t)$ , resulting in the howling-compensated signal  $d(t)$ .

characteristics. The optimal notch filter coefficients in the NHS method, however, depend heavily on the spectral properties of the acoustic feedback path. It has long been known that some of the spectral peaks in the acoustic feedback path magnitude response  $|F(\omega, t)|$  originate from reflections depending on the room boundaries only and are hence independent of the position of loudspeakers, microphones, and other objects in the room<sup>7</sup>. For this reason, manual equalization and notch filtering is largely performed during initialization (e.g., “ringing out” a PA system during sound check [129]) and fixed filters are applied to compensate for the major room resonances.

Similarly, a number of notch filters in the NHS method may be fixed to the so-called “eigenfrequencies” of the room, while the remaining notch filters can be adjusted to suppress variable-frequency howling components, which are due to, e.g., microphone movements [31],[33],[40]. The fixed notch filter design parameters should then be determined during the initialization of the sound reinforcement system, which is usually accomplished by feeding a white noise signal to the loudspeakers at a relatively high amplifier gain and subsequently identifying persisting spectral components in the microphone signal [22],[23],[31],[33].

The variable notch filters differ from the fixed notch filters in that they can be activated and de-activated during normal operation of the sound reinforcement system. While an extensive part of the NHS literature is devoted to strategies for the activation of these notch filters (i.e., after howling detection), hardly any research results are available dealing with the criteria for notch filter de-activation. One such de-activation criterion was proposed by Terada and Murase [26] in the context of AEQ for HA applications, and consists in

<sup>7</sup>This concept has been formalized in the so-called common acoustical pole and zero (CAPZ) model for room transfer functions [128].

de-activating the equalization filters after a time period that is inversely proportional to the time period between two successive occurrences of howling. Finally, we should note that the activation of a notch filter in the electro-acoustic forward path leads to transient components in the loudspeaker signal, which may be perceived as short-lived ringing artifacts [34],[35]. This effect can be avoided by gradually activating and de-activating the notch filters, e.g., using a leaky integrator [53].

#### 2.5.4 Discussion

The NHS approach has many strengths, the most important one being its robustness. Unlike other acoustic feedback control methods, NHS methods have the powerful property of being able to stabilize an unstable system without having to reduce the broadband gain. For this reason, it is advisable that a sound reinforcement system that is operated with a different acoustic feedback control method (e.g., PFC or AFC) be supplemented with an NHS method, which should then be activated when the system stability cannot be restored using the PFC or AFC method. As for computational requirements, the NHS approach has a moderate complexity, in between the cheap PFC approach and the expensive AFC approach. The main computational load is in the frequency analysis and can be governed by properly choosing the frame length  $M$  and hop size  $P$ . Another attractive property is that the extension of the NHS approach to multi-channel systems is relatively straightforward. In the multi-channel case, it is usually more efficient to have the howling detection and notch filtering algorithms operate on the mixed signals instead of on the microphone signals, since the number of channels is usually reduced after mixing. Alternatively, both the mixed signals and the individual microphone signals can be used for howling detection, where the latter may be analyzed at a lower frequency resolution [51].

A difficulty that arises when applying an NHS method for acoustic feedback control, is the multitude of algorithm parameters that have to be set, namely the frame length and hop size, the number of candidate howling components selected in each signal frame, the combination of discriminating features, the thresholds for howling detection, the number of fixed/variable notch filters to use, ... Unfortunately, few guidelines are available for setting these algorithm parameters. As many NHS methods are described in patents, very few experimental results and no true comparisons between different NHS methods are available. A comparison of three NHS methods with particular choices for the algorithm parameters will be provided in Section 2.7, but obviously, many more combinations are possible.

The major shortcoming of the NHS approach is that it cannot deliver an MSG increase that is substantially larger than the MSG increase obtained with the PFC approach. At most, i.e., when all the spectral peaks in the loop gain

could be removed, an MSG increase of 10 dB could be expected based on the statistical analysis by Schroeder [2]. In practice, this maximum value will never be attained since it is nearly impossible to completely flatten the loop gain and still retain an acceptable degree of sound quality. As an example, if we would increase the gain in the single-channel system associated with the acoustic feedback path shown in Fig. 2.4(a) to a value that is 10 dB above the MSG without acoustic feedback control, then over 20 frequencies would satisfy the magnitude condition (2.11) in the Nyquist criterion, most of these lying in the 100–1500 Hz frequency region. Applying the NHS approach would then lead to a broadband attenuation in the 100–1500 Hz band, which would be detrimental for the sound quality (e.g., speech intelligibility). The limited achievable increase in MSG is also observed in manual notch filtering methods, where values of 5 to 8 dB have been obtained [6]. Finally, in terms of sound quality, the signal distortion due to notch filtering is reasonable if the number of filters that are applied concurrently is small and if the notch filter bandwidths are small. In fact, the main decrease in sound quality is due to the reactive nature of most NHS methods, i.e., howling can usually be perceived before it can be suppressed. From this point of view, proactive NHS methods can be viewed as promising acoustic feedback control solutions (see, e.g., [50],[56]), however, their current applicability is limited due to their high computational complexity, comparable to the AFC complexity.

## 2.6 Adaptive Feedback Cancellation

### 2.6.1 Concept

In a sound reinforcement system, the microphone signal  $y(t)$  consists of a source signal component  $v(t)$  and a feedback signal component  $x(t)$ , the latter denoting the entire signal that is fed back from the loudspeaker to the microphone. The AFC approach to acoustic feedback control is aimed at predicting the feedback signal component and then subtracting this prediction from the microphone signal. The predicted feedback signal, denoted as  $\hat{y}[t|\hat{\mathbf{f}}(t)]$ , is obtained by filtering the loudspeaker signal  $u(t)$  with a model  $\hat{F}(q, t)$  of the acoustic feedback path, see Fig. 2.11. This model is calculated using an adaptive filter, that is designed to identify the feedback path impulse response  $\mathbf{f}(t)$  and track its changes. The feedback path and adaptive filter impulse responses are defined at time  $t$  as

$$\mathbf{f}(t) = [f^{(0)}(t) \quad f^{(1)}(t) \quad \dots \quad f^{(n_F)}(t)] \quad (2.56)$$

$$\hat{\mathbf{f}}(t) = [\hat{f}^{(0)}(t) \quad \hat{f}^{(1)}(t) \quad \dots \quad \hat{f}^{(n_{\hat{F}})}(t)] \quad (2.57)$$

respectively.

The closed-loop frequency response of the system shown in Fig. 2.11, employing

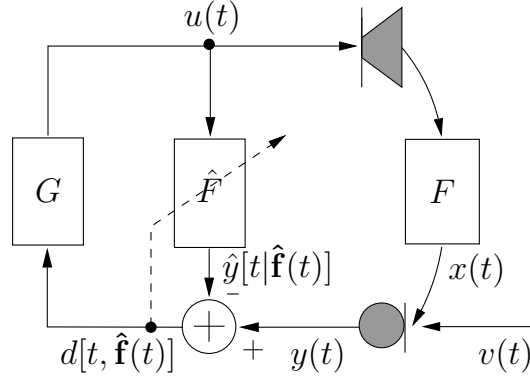


Figure 2.11: Adaptive feedback cancellation (AFC) by predicting the feedback signal component  $x(t)$  in the microphone signal, and hence subtracting the prediction  $\hat{y}[t|\hat{\mathbf{f}}(t)]$  from the microphone signal  $y(t)$ . The prediction is obtained by filtering the loudspeaker signal with a model  $\hat{F}(q, t)$  of the acoustic feedback path, which is calculated using an adaptive filter.

an AFC method, is given by

$$\frac{U(\omega, t)}{V(\omega, t)} = \frac{G(\omega, t)}{1 - G(\omega, t)[F(\omega, t) - \hat{F}(\omega, t)]} \quad (2.58)$$

and, as a consequence, the Nyquist stability criterion can be rewritten as follows,

$$\begin{cases} |G(\omega, t)[F(\omega, t) - \hat{F}(\omega, t)]| \geq 1 \\ \angle G(\omega, t)[F(\omega, t) - \hat{F}(\omega, t)] = n2\pi, \quad n \in \mathbb{Z} \end{cases} \quad (2.59)$$

$$\angle G(\omega, t)[F(\omega, t) - \hat{F}(\omega, t)] = n2\pi, \quad n \in \mathbb{Z} \quad (2.60)$$

which leads to the following expression for the MSG (see also (2.16)),

$$\text{MSG}(t) \text{ [dB]} = -20 \log_{10} \left[ \max_{\omega \in \mathcal{P}} |J(\omega, t)[F(\omega, t) - \hat{F}(\omega, t)]| \right]. \quad (2.61)$$

From (2.61), it immediately follows that the better the fit between the estimated and actual feedback path frequency response, particularly at critical frequencies of the closed-loop system, the larger the achievable MSG increase. Theoretically, if  $\hat{F}(q, t) \equiv F(q, t)$ , the system would no longer exhibit a closed signal loop and hence the MSG would be infinitely large.

While the concept of AFC is relatively simple and similar to the well-known acoustic echo cancellation (AEC) approach, its realization is not straightforward. In the identification of the acoustic feedback path model  $\hat{F}(q, t)$ , a fundamental problem appears which is due to the closed-loop nature of the system. The least-squares (LS) estimate  $\hat{\mathbf{f}}(t)$  of the acoustic feedback path impulse response  $\mathbf{f}(t)$  can straightforwardly be calculated as

$$\hat{\mathbf{f}}(t) = (\mathbf{U}^T \mathbf{U})^{-1} \mathbf{U}^T \mathbf{y} \quad (2.62)$$

where the data vectors and matrices are defined as follows (and where the loudspeaker signal vector  $\mathbf{u}(t)$  is not to be confused with the multi-channel loudspeaker signal vector  $\bar{\mathbf{u}}(t)$  defined in (2.5)),

$$\mathbf{y} = [y(t) \quad y(t-1) \quad \dots \quad y(1)]^T \quad (2.63)$$

$$\mathbf{U} = [\mathbf{u}(t) \quad \mathbf{u}(t-1) \quad \dots \quad \mathbf{u}(1)]^T \quad (2.64)$$

$$\mathbf{u}(t) = [u(t) \quad u(t-1) \quad \dots \quad u(t-n_{\hat{F}})]^T \quad (2.65)$$

The LS estimate may be characterized by its bias and variance [130, Ch. 8]. The bias corresponds to the difference between the expected value of the LS estimate and the true feedback path impulse response, i.e.,

$$\text{bias}\{\hat{\mathbf{f}}(t)\} = \begin{bmatrix} E\{\hat{\mathbf{f}}(t)\} \\ \mathbf{0}_{(n_F-n_{\hat{F}}) \times 1} \end{bmatrix} - \mathbf{f}(t) \quad (2.66)$$

where  $E\{\cdot\}$  denotes the expectation operator. Under a sufficient order assumption (i.e.,  $n_{\hat{F}} = n_F$ ), the expected value of the LS estimate can be shown to correspond to [77]

$$E\{\hat{\mathbf{f}}(t)\} = \mathbf{f}(t) + E\{(\mathbf{U}^T \mathbf{U})^{-1} \mathbf{U}^T \mathbf{v}\}. \quad (2.67)$$

The rightmost term in (2.67) can be understood to be generally non-zero due to the closed-loop nature of the system, which induces a correlation between the source signal and the loudspeaker signal, and hence

$$\text{bias}\{\hat{\mathbf{f}}(t)\} = E\{(\mathbf{U}^T \mathbf{U})^{-1} \mathbf{U}^T \mathbf{v}\} \neq \mathbf{0}. \quad (2.68)$$

The resulting effect in AFC is that the adaptive filter does not only predict and cancel the feedback component in the microphone signal, but also (part of) the source signal component. As a consequence, the feedback-compensated signal  $d[t, \hat{\mathbf{f}}(t)]$  is a distorted estimate of the source signal  $v(t)$ . On the other hand, the variance of the LS estimate can be obtained by considering its covariance matrix<sup>8</sup>, which is calculated as [131]

$$\text{cov}\{\hat{\mathbf{f}}(t)\} = E\{(\hat{\mathbf{f}}(t) - \mathbf{f}(t))(\hat{\mathbf{f}}(t) - \mathbf{f}(t))^T\} \quad (2.69)$$

$$= \left[ E\{\mathbf{U}^T \mathbf{R}_{\mathbf{v}}^{-1} \mathbf{U}\} \right]^{-1} \quad (2.70)$$

where the source signal covariance matrix  $\mathbf{R}_{\mathbf{v}}$  is defined as

$$\mathbf{R}_{\mathbf{v}} = E\{\mathbf{v}\mathbf{v}^T\} \quad (2.71)$$

<sup>8</sup>Note that the covariance matrix of the estimate  $\hat{\mathbf{f}}(t)$  is in fact defined as  $\text{cov}\{\hat{\mathbf{f}}(t)\} = E\{(\hat{\mathbf{f}}(t) - E\{\hat{\mathbf{f}}(t)\})(\hat{\mathbf{f}}(t) - E\{\hat{\mathbf{f}}(t)\})^T\}$ , which corresponds to  $\text{cov}\{\hat{\mathbf{f}}(t)\} = E\{(\hat{\mathbf{f}}(t) - \mathbf{f}(t))(\hat{\mathbf{f}}(t) - \mathbf{f}(t))^T\}$  if  $E\{\hat{\mathbf{f}}(t)\} = \mathbf{f}(t)$ , i.e., if the estimate is unbiased. However, in the analysis of closed-loop identification methods it has been found more meaningful to work directly with the covariance expression  $\text{cov}\{\hat{\mathbf{f}}(t)\} = E\{(\hat{\mathbf{f}}(t) - \mathbf{f}(t))(\hat{\mathbf{f}}(t) - \mathbf{f}(t))^T\}$  even if  $E\{\hat{\mathbf{f}}(t)\} \neq \mathbf{f}(t)$ , see, e.g., [131].



with

$$\mathbf{v} = [v(t) \quad v(t-1) \quad \dots \quad v(1)]^T. \quad (2.72)$$

The interpretation of (2.70) can be related to the double-talk problem occurring in AEC [132]. In AEC, when the loudspeaker signal is active while the source signal is not, the covariance matrix of the acoustic echo path LS estimate is relatively small, since  $\mathbf{R}_v \approx \mathbf{0}$ . However, when both signals are active at the same time (i.e., in a double-talk situation), the covariance matrix may become large, which may be observed in the adaptive filter performance as a decrease in convergence speed, or even a divergence. This problem becomes more severe as the source signal has a larger degree of coloration, since then the source signal covariance matrix  $\mathbf{R}_v$  exhibits a denser structure [132]. In AFC, the closed signal loop results in a continuous double-talk situation, and then this is made even worse by the correlation between the source and loudspeaker signal.

To prevent the adaptive filter from converging to a biased solution, and to increase its convergence speed despite the inevitable continuous double-talk situation, a decorrelation procedure is typically included in the AFC approach, with the aim of reducing the correlation between the source and loudspeaker signal. We can distinguish between two types of decorrelation [77], namely decorrelation in the closed signal loop and decorrelation in the adaptive filtering circuit. The former approach has the disadvantage of distorting the loudspeaker signal, while the latter approach requires somewhat more computations.

## 2.6.2 Realization

### Adaptive Filtering

The adaptive calculation of the LS estimate (2.62) of the acoustic feedback path impulse response, and the subsequent calculation of the feedback-compensated signal can be performed as follows,

$$\varepsilon[t, \hat{\mathbf{f}}(t-1)] = y(t) - \mathbf{u}^T(t) \hat{\mathbf{f}}(t-1) \quad (2.73)$$

$$\mathbf{R}(t) = \lambda \mathbf{R}(t-1) + \mathbf{u}(t) \mathbf{u}^T(t) \quad (2.74)$$

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mathbf{R}^{-1}(t) \mathbf{u}(t) \varepsilon[t, \hat{\mathbf{f}}(t-1)] \quad (2.75)$$

$$d[t, \hat{\mathbf{f}}(t)] = y(t) - \mathbf{u}^T(t) \hat{\mathbf{f}}(t) \quad (2.76)$$

The algorithm in (2.73)-(2.75) is known as the recursive least squares (RLS) algorithm [133, Ch. 13], with an exponential forgetting factor  $\lambda$ . Note that the (a priori) RLS residual  $\varepsilon[t, \hat{\mathbf{f}}(t-1)]$  in (2.73) differs from the (a posteriori) feedback-compensated signal  $d[t, \hat{\mathbf{f}}(t)]$ , which is subsequently processed in the electro-acoustic forward path, in that the former depends on the previous estimate  $\hat{\mathbf{f}}(t-1)$ . Through the application of the so-called matrix inversion lemma (MIL), the explicit inversion of the loudspeaker signal correlation matrix  $\mathbf{R}(t)$  in (2.75) can be avoided, and hence the RLS algorithm requires  $O(n_{\hat{\mathbf{f}}}^2)$  multiplications per time update [133, Ch. 13]. Fast RLS algorithms, requiring only

$O(n_{\hat{F}})$  multiplications per time update, have also been derived [134],[135] and successfully applied to, e.g., AEC [136]-[138, Ch. 6]. These fast RLS algorithms, however, are based on the shift invariance property of the loudspeaker signal vector defined in (2.65), i.e., the difference between  $\mathbf{u}(t-1)$  and  $\mathbf{u}(t)$  consists in discarding the oldest sample value  $u(t-1-n_{\hat{F}})$ , shifting the remaining sample values, and inserting the most recent sample value  $u(t)$  in the first position. This shift invariance property will generally not hold in the AFC context due to the decorrelation that is applied, either in the closed signal loop or in the adaptive filtering circuit, which often involves a nonlinear or time-varying filtering of the loudspeaker signal (see Section 2.6.2, ‘‘Decorrelation’’).

Alternatively, a computationally cheaper adaptive algorithm can be found in the underdetermined recursive least squares (URLS) family [139]. The affine projection algorithm (APA) provides an estimate of the acoustic feedback path impulse response by using only the  $M$  most recent microphone signal samples and loudspeaker signal vectors, i.e.,

$$\boldsymbol{\varepsilon}_M[t, \hat{\mathbf{f}}(t-1)] = \mathbf{y}_M(t) - \mathbf{U}_M^T(t) \hat{\mathbf{f}}(t-1) \quad (2.77)$$

$$\begin{aligned} \hat{\mathbf{f}}(t) &= \hat{\mathbf{f}}(t-1) + \mu \mathbf{U}_M(t) [\mathbf{U}_M^T(t) \mathbf{U}_M(t) + \alpha \mathbf{I}_M]^{-1} \boldsymbol{\varepsilon}_M[t, \hat{\mathbf{f}}(t-1)] \\ d[t, \hat{\mathbf{f}}(t)] &= y(t) - \mathbf{u}^T(t) \hat{\mathbf{f}}(t) \end{aligned} \quad (2.78)$$

where  $\mu$  represents the step size,  $\alpha \mathbf{I}_M$  is an  $M \times M$  identity regularization matrix scaled with the regularization parameter  $\alpha$ , and

$$\mathbf{y}_M(t) = [y(t) \quad \dots \quad y(t-M+1)]^T \quad (2.79)$$

$$\mathbf{U}_M(t) = [\mathbf{u}(t) \quad \dots \quad \mathbf{u}(t-M+1)] \quad (2.80)$$

The APA requires  $O(Mn_{\hat{F}})$  multiplications per time update, which is obviously much less than the RLS algorithm if the APA projection order  $M \ll n_{\hat{F}}$ . Since the APA can be shown to decorrelate input signals that admit an  $M$ th order all-pole representation [140], relatively small values for the projection order have been successfully used in AEC applications with speech signals (e.g.,  $M \in \{2, \dots, 50\}$  [141],  $M = 8$  [142],  $M = 10$  [143]). Fast versions of the APA, requiring only  $O(n_{\hat{F}}) + O(M)$  multiplications per time update, have also been derived [141]-[143], but again these algorithms rely on the shift invariance property of the loudspeaker signal vector, which need not be fulfilled in the AFC context.

Since the required adaptive filter order in room acoustic applications may be very high, a computational load of  $O(Mn_{\hat{F}})$  multiplications per time update may still be too heavy to allow for a real-time application. For this reason, in most of the existing AFC schemes a projection order  $M = 1$  is applied, which leads to the well-known normalized least mean squares (NLMS) algorithm,

$$\varepsilon[t, \hat{\mathbf{f}}(t-1)] = y(t) - \mathbf{u}^T(t) \hat{\mathbf{f}}(t-1) \quad (2.81)$$

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mu \frac{\mathbf{u}(t)\varepsilon(t)}{\mathbf{u}^T(t)\mathbf{u}(t) + \alpha} \quad (2.82)$$

$$d[t, \hat{\mathbf{f}}(t)] = y(t) - \mathbf{u}^T(t) \hat{\mathbf{f}}(t) \quad (2.83)$$

The required number of multiplications per time update is  $O(n_{\hat{F}})$ , more specifically  $4n_{\hat{F}} + 6$  (if the calculation of the feedback-compensated signal in (2.83) is also taken into account). The choice of the NLMS step size  $\mu$  is crucial to obtain a good compromise between a stable and fast convergence. In our own experience, step size values in the range  $\mu = 0.01 - 0.05$  are best suited for speech applications [97],[99],[101], while values around  $\mu = 0.005$  are recommended when working with audio signals [105].

Finally, the choice of the adaptive filter order  $n_{\hat{F}}$  is obviously extremely important, regardless of which adaptive filtering algorithm is used. It is clear that the choice of  $n_{\hat{F}}$  has a profound influence on the computational requirements of the AFC approach. One could argue that it may be sufficient to choose  $n_{\hat{F}}$  such that the largest components in the acoustic feedback path impulse response (originating from the early reflections) can be modeled. Unfortunately, such an approach would be inefficient for two reasons: firstly, large impulse response components do not necessarily correspond to large frequency response components and hence stability may not be improved by only cancelling the early reflections. Secondly, if the impulse response is undermodeled (i.e.,  $n_{\hat{F}} < n_F$ ) then an additional bias component will appear in the LS estimate (in addition to the bias due to the source and loudspeaker signal correlation) and moreover its variance will increase [144]. The best compromise between computational complexity and feedback control performance probably consists in choosing  $n_{\hat{F}}$  just large enough to obtain a satisfying MSG increase, and applying a technique for reducing the bias and variance due to undermodeling [144]-[146]. We should point out that the technique proposed by Rombouts *et al.* [144] for consistently identifying undermodeled room impulse responses is particularly interesting in the context of AFC, since it additionally provides a decorrelation in the adaptive filtering circuit.

We should emphasize that the above adaptive algorithms are often not implemented as such, since both the robustness and the efficiency of these algorithms can be further improved [99]. A robust adaptive filter implementation for AFC may include the following features: an adaptation control that freezes the adaptive filter coefficients during source signal onsets [99], a foreground/background adaptive filter implementation to combine good tracking properties with a small steady-state error [99], and a regularization method that compensates for the coloration of the loudspeaker signal [99],[101]. Moreover, the AFC efficiency in terms of computational load and convergence speed can be improved by considering a subband or frequency domain adaptive filter implementation rather than the time domain implementations shown here [99].

### Decorrelation

Decorrelation of the source and loudspeaker signals in the closed signal loop can be achieved by inserting a decorrelating signal operation in the electro-acoustic forward path, see Fig. 2.12. We can distinguish between the following approaches,

- Noise injection [60],[70],[79]-[82]: A white noise signal  $n(t)$  is added to the feedback-compensated signal after the electro-acoustic forward path processing (but usually before the forward path amplification), see Fig. 2.12(a), i.e.,

$$u(t) = K \left[ J(q, t) d[t, \hat{\mathbf{f}}(t)] + n(t) \right]. \quad (2.84)$$

The acoustic feedback path identification can then proceed in two ways: if the loudspeaker signal  $u(t)$  (including the noise signal  $n(t)$ ) is used as the input signal to the adaptive filter, then the effect of the noise injection is that the source and loudspeaker signal cross-correlation is decreased, hence the bias will be reduced but not completely eliminated. A second possibility is to use the noise signal  $n(t)$  as the adaptive filter input signal, which leads to an unbiased estimate since the source signal and the noise signal are uncorrelated. In this case, however, the adaptive filter convergence will be rather slow since not only the source signal, but also the source signal component in the feedback signal acts as a disturbance towards the feedback path identification. With the aim of reducing the influence of the noise injection on sound quality, the noise spectrum can be shaped such as to render the noise less perceptible, e.g., by A-weighting [80] or psychoacoustic noise shaping [81],[82]. Unfortunately, noise shaping decreases the decorrelation effect, making the noise injection less effective in removing the bias.

- Time-varying processing [70],[83]-[85]: Each of the LPTV filters described in Section 2.4 can be used as a decorrelating filter in the electro-acoustic forward path, see Fig. 2.12(b),

$$u(t) = G(q, t) \left[ H(q, t) d[t, \hat{\mathbf{f}}(t)] \right]. \quad (2.85)$$

An FS decorrelating filter has an LPTV frequency response as given by (2.26) and has successfully been applied to AFC for speech applications [83]-[85]. While the perceptible signal distortion introduced by the FS operation appears to be acceptable for speech signals [85], the FS decorrelation technique was found to be perceptually inadequate for audio applications [82]. The sinusoidal PM and DM filters, of which the LPTV frequency response is given in (2.24) and (2.29) respectively, have also been applied as decorrelation filters in AFC [70],[83],[84]. Note that a beneficial side effect of using LPTV decorrelation filters is that these filters also stabilize the closed-loop system by smoothing the loop gain (see Section 2.4).

- Nonlinear processing [70],[87]: In the context of stereo AEC, the correlation between the stereo channels, which leads to an identifiability problem of the acoustic echo path models [147], has been reduced by applying nonlinear decorrelating operations to the loudspeaker signals [86]. These nonlinear operations can also be used to reduce the correlation between the source and loudspeaker signals in an AFC application. In particular, half-wave rectification has been successfully applied to AFC decorrelation [70],[87], see Fig. 2.12(b), i.e.,

$$u(t) = G(q, t) \left[ H(d[t, \hat{\mathbf{f}}(t)], t) \right] \quad (2.86)$$

$$= G(q, t) \left[ d[t, \hat{\mathbf{f}}(t)] + \alpha \left( \frac{d[t, \hat{\mathbf{f}}(t)] + |d[t, \hat{\mathbf{f}}(t)]|}{2} \right) \right] \quad (2.87)$$

The parameter  $\alpha$  can be tuned to obtain the best compromise between decorrelation and perceptible signal distortion.

- Forward path delay: in hearing aid AFC applications, inserting a processing delay of  $d_1$  samples in the electro-acoustic forward path has been proposed to decorrelate the source and loudspeaker signals [75],[88], see Fig. 2.12(b), i.e.,

$$u(t) = G(q, t) d[t - d_1, \hat{\mathbf{f}}(t - d_1)]. \quad (2.88)$$

This approach is particularly useful for source signals that have an autocorrelation function that decays rapidly, e.g., voiceless speech signals, provided that the delay value  $d_1$  is chosen accordingly.

Note that when applying decorrelation in the closed signal loop, a trade-off between bias reduction and sound quality should always be sought by properly tuning the decorrelation parameters. Usually, a perceptible signal distortion is unavoidable, either because of the decorrelating signal operation itself (when strong decorrelation is applied), or because of the bias in the acoustic feedback path estimate (when weak decorrelation is applied) [77].

Decorrelation in the adaptive filtering circuit does not require the above trade-off and generally, the stronger the decorrelation, the better will be the attained sound quality. Two such approaches have been proposed,

- Adaptive filter delay [89],[90]: Due to the time needed for the loudspeaker sound to propagate through a direct coupling to the microphone, the acoustic feedback path impulse response typically exhibits an initial delay (sometimes referred to as the “dead time”, see Fig. 2.3), the value of which is proportional to the loudspeaker-microphone distance. If this initial delay (or a lower bound for it) is known a priori and corresponds to  $d_2 T_s$  s, then the first  $d_2$  coefficients in the acoustic feedback path model can be forced to zero, i.e.,

$$\hat{F}(q, t) = \hat{f}^{(d_2)}(t) q^{-d_2} + \hat{f}^{(d_2+1)}(t) q^{-(d_2+1)} + \dots + \hat{f}^{(n_{\hat{F}})} q^{-n_{\hat{F}}} \quad (2.89)$$

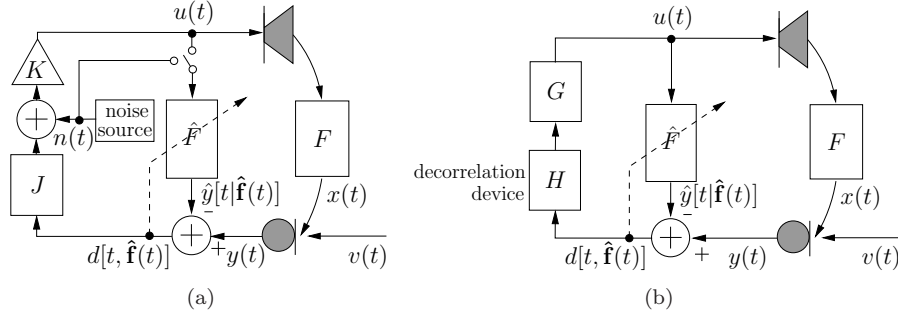


Figure 2.12: AFC with decorrelation in the closed signal loop. (a) Decorrelation by noise injection: the adaptive filter input signal can be either the loudspeaker signal  $u(t)$  or the noise signal  $n(t)$ . (b) Decorrelation in the electro-acoustic forward path: the decorrelation device corresponds to an LPTV filter  $H(q, t)$ , a nonlinear mapping  $H(\cdot, t)$ , or a processing delay  $q^{-d_1}$ .

As a consequence, the first  $d_2$  rows in the expression (2.62) for the LS estimate of the acoustic feedback path impulse response need not be considered, and likewise for the bias vector in (2.68). If we now assume that the source and loudspeaker signal cross-correlation function is small for time lags larger than  $d_2$  samples, then the remaining bias can be considered negligible.

- Decorrelating prefilters: [77],[87],[93]: From a system identification point of view, the bias in the LS estimate of the acoustic feedback path model can be eliminated by using an appropriate noise model in the identification [131], i.e., a model of the signal that disturbs the identification, more specifically the source signal in the AFC context. If we assume a (time-varying) parametric source signal model  $H(q, t)$ ,

$$v(t) = H(q, t)e(t) \quad (2.90)$$

and that an estimate  $\hat{H}(q, t)$  of  $H(q, t)$  is available, then the unbiased identification approach consists in prefiltering the loudspeaker and microphone signals with the inverse source signal model estimate before feeding these signals to the adaptive filtering algorithm. Note that the source signal excitation signal  $e(t)$  in (2.90) is assumed to be an uncorrelated signal (i.e., white noise or a Dirac impulse). This approach is depicted in Fig. 2.13(a), where the prefiltered loudspeaker and microphone signals are calculated as

$$\tilde{y}[t, \hat{\mathbf{h}}(t)] = \hat{H}^{-1}(q, t)y(t) \quad (2.91)$$

$$\tilde{u}[t, \hat{\mathbf{h}}(t)] = \hat{H}^{-1}(q, t)u(t) \quad (2.92)$$

and  $\hat{\mathbf{h}}(t)$  contains the estimated source signal model parameters. This approach was originally developed for hearing aid AFC applications [76],[91],

[92] and later on extended to room acoustic applications [77],[87],[93].

Both approaches to decorrelation in the adaptive filtering circuit rely on additional information that is not necessarily available a priori and may moreover be time-varying, i.e., the initial delay of the acoustic feedback path and the source signal model. The problem of how to concurrently estimate the initial delay and the model coefficients of the acoustic feedback path impulse response, has not yet been treated in the literature. On the other hand, the concurrent estimation of the source signal model and the acoustic feedback path model has been studied extensively by Rombouts *et al.* [96]-[98] for speech applications and by van Waterschoot and Moonen [104],[105] for audio applications. For speech source signals, the parametric source signal model preferably consists of a cascade of two all-pole models [96],[97],

$$H(q, t) = \frac{1}{A(q, t)} \frac{1}{C(q, t)} \quad (2.93)$$

with

$$A(q, t) = 1 - \sum_{i=-1}^1 \alpha^{(i)}(t) q^{-K-(l/D)-i} \quad (2.94)$$

$$C(q, t) = 1 + \sum_{i=1}^{n_C} c^{(i)}(t) q^{-i} \quad (2.95)$$

The 3-tap fractional pitch prediction model  $1/A(q, t)$  is used to model the periodic speech components that stem from the vibration of the vocal chords. Here,  $K + l/D$  represents the fractional pitch lag, with  $K$  the integer pitch lag,  $D$  the interpolation ratio, and  $l \in \{0, \dots, D - 1\}$  the fractional phase [148]. The all-pole model  $1/C(q, t)$  represents the vocal tract response that produces the formant speech components [149]. The cascade model in (2.93)-(2.95) can also be used for monophonic audio signals, while for polyphonic audio signals a cascade of a constrained pole-zero model with an all-pole model appears to be better suited [105],[150], i.e.,

$$H(q, t) = \frac{B(q, t)}{A(q, t)} \frac{1}{C(q, t)} \quad (2.96)$$

with

$$\frac{A(q, t)}{B(q, t)} = \prod_{i=1}^{n_A/2} \frac{1 - 2\nu_i \cos \theta_i(t) q^{-1} + \nu_i^2 q^{-2}}{1 - 2\rho_i \cos \theta_i(t) q^{-1} + \rho_i^2 q^{-2}}. \quad (2.97)$$

The constrained pole-zero model  $B(q, t)/A(q, t)$  then models the tonal components in the audio signal, while the all-pole model  $1/C(q, t)$  models the “noise-like” components. The constrained pole-zero model is usually parametrized using a second-order sections structure, as shown in (2.97), where the  $\theta_i(t)$





A similar problem was previously encountered in AEC, and residual echo suppression postfilters have successfully been applied in this area [151]-[153]. These postfilters operate on the echo-compensated signal and attempt to suppress the residual echo component using a spectral subtraction approach. Several attempts have been made to apply the AEC postfiltering approach to the AFC scenario [62],[90], resulting in the AFC scheme shown in Fig. 2.13(b). We should emphasize that, again, the correlation between the loudspeaker and source signal makes the residual feedback suppression problem much harder in the AFC case as compared to the AEC case. Since the postfiltering approach is based on spectral subtraction, the postfilter is usually designed directly in the frequency domain.

Janse and Belt [62] propose the following procedure to determine the postfilter magnitude response,

$$|\tilde{H}(\omega_k, t)| = \max \left\{ \frac{|Y(\omega_k, t)| - \gamma(|\hat{Y}[\omega_k, t, \hat{\mathbf{f}}(t)]| + |\hat{R}[\omega_k, t, \hat{\mathbf{f}}(t)]|)}{|D[\omega_k, t, \hat{\mathbf{f}}(t)]|}, 0 \right\} \quad (2.99)$$

$$|H(\omega_k, t)| = \lambda |H(\omega_k, t-1)| + (1-\lambda) |\tilde{H}(\omega_k, t)| \quad (2.100)$$

where  $|Y(\omega_k, t)|$ ,  $|\hat{Y}[\omega_k, t, \hat{\mathbf{f}}(t)]|$ , and  $|D[\omega_k, t, \hat{\mathbf{f}}(t)]|$  denote the short-term DFT magnitude spectra of the microphone signal, the feedback signal estimate, and the feedback-compensated signal, respectively, which are defined similarly to (2.41). Ideally, the filter in (2.99) should behave as follows: when the source signal component dominates in the short-term magnitude spectrum of the microphone signal, the amount of spectral subtraction should be small, while if the feedback signal component dominates, the amount of subtraction should be large [62]. The subtraction factor  $\gamma$  is chosen larger than one in case the estimated maximum loop gain  $\max_{\omega} |G(\omega, t)\hat{F}(\omega, t)| \geq 1$ , while  $\gamma < 1$  if  $\max_{\omega} |G(\omega, t)\hat{F}(\omega, t)| < 1$ . The first-order low-pass filtering operation in (2.100) is performed to obtain a smoothly time-varying postfilter behavior. Unfortunately, the postfilter response in (2.99) also depends on an estimate of the short-term residual feedback signal spectrum  $|\hat{R}[\omega_k, t, \hat{\mathbf{f}}(t)]|$ , yet no details are provided in [62] on how to obtain this estimate.

An alternative postfilter design procedure for residual feedback suppression was proposed by Ortega *et al.* [90], which is based on the observation that an optimal expression for the postfilter (in the sense of forcing the closed-loop frequency response in (2.10) to be exactly equal to the electro-acoustic forward path response  $G(\omega, t)$ ) is given by

$$H(\omega, t) = \frac{1}{1 + G(\omega, t)[F(\omega, t) - \hat{F}(\omega, t)]} \quad (2.101)$$

$$= 1 - \sqrt{\frac{S_r[\omega, t, \hat{\mathbf{f}}(t)]}{S_d[\omega, t, \hat{\mathbf{f}}(t)]}} \quad (2.102)$$

where  $S_r[\omega, t, \hat{\mathbf{f}}(t)]$  and  $S_d[\omega, t, \hat{\mathbf{f}}(t)]$  denote the short-term power spectral den-

sity (PSD) of the residual feedback signal and feedback-compensated signal, respectively. Here,  $S_d[\omega, t, \hat{\mathbf{f}}(t)]$  is estimated from the feedback-compensated signal  $d[t, \hat{\mathbf{f}}(t)]$  using the periodogram followed by a Mel-scale-based frequency smoothing. Finally,  $S_r[\omega, t, \hat{\mathbf{f}}(t)]$  is estimated recursively,

$$\begin{aligned} \hat{S}_r[\omega, t, \hat{\mathbf{f}}(t-1)] &= [\delta + 2\lambda(1-\lambda)]\hat{S}_r[\omega, t-1, \hat{\mathbf{f}}(t-2)] \\ &+ (1-\delta)\lambda^2\hat{S}_d[\omega, t-1, \hat{\mathbf{f}}(t-1)] + (1-\delta)(1-\lambda)^2 \frac{S_r^2[\omega, t-1, \hat{\mathbf{f}}(t-2)]}{S_d[\omega, t-1, \hat{\mathbf{f}}(t-1)]} \end{aligned} \quad (2.103)$$

where the parameters  $\lambda$  and  $\delta$  are chosen to be around 0.3 and 0.8, respectively [90]. Consequently, the first term on the right hand side of (2.103) dominates the other terms, and hence it can be understood that the initialization of the residual feedback signal PSD estimate at  $t = 0$  has a crucial effect on the quality of the estimate in (2.103).

It should be noted that a postfilter may also be used in the AFC scheme with the aim of preventing closed-loop system instability rather than suppressing the residual feedback signal. In this case, the postfilter should behave as a bank of notch filters, operating at the critical frequencies of the closed-loop system. Schmidt *et al.* [70],[102] propose an ANF postfilter that does not directly use any information from the AFC adaptive filter, and hence does not behave differently from the ANF that operates without an AFC (see Section 2.3.2). Rombouts *et al.* [50],[99] propose a postfilter based on a two-stage NHS method, in which the NHS howling detection is replaced by a proactive detection of critical frequencies by inspecting the estimated loop gain  $|G(\omega, t)\hat{F}(\omega, t)|$  using the most recent AFC acoustic feedback path estimate  $\hat{F}(q, t)$ .

### 2.6.3 Initialization

Similarly to the NHS method, an initialization procedure that is performed during the startup of the sound reinforcement system is useful to improve the performance of the AFC method. The room acoustics information that is gathered during the initialization can be elegantly incorporated in the AFC adaptive filtering algorithm using a technique known as regularization [101],[154]. The most straightforward approach to regularization consists in calculating an off-line estimate of the acoustic feedback path impulse response, and subsequently using this estimate as the initial parameter vector  $\hat{\mathbf{f}}(0)$  in any of the adaptive algorithms discussed in Section 2.6.2. While this approach may lead to a considerable improvement of the adaptive filter's convergence speed, it is non-robust to changes in the acoustic feedback path impulse response. More particularly, the impulse response may be considerably different during initialization and during operation of the system, e.g., due to the presence of an audience on the room acoustics.

A more advanced approach to regularization consists in identifying the acoustic feedback path model in a Bayesian minimum mean square error (MMSE)

framework instead of in an LS framework [101]. The acoustic feedback path impulse response  $\mathbf{f}(t)$  is then viewed as a stochastic quantity on which some prior knowledge may be available, e.g., the mean  $E\{\mathbf{f}(t)\} = \mathbf{f}_0$  and covariance matrix  $\text{cov}\{\mathbf{f}(t)\} = \mathbf{R}_f$ . In the Bayesian MMSE framework, the optimal impulse response estimate is then given by [101]

$$\hat{\mathbf{f}}(t) = \mathbf{f}_0 + (\mathbf{U}^T \mathbf{R}_v^{-1} \mathbf{U} + \mathbf{R}_f^{-1})^{-1} \mathbf{U}^T \mathbf{R}_v^{-1} (\mathbf{y} - \mathbf{U} \mathbf{f}_0) \quad (2.104)$$

which, in contrast to the LS estimate in (2.62), depends both on the acoustic feedback path statistics through  $\mathbf{f}_0$  and  $\mathbf{R}_f$ , and on the source signal statistics through  $\mathbf{R}_v$  defined in (2.71). In the context of adaptive filtering, the mean of the acoustic feedback path impulse response is usually chosen either as  $\mathbf{f}_0 = \mathbf{0}$  or as  $\mathbf{f}_0 = \hat{\mathbf{f}}(t-1)$ , which results in two well-known types of regularization, more specifically, Tikhonov regularization (TR) and Levenberg-Marquardt regularization (LMR), respectively [101]. On the other hand, the covariance matrix  $\mathbf{R}_f$  is constructed using an initial impulse response measurement or using the available room acoustic parameters such as the reverberation time and the loudspeaker-microphone distance [101]. The resulting adaptive filtering algorithms, known as TR-RLS, LMR-RLS, LMR-APA, and LMR-NLMS, do not require significantly more computations as compared to the original RLS, APA, and NLMS algorithms, if the covariance matrix  $\mathbf{R}_f$  is constructed to be a diagonal matrix [101],[154].

#### 2.6.4 Discussion

The AFC approach is widely considered to be the most promising solution to the acoustic feedback problem. Its most attractive property lies in the fact that the effect of acoustic feedback can be completely cancelled, provided that the AFC algorithm converges to the desired solution, and hence the MSG can be increased considerably. Experiments have shown that MSG increases of 15–20 dB are practically achievable [85],[99], which is two to three times more than the MSG increases obtained with the PFC and NHS approaches (see Sections 2.4.3 and 2.5.4). As a consequence, a sound reinforcement system equipped with an AFC method can generally operate at a reasonably large gain margin and hence howling, ringing, and reverberation artifacts can be avoided, resulting in a high sound quality. We should note, however, that in terms of sound quality, the choice of the decorrelation method is of crucial importance. In particular, when applying decorrelation in the closed signal loop, signal distortion appears to be unavoidable, either because the decorrelation itself is perceptible, or because the source signal is partially cancelled when the decorrelation is insufficient [77]. From this point of view, it is highly desirable to perform the decorrelation in the adaptive filtering circuit instead of in the closed signal loop. In terms of robustness, the AFC approach has benefited much from recent improvements such as postfiltering [62],[89],[90], notch filtering [70],[99],[102], adaptation control [99], and regularization [99],[101],[154].

The main disadvantage of the AFC approach is its computational complexity, which is typically much higher than the PFC and NHS complexity. Even when the cheapest adaptive filtering algorithm is applied, i.e., the NLMS algorithm which requires  $O(n_{\hat{F}})$  multiplications per time update, the AFC complexity may still exclude a real-time implementation. The reason for this is twofold: first of all, since the acoustic feedback path is modeled by its impulse response, a very high adaptive filter order is typically required. Secondly, since a sufficiently high sampling rate should be used to obtain a good sound quality (especially for audio applications), the impulse response is densely sampled hence requiring many coefficients, and moreover, a large number of adaptive filter iterations has to be performed per second. Nevertheless, several real-time AFC implementations for single-channel systems have been reported: Goertz has tested a real-time AFC setup with a 2646-tap adaptive filter (i.e., modeling the first 60 ms of the acoustic feedback path impulse response at  $f_s = 44.1$  kHz) in a room with  $T_{60} = 1.2$  s, thereby achieving a 5 dB MSG increase [80]. Rombouts *et al.* have reported MSG increases up to 14 dB in a real-time AFC experiment with a frequency domain adaptive filter of order 2048, operating at a sampling frequency of 16 kHz in a room with  $T_{60} = 120$  ms [99].

The high complexity also puts a limit on the generalization of the AFC approach to multi-channel systems. Since no results are available on how to exploit the fact that the different acoustic feedback path impulse responses of a multi-channel system share some underlying room acoustic properties, the state of the art in multi-channel AFC consists in applying  $S \cdot L$  single-channel AFC algorithms in a system having  $S$  microphones and  $L$  loudspeakers, hence the resulting complexity also increases with a factor  $S \cdot L$ .

## 2.7 Evaluation

### 2.7.1 Evaluated Algorithms

From the above exposition, it is clear that a multitude of acoustic feedback control methods has been proposed. An experimental evaluation of all the existing methods and realizations is beyond the scope of this paper. We will however provide an evaluation of a selection of methods and realizations that we consider representative for the state of the art. The evaluation is based on computer simulations rather than real-time experiments, to make sure the simulation scenario is exactly reproducible for the different algorithms. From each of the three presented categories of feedback control methods (i.e., PFC, NHS, and AFC methods), we will select three different state-of-the-art algorithms.

As for PFC, we evaluate three of the PM techniques described in Section 2.4.1: sinusoidal PM, FS, and sinusoidal DM. The corresponding PFC algorithms are denoted as **PFC-PM**, **PFC-FS**, and **PFC-DM**, respectively. The PFC-PM

and PFC-FS algorithms are realized as shown in (2.34) and (2.36), respectively, where the discrete-time Hilbert transform  $\hat{y}(t)$  is estimated using the method proposed in [109]. For the PFC-DM algorithm, we use a Hamming-windowed and truncated linear interpolation filter as given in (2.39), with an interpolation ratio  $D = 8$  and a filter length of  $2I = 32$  taps. The PFC parameters are tuned to provide a firm trade-off between the resulting MSG and signal distortion, and also taking into account the parameter values suggested in the PFC literature. In the PFC-FS algorithm, following [2], the modulation frequency is set to  $f_m = 5$  Hz. In the PFC-PM and PFC-DM algorithms, however, a lower value should be used to avoid excessive signal distortion, hence for these two approaches we set  $f_m = 1$  Hz. In the PFC-PM algorithm, a modulation index  $\beta = 3.8$  was found to produce better results than  $\beta = 2.4$ , while the PFC-DM algorithm is implemented with a modulation depth of  $\Delta_\tau = 32$  samples and a delay offset of  $\tau_0 = \Delta_\tau + 2I = 64$  samples.

In the NHS approach, many different howling detection criteria can be designed by combining the spectral and temporal microphone signal features defined in Section 2.5.2. An elaborate evaluation of each of these features, both in terms of howling detection accuracy and NHS feedback control performance, can be found in [59]. Here, we will only consider the following three approaches. In the first algorithm (denoted as **NHS-1**), the howling detection is performed as suggested in [31],[33], using a combination of the PHPR and IPMP features defined in (2.46) and (2.51), respectively. Howling is then detected if for a certain frequency, both (2.47) is fulfilled (with  $T_{\text{PHPR}} = 30$  dB) and  $\text{IPMP} \geq 3/5$  with  $Q_M = 5$ . The second algorithm (denoted as **NHS-2**) uses the PAPR feature (2.43) for howling detection, following, e.g., [48],[49]. It was found that the PAPR threshold should preferably have a different value in speech and audio applications [59], e.g.,  $T_{\text{PAPR}}^{(\text{speech})} = 33$  dB and  $T_{\text{PAPR}}^{(\text{audio})} = 55$  dB. Choosing  $T_{\text{PAPR}}^{(\text{audio})} > T_{\text{PAPR}}^{(\text{speech})}$  is recommended since the tonal components in an audio signal are much more easily misclassified as howling components. Finally, in the third algorithm (denoted as **NHS-3**), we apply the howling detection criterion proposed in [27]-[29], which combines the PNPR and IMSD features, defined in (2.48) and (2.52), respectively. According to [27]-[29], the PNPR and IMSD features are used to calculate two secondary features, namely the ‘‘peakness’’ and ‘‘slopeness’’, which are subsequently combined into a so-called feedback existence probability (FEP) function as follows:

$$\text{FEP}(\check{\omega}_i, t) = 0.7 \cdot \text{slopeness}(\check{\omega}_i, t) + 0.3 \cdot \text{peakness}(\check{\omega}_i, t). \quad (2.105)$$

The relation between the PNPR and peakness features is given in (2.50), while the calculation of the slopeness from the IMSD is performed using a nonlinear mapping (which is not explicitly given in [27]-[29]) that is chosen to be

$$\text{slopeness}(\check{\omega}_i, t) = e^{-|\text{IMSD}(\check{\omega}_i, t)|}. \quad (2.106)$$

Again, we found that a different value of the FEP threshold should be used in speech and audio applications, e.g.,  $T_{\text{FEP}}^{(\text{speech})} = 0.7$  (as suggested in [27]-[29])

and  $T_{\text{FEP}}^{(\text{audio})} = 0.9$  (with howling being detected if  $\text{FEP}(\check{\omega}_i, t) \geq T_{\text{FEP}}$ ). Since the howling detection in the NHS-1 and NHS-3 algorithms is more advanced as compared to the NHS-2 algorithm, we can expect a larger false alarm probability when using the latter algorithm [48],[49],[59]. To compensate for this effect, the notch filters applied in the NHS-2 algorithm are given a very small bandwidth, i.e., 1/60 octave, as suggested in [48],[49]. The NHS-1 and NHS-3 algorithms work with 1/10 octave notch filters, following [31],[33],[50],[99]. Also, the maximum number of cascaded notch filters as defined in (2.54) is set to  $n_H/2 = 12$  in the NHS-1 and NHS-3 algorithms, and to  $n_H/2 = 48$  in the NHS-2 algorithm. Finally, we should mention that all three NHS algorithms under consideration apply a DFT-based frequency analysis as in (2.41), with  $M = 2048$  at  $f_s = 16$  kHz, or  $M = 4096$  at  $f_s = 44.1$  kHz, and with  $P = M/2$ , from which  $N = 3$  candidate howling components are identified by peak picking.

The AFC approach will be evaluated using three different decorrelation methods, see Section 2.6.2. We refer to [78] for an evaluation of the decorrelation methods that are not covered here. The first AFC algorithm (denoted as **AFC-NI**) includes a decorrelation by noise injection, in which a white noise signal is added to the feedback-compensated signal before amplification, as suggested in [60],[70],[79]-[81]. The loudspeaker signal is chosen as the input signal to the adaptive filter (i.e., the switch in Fig. 2.12(a) is set to its vertical position). The injected noise power is adjusted to be 10 dB below the long-term feedback-compensated signal power, which results in an adaptive filter convergence speed that is comparable to the other AFC algorithms under consideration. The second algorithm (denoted as **AFC-FS**) features a decorrelation by a time-varying processing, more specifically by frequency shifting, following [70],[83]-[85]. The FS operation is realized as in the PFC-FS algorithm, and the same modulation frequency  $f_m = 5$  Hz will be used. The third algorithm (denoted as **AFC-PF**) is based on decorrelating prefilters, as proposed in [77],[87],[93]. We will use a cascade source signal model that consists of a pitch prediction model and an all-pole model, as defined in (2.93)-(2.95), which are estimated concurrently with the acoustic feedback path model using the PEM-AFROW algorithm [96],[97],[99]. For this algorithm to be applicable for both speech and audio source signals, the search range for the pitch lag  $K$  should be chosen large enough, e.g.,  $K \in \{[f_s/1000], \dots, [f_s/100]\}$  [105]. The fractional delay in the pitch prediction model (2.94) is approximated by a linear interpolation filter similar to the interpolation filter in the PFC-DM algorithm. The all-pole model order is set to  $n_C = 20$ , and both source signal models are estimated using 50 % overlapping data windows of length  $M = 320$  at  $f_s = 16$  kHz in case of speech source signals [96],[97], and of length  $M = 2048$  at  $f_s = 44.1$  kHz for audio source signals [105]. Moreover, a processing delay of half the data window length is inserted in the electro-acoustic forward path, as suggested in [96],[97],[105]. In all three AFC algorithms, the acoustic feedback path model order is equal to the length of the feedback path impulse response,

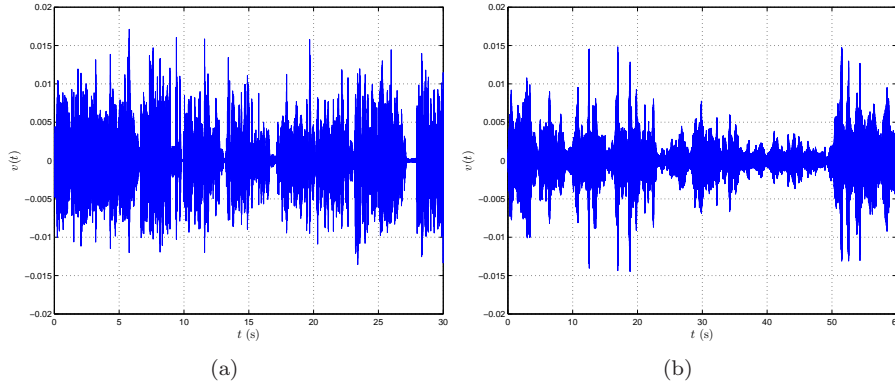


Figure 2.14: Source signals used in the evaluation of acoustic feedback control methods: (a) speech source signal ( $f_s = 16$  kHz), (b) music source signal ( $f_s = 44.1$  kHz).

i.e.,  $n_{\hat{F}} = n_F$ , and the NLMS algorithm (2.81)-(2.83) is used to update the adaptive filter coefficients. The NLMS step size is chosen to be  $\mu = 0.02$  for speech source signals and  $\mu = 0.005$  for audio source signals, while the regularization parameter is set to  $\alpha = 10^{-6}$ .

## 2.7.2 Evaluation Procedure

We will evaluate the performance of each of the nine algorithms described above in two simulation scenarios: a 30 s simulation at  $f_s = 16$  kHz with a speech source signal, and a 60 s simulation at  $f_s = 44.1$  kHz with an audio source signal. The speech signal is plotted in Fig. 2.14(a) and is taken from an interview with two male Dutch-speaking subjects that was digitally broadcasted by the Flemish Radio and Television Network (VRT), resampled to  $f_s = 16$  kHz. The audio signal is an excerpt from a CD recording of the Partita No. 2 in D minor (Allemande) for solo violin by J. S. Bach, and is shown in Fig. 2.14(b). These signals were scaled to have an RMS value of -55 dBV, which corresponds to the output voltage of a typical microphone used in sound reinforcement applications.

Each simulation consists of four equally long phases, as shown in Fig. 2.15. In the first phase, the electro-acoustic forward path broadband gain factor  $K(t)$ , defined in (2.14), is set to a value  $K_1$  that would result in a 3 dB gain margin if no acoustic feedback control were performed. In particular, this first phase should allow the AFC algorithms to partially converge before the gain is increased beyond the point of instability. In the second phase, the gain  $20 \log_{10} K(t)$  is then linearly increased up to a value  $20 \log_{10} K_2 = 20 \log_{10} K_1 + \Delta K$  beyond the point of instability (where  $\Delta K$  is defined on a dB-scale for ease



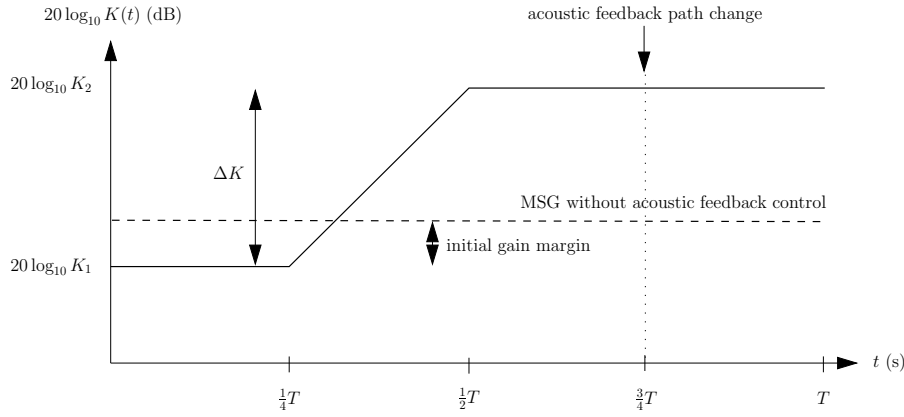


Figure 2.15: Electro-acoustic forward path gain  $20 \log_{10} K(t)$  vs. time for the acoustic feedback control simulations.

of notation). Since the different acoustic feedback control methods stabilize the closed-loop system to a different degree, the maximum gain increase  $\Delta K$  that can be allowed while maintaining a stable operation (which should not be confused with the MSG defined in (2.16)), differs depending on which method is being used. More specifically, we have found that the maximum gain increase is around  $\Delta K = 3$  dB for the PFC algorithms,  $\Delta K = 5$  dB for the NHS algorithms, and  $\Delta K = 10$  dB for the AFC algorithms. In the third and the fourth phase of the simulation, the gain factor is fixed to  $K_2$ , and at the end of the third phase, an acoustic feedback path change is simulated. The acoustic feedback path used in the first three simulation phases corresponds to the RIR shown in Fig. 2.3, while the feedback path in the fourth phase is equal to a RIR measured in the same room as the first RIR, after a 1 m displacement of the microphone.

Our goal is to evaluate the acoustic feedback control methods based on three general objectives: the achievable amplification, the sound quality, and the reliability. These objectives can be quantified by a number of performance measures, which are calculated during the third and fourth simulation phases, since these phases correspond to the preferential mode of operation for the sound reinforcement system. The *achievable amplification* is measured by the MSG and the MSG increase, which by using (2.16) are defined as

$$\text{MSG}(t) \text{ [dB]} = -20 \log_{10} \left[ \max_{\omega \in \mathcal{P}} |H(\omega, t) J(\omega, t) F(\omega, t)| \right] \quad (2.107)$$

$$\Delta \text{MSG}(t) \text{ [dB]} = -20 \log_{10} \left[ \frac{\max_{\omega \in \mathcal{P}} |H(\omega, t) J(\omega, t) F(\omega, t)|}{\max_{\omega \in \mathcal{P}} |J(\omega, t) F(\omega, t)|} \right] \quad (2.108)$$



for the PFC and NHS methods, where  $H(\omega, t)$  represents the frequency response of the PM filter or the bank of adjustable notch filters, respectively. In case of the AFC method, these measures are defined using (2.16) and (2.61) as follows,

$$\text{MSG}(t) \text{ [dB]} = -20 \log_{10} \left[ \max_{\omega \in \mathcal{P}} |J(\omega, t)[F(\omega, t) - \hat{F}(\omega, t)]| \right] \quad (2.109)$$

$$\Delta\text{MSG}(t) \text{ [dB]} = -20 \log_{10} \left[ \frac{\max_{\omega \in \mathcal{P}} |J(\omega, t)[F(\omega, t) - \hat{F}(\omega, t)]|}{\max_{\omega \in \mathcal{P}} |J(\omega, t)F(\omega, t)|} \right] \quad (2.110)$$

We will use the instantaneous value of the  $\text{MSG}(t)$ , as well as the mean and maximum value of the  $\Delta\text{MSG}(t)$ , as a performance measure in the evaluation.

An objective measure for quantifying the *sound quality* resulting from acoustic feedback control was proposed in the context of hearing aid AFC in [155]. This measure, known as the frequency-weighted log-spectral signal distortion (SD), is defined as<sup>9</sup>

$$\text{SD}(t) = \sqrt{\int_0^{f_s/2} w_{\text{ERB}}(f) \left( 10 \log_{10} \frac{S_d(f, t)}{S_v(f, t)} \right)^2 df} \quad (2.111)$$

where  $S_d(f, t)$  and  $S_v(f, t)$  denote the short-term PSD of the feedback-compensated signal and source signal, respectively, and  $w_{\text{ERB}}(f)$  is a weighting function that gives equal weight to each auditory critical band in the Nyquist interval, following Table II of the ANSI S3.5-1997 standard [156]. The short-term PSD is estimated as the squared magnitude of the short-term DFT, which is calculated using 50 % overlapping data windows of length  $M = 2048$  at  $f_s = 16$  kHz, or  $M = 4096$  at  $f_s = 44.1$  kHz. The integration in (2.111) is then approximated by a summation over the DFT frequency bins. Both the mean and maximum value of the SD measure will be used in the evaluation.

Finally, the *reliability* is quantified using two performance measures: the howling occurrence probability (HOP) and the time to recover from instability (TRI). These measures rely on an estimate of the time intervals during which howling occurs in the simulation. Howling occurrences are manually identified using the following procedure:

1. a rough estimate of the howling time intervals is obtained by listening to the feedback-compensated signal,
2. a spectrogram of the feedback-compensated signal is plotted for each of the time intervals identified in the first step, and the frequency bin(s) in which howling occurs are visually identified from the spectrogram,

---

<sup>9</sup>Note that in a real-time experiment, the source signal is not available, hence its PSD  $S_v(\omega, t)$  cannot be calculated. The SD measure can then be calculated by comparing the loudspeaker signal PSD with the PSD of a reference signal that is obtained in a secondary experiment, in the absence of acoustic feedback [155].

3. a time-varying PAPR feature is calculated for each of the time intervals identified in the first step, where the peak PSD is estimated by averaging the power in the howling frequency bins identified in the second step,
4. the time interval during which howling occurs is then defined by the time points on either side of the PAPR maximum value, at which the PAPR has decreased to a value that is 3 dB below the maximum value.

From the time points identified in the last step of the above procedure, we can estimate the time duration  $\Delta t_i$  (s) of each howling occurrence,  $i = 1, \dots, N_{\text{HO}}$ , with  $N_{\text{HO}}$  the number of howling occurrences estimated in the first step of the above procedure. The HOP and TRI measures are then defined as follows,

$$\text{HOP (\%)} = \frac{\sum_{i=1}^{N_{\text{HO}}} \Delta t_i}{T} \quad (2.112)$$

$$\text{TRI (s)} = \frac{\sum_{i=1}^{N_{\text{HO}}} \Delta t_i}{N_{\text{HO}}} \quad (2.113)$$

where  $T$  (s) denotes the length of the simulation.

### 2.7.3 Simulation Results

The instantaneous value of the  $\text{MSG}(t)$  measure versus time is displayed in Fig. 2.16 (where the left column contains the results obtained with the speech source signal, and the right column gives the results for the audio source signal). These  $\text{MSG}(t)$  curves have been smoothed with a one-pole low-pass filter to improve the clarity of the figures. The instantaneous value of the electro-acoustic forward path gain  $20 \log_{10} K(t)$  and the  $\text{MSG}$  values obtained without acoustic feedback control are also shown (where “ $\text{MSG } F_1(q)$ ” and “ $\text{MSG } F_2(q)$ ” denote the  $\text{MSG}$  before and after the acoustic feedback path change, respectively). In the PFC simulation results shown in Figs. 2.16(a)-(b), the periodic behaviour of the PM filters is clearly visible from the  $\text{MSG}$  curves. It can also be observed that these algorithms behave in a deterministic way, in the sense that their performance is independent of the instantaneous source signal and electro-acoustic forward path gain values. The PFC-DM algorithm generally performs somewhat worse compared to the other two PFC algorithms, while the PFC-PM algorithm performance can be seen to slightly improve at a higher sampling frequency. From the NHS simulation results shown in Figs. 2.16(c)-(d), the howling detection performance of the different NHS algorithms can also be judged. An instantaneous increase in the  $\text{MSG}$  curves indeed corresponds to the activation of a new notch filter (or the adjustment of an existing notch filter), while an  $\text{MSG}$  decrease occurs at the acoustic feedback path change. Ideally, no notch filters should be activated before the gain value  $20 \log_{10} K(t)$  exceeds the instantaneous  $\text{MSG}$  curves. However, this ideal behavior is exhibited only by the NHS-1 algorithm in the speech simulation. In all other cases, some notch filters are activated earlier, which indicates that some tonal source

signal components are wrongly identified as howling components. The behavior of the different NHS algorithms in terms of the  $\text{MSG}(t)$  measure is comparable for speech source signals, while the NHS-1 and NHS-2 algorithms behave quite differently from the NHS-3 algorithm in the audio simulation. We should stress that the high MSG values obtained with the NHS-1 and NHS-2 algorithms in the audio simulation are in fact caused by an excessive amount of notch filtering that is due to the poor howling detection performance and leads to a broadband attenuation of the microphone signal. Hence the resulting sound quality obtained with these methods is extremely poor for audio applications (see the discussion on the results in Table 2.2 below). Finally, the simulation results obtained with the AFC algorithms are shown in Figs. 2.16(e)-(f). In the speech simulation, the MSG performance of the AFC-NI and AFC-PF algorithms appears to be better compared to the AFC-FS algorithm. In the audio simulation, the AFC-NI algorithm initially outperforms the other algorithms, however, the AFC-PF algorithm eventually provides the highest MSG value. All three AFC algorithms appear to react in a relatively robust way to the acoustic feedback path change in the fourth phase of the simulation, except for the AFC-NI algorithm in the audio simulation.

The performance measures calculated during the third and fourth simulation phases, are shown in Tables 2.1 and 2.2 for the speech and audio simulations, respectively. Some general observations can be made concerning the performance of the different acoustic feedback control methods. The achievable amplification in terms of the MSG increase is relatively low for the PFC algorithms, and highest for the AFC algorithms, which is consistent with the MSG increase values reported in the literature. It can also be observed that, for the NHS and AFC algorithms, the MSG increase is larger when the electro-acoustic forward path gain is raised to a higher value. This effect can be explained by noting that more notch filters are activated as the gain is increased, while the AFC convergence is known to benefit from a gain increase since the power ratio of the feedback signal and source signal then also increases [99],[105]. In terms of sound quality, the SD performance measure reveals that the perceptual signal distortion is worse for the PFC algorithms and for the AFC-NI algorithm. The other AFC algorithms provide a much higher sound quality, and generally perform somewhat better than the NHS algorithms. As mentioned earlier, the NHS-1 and NHS-2 algorithms result in an extremely poor sound quality in audio applications, which is due to the poor howling detection performance. The reliability of the evaluated algorithms is seen to be slightly worse in the audio simulation as compared to the speech simulation, especially for the PFC algorithms.

Within each acoustic feedback control method, the relative performance of the different algorithms can be compared using the measures in Tables 2.1 and 2.2. Amongst the PFC algorithms, the PFC-PM algorithm should generally be preferred since it performs best in terms of nearly all performance measures.

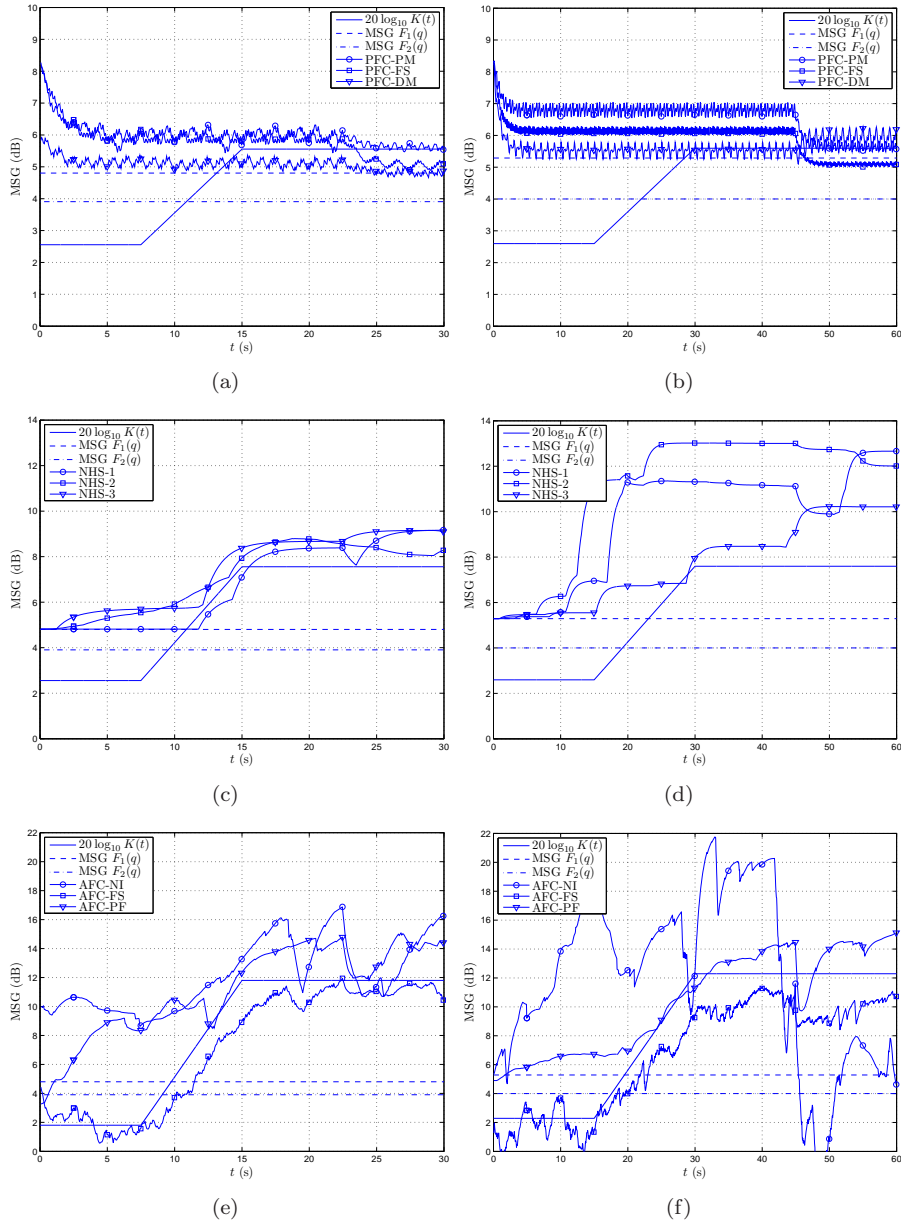


Figure 2.16: Instantaneous MSG vs. time for simulations with speech (left column) and audio (right column) source signals: (a)-(b) PFC methods ( $\Delta K = 3$  dB), (c)-(d) NHS methods ( $\Delta K = 5$  dB), (e)-(f) AFC methods ( $\Delta K = 10$  dB). Note the scale difference on the vertical axis between (a)-(b), (c)-(d), and (e)-(f).

Amongst the NHS algorithms, the NHS-3 algorithm is the only algorithm that is suited for audio applications, and moreover, in terms of achievable amplification and sound quality, this algorithm outperforms the NHS-1 and NHS-2 algorithms for speech applications also. We should note, however, that the NHS-3 howling detection method is computationally more demanding compared to the other NHS howling detection methods. Amongst the AFC algorithms, the AFC-NI algorithm yields the highest MSG increase in the speech simulation, which however comes at the cost of a poor sound quality. The AFC-PF algorithm provides the best sound quality and still allows for a relatively high MSG increase. In the audio simulation, the performance of the AFC-NI and AFC-FS algorithms is highly fluctuating, which can be observed from the discrepancy between the mean and maximum MSG values. The AFC-PF algorithm, on the other hand, produces a more steady MSG behavior in the audio simulation. The superior sound quality of the AFC-PF algorithm compared to all other evaluated algorithms results from the fact that the decorrelation is applied in the adaptive filtering circuit instead of in the closed signal loop. Note that the reliability of the AFC algorithms can be further improved by including additional features such as adaptation control, foreground/background adaptive filtering, regularization, and postfiltering, see [99] for an overview.

The feedback-compensated signals obtained in the different simulations are all available for download<sup>10</sup>, such that the sound quality can be assessed subjectively by the reader. Also, the source signals and acoustic feedback path impulse responses used in the simulations can be downloaded for benchmarking purposes.

## 2.8 Conclusion and Future Challenges

In this paper, we have attempted to provide a comprehensive overview of five decades of research in acoustic feedback control. The available literature has been reviewed following a classification of the state-of-the-art solutions into four categories: phase modulation methods, gain reduction methods, spatial filtering methods, and room modeling methods. We have also provided an in-depth treatment of three widely used acoustic feedback control methods, namely phase-modulating feedback control (PFC), notch-filter-based howling suppression (NHS), and adaptive feedback cancellation (AFC), thereby discussing conceptual as well as realization issues. Finally, several different realizations of these three methods have been evaluated and compared, in terms of their achievable amplification, sound quality, and reliability.

From the simulation results presented in this paper, we can conclude that the AFC method is superior to the PFC and NHS methods in terms of achievable

---

<sup>10</sup><ftp://ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/08-13.html>

Table 2.1: Performance measures for comparative PFC, NHS, and AFC simulations: speech source signal

		PFC			NHS			AFC			
		PFC-PM	PFC-FS	PFC-DM	NHS-1	NHS-2	NHS-3	AFC-NI	AFC-FS	AFC-PF	
achievable amplification	mean( $\Delta$ MSG)	$\Delta K = 3$	1.4	1.1	0.6	3.4	4.1	4.1	6.8	1.3	4.5
		$\Delta K = 5$				4.3	4.1	4.6	7.8	3.1	6.9
		$\Delta K = 10$							9.8	6.6	9.6
	max( $\Delta$ MSG)	$\Delta K = 3$	4.1	4.1	4.7	5.0	4.7	5.1	9.1	8.6	8.1
		$\Delta K = 5$				5.3	4.7	5.3	10.5	9.5	9.3
		$\Delta K = 10$							13.7	11.1	12.8
sound quality	mean(SD)	$\Delta K = 3$	6.2	7.1	7.9	3.3	3.6	3.1	13.8	5.6	2.4
		$\Delta K = 5$				4.3	4.5	3.8	14.0	5.6	2.6
		$\Delta K = 10$							15.1	6.0	3.9
	max(SD)	$\Delta K = 3$	10.7	11.6	16.2	7.0	7.2	5.5	30.4	8.3	6.6
		$\Delta K = 5$				20.7	8.2	6.5	30.1	8.4	6.5
		$\Delta K = 10$							31.7	10.6	16.2
reliability	HOP (%)	$\Delta K = 3$	0	0	0	3.6	0	0	0	0	0
		$\Delta K = 5$				8.5	0	1.5	0	0	0
		$\Delta K = 10$							1.3	0	2.6
	TRI (s)	$\Delta K = 3$	N/A	N/A	N/A	1.09	N/A	N/A	N/A	N/A	N/A
		$\Delta K = 5$				0.85	N/A	0.22	N/A	N/A	N/A
		$\Delta K = 10$							0.19	N/A	0.77

Table 2.2: Performance measures for comparative PFC, NHS, and AFC simulations: audio source signal

		PFC			NHS			AFC			
		PFC-PM	PFC-FS	PFC-DM	NHS-1	NHS-2	NHS-3	AFC-NI	AFC-FS	AFC-PF	
achievable amplification	mean( $\Delta$ MSG)	$\Delta K = 3$	1.6	1.0	1.1	6.6	8.1	2.5	-3.2	0.1	3.0
		$\Delta K = 5$				6.7	8.1	4.7	-2.7	1.8	4.6
		$\Delta K = 10$							6.3	5.4	9.0
	max( $\Delta$ MSG)	$\Delta K = 3$	3.9	3.9	4.6	8.7	8.7	3.9	15.0	6.1	4.5
		$\Delta K = 5$				8.8	8.7	6.2	16.0	6.8	6.5
		$\Delta K = 10$							17.2	8.6	11.3
sound quality	mean(SD)	$\Delta K = 3$	8.9	52.1	9.2	17.9	35.3	3.6	19.0	6.4	3.7
		$\Delta K = 5$				17.6	35.3	4.4	19.2	6.5	4.0
		$\Delta K = 10$							19.9	7.1	5.3
	max(SD)	$\Delta K = 3$	23.9	72.7	25.4	43.1	45.2	6.3	27.3	10.8	6.1
		$\Delta K = 5$				42.9	44.8	10.6	27.7	11.1	7.1
		$\Delta K = 10$							28.2	14.7	19.6
reliability	HOP (%)	$\Delta K = 3$	11.1	52.0	19.3	0	0	0	0	2.2	0.5
		$\Delta K = 5$				0	0	2.2	0	2.2	0.5
		$\Delta K = 10$							0.4	4.7	7.2
	TRI (s)	$\Delta K = 3$	0.23	$\infty$	0.27	N/A	N/A	N/A	N/A	0.67	0.33
		$\Delta K = 5$				N/A	N/A	0.67	N/A	0.67	0.33
		$\Delta K = 10$							0.23	0.46	1.44

amplification and sound quality, while its reliability is comparable to the reliability of the PFC and NHS methods. The AFC method should preferably be combined with a decorrelation approach that operates in the adaptive filtering circuit, e.g., using decorrelating prefilters (AFC-PF), since this approach appears to be beneficial w.r.t. the achievable amplification and sound quality. We have found the AFC-PF approach to be capable of providing an average MSG increase of approximately 9 dB, and a maximum MSG increase around 12 dB.

Looking into future research challenges in acoustic feedback control, it appears that there is little room for improvement in the PFC and NHS methods. Since these methods aim at smoothing the loop gain, a theoretical upper bound for the achievable MSG increase is given by the ratio of the peak and average magnitude response of the acoustic feedback path, which was found to be typically around 10 dB [2]. In practice, however, this upper bound is generally not achieved since the allowable values of the PFC modulation frequency and modulation depth are bounded by constraints on the signal distortion, while the number of active notch filters in the NHS method should be limited to avoid a broadband attenuation that ultimately affects sound quality. From our comparative simulation results, we may conclude that the best PFC solution consists in the use of a sinusoidal PM at low modulation frequency, while the preferable NHS solution is based on combining the howling detection method proposed by Osmanovic *et al.* [27]-[29] with a state-of-the-art biquadratic notch filter design method, e.g., the pole-zero placement technique recently proposed in [119].

On the other hand, we believe that since the AFC method appears to produce promising results, the main challenges for future research in acoustic feedback control lie in further increasing the AFC reliability and reducing its computational complexity. In terms of reliability, recent research has pointed out that so-called hybrid AFC methods, in which AFC is combined with other methods for acoustic feedback control, are far more robust compared to the traditional AFC approach. However, we believe that in the existing hybrid AFC methods, the cooperation between the different methods is still suboptimal. For example, in the combined AFC and postfiltering methods proposed in [62],[89],[90], the postfilter design is solely based on the feedback-compensated signal spectrum, while it is known from AEC that the joint design of a cancellation filter and a postfilter generally results in a better performance [152],[153]. A related issue is the combination of AFC with a gain reduction method: in [70],[102], the AFC and ANF filters are adapted independently, while in the combined AFC and AEQ approach proposed in [62] and in the combined AFC and NHS approach proposed in [99], the AEQ/NHS design is based on the most recent AFC estimate. Similarly to the joint AFC and postfilter design, it can be expected that a joint estimation of the AFC and gain reduction filter coefficients is to be preferred over a decoupled estimation. Finally, a similar remark



can be made on the joint design of an AFC and a spatial filtering method, which would probably outperform the state-of-the-art approach of AFC combined with a fixed beamformer [70] or an adaptive beamformer steered by the feedback-compensated signal [62].

The greatest challenge in AFC, however, consists in reducing the computational complexity. Since typically an already cheap NLMS-type algorithm is used, a significant complexity reduction in the AFC adaptive filtering algorithm cannot be expected. The fundamental problem lies in the fact that in AFC, the acoustic feedback path is traditionally modeled using its impulse response, which typically has a large number of coefficients. This is especially so when a high sampling frequency is applied (e.g., in audio applications). The impulse response is then more densely sampled and in addition more adaptive filter updates have to be performed per second. However, from a stability point of view, it may suffice to only model the peaks in the acoustic feedback path magnitude response instead of the complete impulse response. This may be achieved with frequency domain adaptive filtering (FDAF). However, since the frequency domain models currently used in FDAF have a fixed and uniform frequency resolution, the required FDAF filter order should still be high to guarantee that the magnitude peaks are modeled with sufficient accuracy, see, e.g., the FDAF experimental results in [99]. Another possibility for reducing the acoustic feedback path model complexity consists in using a time domain model different from the FIR model. Since the peaks in the acoustic feedback path magnitude response can be modeled as narrowband resonances, an IIR (or pole-zero) model seems to be an appropriate choice. The use of such models in room acoustics has both been recommended [128],[157],[158] and discouraged [159],[160], however, no results on the use of IIR models in AFC are available. The appeal of using such models in room acoustic applications is related to the conjecture that the IIR model denominator coefficients can in fact be assumed time-invariant in a certain acoustic environment, regardless of the loudspeaker and microphone positions [128]. A related model, which also exploits the assumption of time-invariant room acoustic resonance frequencies, is based on the use of orthogonal basis functions such as the discrete-time Laguerre or Kautz functions, which have been evaluated in an AEC context in [161],[162].

Another great challenge in acoustic feedback control, and in AFC in particular, is to generalize the methods proposed in a single-channel context to multi-channel systems. Since the number of acoustic feedback paths in a multi-channel system equals the number of loudspeakers times the number of microphones, the AFC computational complexity can be expected to increase very quickly in a multi-channel context. Again, the use of IIR models or models based on orthogonal basis functions may bring some relief, since, following the arguments in [128],[162], these models could then share a common denominator. Another problem arising in multi-channel AFC is related to the identifiability of the acoustic feedback path models in case the loudspeaker signals are corre-

lated. A similar problem occurs in multi-channel AEC, and has received quite some attention in the literature, see, e.g., [147],[163].

## Bibliography

- [1] D. A. Bohn, “Pro audio reference,” ©Rane Corp. [Online]. Available: <http://www.rane.com/digi-dic.html>
- [2] M. R. Schroeder, “Improvement of acoustic-feedback stability by frequency shifting,” *J. Acoust. Soc. Amer.*, vol. 36, no. 9, pp. 1718–1724, Sep. 1964.
- [3] H. Nyquist, “Regeneration theory,” *Bell Syst. Tech. J.*, vol. 11, pp. 126–147, 1932.
- [4] J. C. Willems, *The analysis of feedback systems*. Cambridge, Massachusetts: MIT Press, 1971.
- [5] R. W. Guelke and A. D. Broadhurst, “Reverberation time control by direct feedback,” *Acustica*, vol. 24, pp. 33–41, 1971.
- [6] P. Mapp and C. Ellis, “Improvements in acoustic feedback margin in sound reinforcement systems,” in *Preprints AES 105th Convention*, San Francisco, CA, USA, Sep. 1998, AES Preprint 4850.
- [7] M. R. Schroeder, “Improvement of acoustic feedback stability in public address systems,” in *Proc. 3rd Int. Congr. Acoust. (ICA '59)*, Stuttgart, Germany, 1959, pp. 771–775.
- [8] —, “Stop feedback in public address systems,” *Radio Electronics*, vol. 40, Feb. 1960.
- [9] —, “Improvement of feedback stability of public address systems by frequency shifting,” *J. Audio Eng. Soc.*, vol. 10, no. 2, pp. 108–109, Apr. 1962.
- [10] A. J. Prestigiaco and D. J. MacLean, “A frequency shifter for improving acoustic feedback stability,” *J. Audio Eng. Soc.*, vol. 10, no. 2, pp. 110–113, Apr. 1962.
- [11] M. D. Burkhard, “A simplified frequency shifter for improving acoustic feedback stability,” *J. Audio Eng. Soc.*, vol. 11, no. 3, pp. 234–237, Jul. 1963.
- [12] C. Vila Deutschbein, “Digital frequency shifting for electroacoustic feedback suppression,” in *Preprints AES 118th Convention*, Barcelona, Spain, May 2005, AES Preprint 6505.

- [13] J. Alisobhani and S. G. Knorr, "Improvement of acoustic-feedback stability by bandwidth compression," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-28, no. 6, pp. 636–644, Dec. 1980.
- [14] L. N. Mishin, "A method for increasing the stability of sound amplification systems," *Akust. Z.*, vol. 4, pp. 64–72, Jan./Mar. 1958, [Sov. Phys.-Acoust., vol. 4, pp. 64–71, 1958].
- [15] G. Nishinomiya, "Improvement of acoustic feedback stability of public address system by warbling," in *Proc. 6th Int. Congr. Acoust. (ICA '68)*, vol. E, Tokyo, Japan, 1968, pp. 93–96.
- [16] P. U. Svensson, "Computer simulations of periodically time-varying filters for acoustic feedback control," *J. Audio Eng. Soc.*, vol. 43, no. 9, pp. 667–677, Sep. 1995.
- [17] J. L. Nielsen and U. P. Svensson, "Performance of some linear time-varying systems in control of acoustic feedback," *J. Acoust. Soc. Amer.*, vol. 106, no. 1, pp. 240–254, Jul. 1999.
- [18] M. A. Poletti, "The stability of multichannel sound systems with frequency shifting," *J. Acoust. Soc. Amer.*, vol. 116, no. 2, pp. 853–871, Aug. 2004.
- [19] E. T. Patronis, Jr., "Acoustic feedback detector and automatic gain control," U.S. Patent 4,079,199, Mar., 1978.
- [20] —, "Electronic detection of acoustic feedback and automatic sound system gain control," *J. Audio Eng. Soc.*, vol. 26, no. 5, pp. 323–326, May 1978.
- [21] S. Ando, "Howling detection and prevention circuit and a loudspeaker system employing the same," U.S. Patent 6,252,969, Jun., 2001.
- [22] Y. Nagata, S. Suzuki, M. Yamada, M. Yoshida, M. Kitano, K. Kuroiwa, and S. Kimura, "Howling remover having cascade connected equalizers suppressing multiple noise peaks," U.S. Patent 5,710,823, Jan., 1998.
- [23] —, "Howling remover composed of adjustable equalizers for attenuating complicated noise peaks," U.S. Patent 5,729,614, Mar., 1998.
- [24] M. Hanajima, M. Yoneda, and T. Okuma, "Howling eliminator," WIPO Patent Application WO/1999/021396, Apr., 1999.
- [25] —, "Howling eliminating apparatus," U.S. Patent 6,125,187, Sep., 2000.
- [26] Y. Terada and A. Murase, "Howling control device and howling control method," U.S. Patent 7,190,800, Mar., 2007.

- [27] N. Osmanovic and V. Clarke, “Acoustic feedback cancellation system,” U.S. Patent Application 2007/0019824 A1, Jan., 2007.
- [28] —, “Acoustic feedback cancellation system,” WIPO Patent Application WO/2007/013981, Feb., 2007.
- [29] N. Osmanovic, V. E. Clarke, and E. Velandia, “An in-flight low latency acoustic feedback cancellation algorithm,” in *Preprints AES 123rd Convention*, New York, NY, USA, Oct. 2007, AES Preprint 7266.
- [30] J. B. Foley, “Adaptive periodic noise cancellation for the control of acoustic howling,” in *Proc. IEE Colloq. Adaptive Filters*, London, UK, Mar. 1989, pp. 7/1–7/4.
- [31] D. M. Oster, M. P. Lewis, and T. J. Tucker, “Method and apparatus for adaptive audio resonant frequency filtering,” WIPO Patent Application WO/1991/020134, Dec., 1991.
- [32] S. M. Kuo and J. Chen, “New adaptive IIR notch filter and its application to howling control in speakerphone system,” *IEE Electronics Lett.*, vol. 28, no. 8, pp. 764–766, Apr. 1992.
- [33] M. P. Lewis, T. J. Tucker, and D. M. Oster, “Method and apparatus for adaptive audio resonant frequency filtering,” U.S. Patent 5,245,665, Sep., 1993.
- [34] M. H. Er, T. H. Ooi, L. S. Li, and C. J. Liew, “A DSP-based acoustic feedback canceller for public address systems,” in *Proc. Int. Conf. Signal Process. (ICSP '93)*, Beijing, China, Oct. 1993, pp. 1251–1254.
- [35] —, “A DSP-based acoustic feedback canceller for public address systems,” *Microprocessors and Microsystems*, vol. 18, no. 1, pp. 39–47, Jan./Feb. 1994.
- [36] A. Kawamura, M. Matsumoto, M. Serikawa, and H. Numazu, “Sound amplifying apparatus with automatic howl-suppressing function,” U.S. Patent 5,442,712, Aug., 1995.
- [37] M. Tahernezehadi and L. Liu, “An adaptive notch filter for howling cancellation,” *Acoust. Lett.*, vol. 18, no. 8, pp. 142–145, 1995.
- [38] W. Staudacher, “Acoustic feedback cancellation for equalized amplifying systems,” U.S. Patent 5,533,120, Jul., 1996.
- [39] J. Timoney and F. B. Foley, “Robust performance of the adaptive periodic noise canceller in a closed-loop system,” in *Proc. 9th European Signal Process. Conf. (EUSIPCO '98)*, Rhodes, Greece, Sep. 1998, pp. 1177–1180.

- [40] J. E. Lane, D. Hoory, and J. Choe, “Method and apparatus for suppressing acoustic feedback in an audio system,” U.S. Patent 5,717,772, Feb., 1998.
- [41] R. Porayath and D. J. Mapes-Riordan, “Acoustic feedback elimination using adaptive notch filter algorithm,” U.S. Patent 5,999,631, Dec., 1999.
- [42] A. Kawamura, M. Matsumoto, M. Serikawa, and H. Numazu, “Sound amplifying apparatus with automatic howl-suppressing function,” European Patent EP0 599 450, Nov., 2001.
- [43] P. R. Williams, “Method and system for elimination of acoustic feedback,” WIPO Patent Application WO/2002/021 817, Mar., 2002.
- [44] W. Loetwassana, R. Punalard, and W. Silaphan, “Adaptive howling canceller using adaptive IIR notch filter: simulation and implementation,” in *Proc. 2003 IEEE Int. Conf. Neural Networks and Signal Process. (ICNNSP '03)*, Nanjing, China, Dec. 2003, pp. 848–851.
- [45] J. Timoney, F. B. Foley, and A. T. Schwarzbacher, “An explicit criterion for adaptive periodic noise canceller robustness applied to feedback cancellation,” in *4th Electron. Circuits Syst. Conf. (ECS '03)*, Bratislava, Slovakia, Sep. 2003, pp. 23–26.
- [46] J. Wei, L. Du, Z. Chen, and F. Yin, “A new algorithm for howling detection,” in *Proc. 2003 IEEE Int. Symp. Circuits Syst. (ISCAS '03)*, vol. 4, Bangkok, Thailand, May 2003, pp. 409–411.
- [47] A. F. Rocha and A. J. S. Ferreira, “An accurate method of detection and cancellation of multiple acoustic feedbacks,” in *Preprints AES 118th Convention*, Barcelona, Spain, May 2005, AES Preprint 6335.
- [48] M. Börsch, “Method for constraining electroacoustic feedback,” European Patent Application EP1 684 543 A1, Jul., 2006.
- [49] —, “Method for suppressing electro-acoustic feedback,” U.S. Patent Application 2006/0 159 282 A1, Jul., 2006.
- [50] G. Rombouts, T. van Waterschoot, and M. Moonen, “Proactive notch filtering for acoustic feedback cancellation,” in *Proc. 2nd Annual IEEE Benelux/DSP Valley Signal Process. Symp. (SPS-DARTS '06)*, Antwerp, Belgium, Mar. 2006, pp. 169–172. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/06-81.html>
- [51] R. Abe, “Howling suppression device and howling suppression method,” U.S. Patent 7,295,670, Nov., 2007.

- [52] W. Loetwassana, R. Punalard, A. Lorsawatsiri, J. Koseeyaporn, and P. Wardkein, "Adaptive howling suppressor in an audio amplifier system," in *Proc. 2007 Asia-Pacific Conf. Commun. (APCC '07)*, Bangkok, Thailand, Oct. 2007, pp. 445–448.
- [53] D. Somasundaram, "Feedback cancellation in a sound system," European Patent Application EP1 903 833 A1, Mar., 2008.
- [54] T. Kawamura and T. Kanamori, "Howling detection device and method," U.S. Patent Application 2008/0021 703 A1, Jan., 2008.
- [55] P. Gil-Cacho, T. van Waterschoot, M. Moonen, and S. H. Jensen, "Regularized adaptive notch filters for acoustic howling suppression," submitted for publication in *Proc. 17th European Signal Process. Conf. (EUSIPCO '09)*, ESAT-SISTA Technical Report TR 09-43, Katholieke Universiteit Leuven, Belgium, Feb. 2009. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/09-43.html>
- [56] T. Ito, "Apparatus detecting howling by decay profile of impulse response in sound system," U.S. Patent 6,442,280, Aug., 2002.
- [57] S. Ibaraki, H. Furukawa, and H. Naono, "Pre-howling howlback detection method," in *Proc. 1986 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '86)*, Tokyo, Japan, Apr. 1986, pp. 941–944.
- [58] Y. Takahashi, M. Tohyama, and Y. Yamasaki, "Cumulative spectral analysis for transient decaying signals in a transmission system including a feedback loop," *J. Audio Eng. Soc.*, vol. 54, no. 7/8, pp. 620–629, July/Aug. 2006.
- [59] T. van Waterschoot and M. Moonen, "Comparative evaluation of howling detection criteria in notch-filter-based howling suppression," in *Preprints AES 126th Convention*, Munich, Germany, May 2009, to appear. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/08-216.html>
- [60] J. H. Stott and N. D. Wells, "Method and apparatus for reduction of unwanted feedback," U.S. Patent 6,269,165, Jul., 2001.
- [61] T. K. Duong, E. Lefort, and M. G. Bellanger, "Acoustic feedback cancelling electro-acoustic transducer network," U.S. Patent 4,485,272, Nov., 1984.
- [62] C. P. Janse and H. J. W. Belt, "Sound reinforcement system having an echo suppressor and loudspeaker beamformer," WIPO Patent Application WO/2003/010996, Feb., 2003.
- [63] K. Kobayashi, K. Furuya, and A. Kataoka, "An adaptive microphone array for howling cancellation," *Acoust. Sci. & Tech.*, vol. 24, no. 1, pp. 45–47, Jan. 2003.

- [64] —, “A microphone array for howling cancellation,” *J. Acoust. Soc. Jpn.*, vol. 60, no. 3, pp. 115–125, Mar. 2004, (in Japanese).
- [65] G. Rombouts, A. Spriet, and M. Moonen, “Generalized sidelobe canceller based acoustic feedback cancellation,” in *Proc. 14th European Signal Process. Conf. (EUSIPCO '06)*, Firenze, Italy, Sep. 2006.
- [66] —, “Generalized sidelobe canceller based combined acoustic feedback- and noise cancellation,” *Signal Processing*, vol. 88, no. 3, pp. 571–581, Mar. 2008.
- [67] M. Goodwin and G. Elko, “Beam dithering: Acoustic feedback control using a modulated-directivity loudspeaker array,” in *Preprints AES 93rd Convention*, San Francisco, CA, USA, Oct. 1992, AES Preprint 3384.
- [68] G. W. Elko and M. M. Goodwin, “Beam dithering: Acoustic feedback control using a modulated-directivity loudspeaker array,” in *Proc. 1993 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '93)*, vol. 1, Minneapolis, MN, USA, Apr. 1993, pp. 173–176.
- [69] S. Ushiyama, T. Hirai, M. Tohyama, and Y. Shimizu, “Howling suppression by smoothing the open-loop transfer function,” *IEICE Tech. Rep.*, vol. 94, no. 20, pp. 23–28, Apr. 1994, (in Japanese).
- [70] G. Schmidt and T. Haulick, “Signal processing for in-car communication systems,” *Signal Processing*, vol. 86, no. 6, pp. 1307–1326, Jun. 2006, special Issue on Applied Speech and Audio Processing.
- [71] M. Miyoshi and Y. Kaneda, “Inverse filtering of room acoustics,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-36, no. 2, pp. 145–152, Feb. 1988.
- [72] S. J. Elliot and P. A. Nelson, “Multiple-point equalization in a room using adaptive filters,” *J. Audio Eng. Soc.*, vol. 37, no. 11, pp. 899–907, Nov. 1989.
- [73] P. A. Nelson, H. Hamed, and S. J. Elliot, “Adaptive inverse filters for stereophonic sound reproduction,” *IEEE Trans. Signal Process.*, vol. 40, no. 7, pp. 1621–1632, Jul. 1992.
- [74] J. C. Sarris, F. Jacobsen, and G. E. Cambourakis, “Sound equalization in a large region of a rectangular enclosure,” *J. Acoust. Soc. Amer.*, vol. 116, no. 6, pp. 3271–3274, Dec. 2004.
- [75] M. G. Siqueira and A. Alwan, “Steady-state analysis of continuous adaptation in acoustic feedback reduction systems for hearing-aids,” *IEEE Trans. Speech Audio Process.*, vol. 8, no. 4, pp. 443–453, Jul. 2000.

- [76] J. Hellgren and U. Forssell, "Bias of feedback cancellation algorithms in hearing aids based on direct closed loop identification," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 7, pp. 906–913, Nov. 2001.
- [77] T. van Waterschoot, G. Rombouts, and M. Moonen, "On the performance of decorrelation by prefiltering for adaptive feedback cancellation in public address systems," in *Proc. 4th IEEE Benelux Signal Process. Symp. (SPS '04)*, Hilvarenbeek, The Netherlands, Apr. 2004, pp. 167–170. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/04-24.html>
- [78] T. van Waterschoot and M. Moonen, "Assessing the acoustic feedback control performance of adaptive feedback cancellation in sound reinforcement systems," submitted for publication in *Proc. 17th European Signal Process. Conf. (EUSIPCO '09)*, ESAT-SISTA Technical Report TR 09-01, Katholieke Universiteit Leuven, Belgium, Feb. 2009. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/09-01.html>
- [79] S. Ibaraki, H. Furukawa, and H. Naono, "Howling canceller," U.S. Patent 4,747,132, May, 1988.
- [80] A. Goertz, "An adaptive subtraction filter for feedback cancellation in public address sound systems," in *Proc. 15th Int. Congr. Acoust. (ICA '95)*, Trondheim, Norway, Jun. 1995, pp. 69–72.
- [81] T. van Waterschoot, "Akoestische feedbackonderdrukker," Master's thesis, K.U.Leuven, Departement Elektrotechniek (ESAT), Leuven, Belgium, Jun. 2001, (in Dutch).
- [82] C. P. Janse and C. C. Tchang, "Acoustic feedback suppression," WIPO Patent Application WO/2005/079109, Aug., 2005.
- [83] C. P. Janse and P. A. A. Timmermans, "Signal amplifier system with improved echo cancellation," WIPO Patent Application WO/1995/28034, Oct., 1995.
- [84] —, "Signal amplifier system with improved echo cancellation," U.S. Patent 5,748,751, May, 1998.
- [85] S. Kamerling, K. Janse, and F. van der Meulen, "A new way of acoustic feedback suppression," in *Preprints AES 104th Convention*, Amsterdam, The Netherlands, May 1998, AES Preprint 4735.
- [86] D. R. Morgan, J. L. Hall, and J. Benesty, "Investigation of several types of nonlinearities for use in stereo acoustic echo cancellation," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 6, pp. 686–696, Sep. 2001.



- [87] T. van Waterschoot, K. Eneman, and M. Moonen, “Instrumental variable methods for acoustic feedback cancellation,” Katholieke Universiteit Leuven, Belgium, Tech. Rep. ESAT-SISTA TR 05-14, Oct. 2004. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/05-14.html>
- [88] P. Estermann and A. Kaelin, “Feedback cancellation in hearing aids: results from using frequency-domain adaptive filters,” in *Proc. 1994 IEEE Int. Symp. Circuits Syst. (ISCAS '94)*, vol. 2, London, UK, May/June 1994, pp. 257–260.
- [89] F. Gallego, E. Lleida, E. Masgrau, and A. Ortega, “Method and system for suppressing echoes and noises in environments under variable acoustic and highly feedback conditions,” WIPO Patent Application WO/2002/101728, Dec., 2002.
- [90] A. Ortega, E. Lleida, and E. Masgrau, “Speech reinforcement system for car cabin communications,” *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 917–929, Sep. 2005.
- [91] J. Hellgren, “Analysis of feedback cancellation in hearing aids with Filtered-X LMS and the direct method of closed-loop identification,” *IEEE Trans. Speech Audio Process.*, vol. 10, no. 2, pp. 119–131, Feb. 2002.
- [92] A. Spriet, M. Moonen, and I. Proudler, “Feedback cancellation in hearing aids: an unbiased modelling approach,” in *Proc. 11th European Signal Process. Conf. (EUSIPCO '02)*, Toulouse, France, Sep. 2002, pp. 531–534.
- [93] A. Ortega, E. Lleida, E. Masgrau, L. Buera, and A. Miguel, “Acoustic feedback cancellation in speech reinforcement systems for vehicles,” in *Proc. Interspeech 2005*, Lisbon, Portugal, Sep. 2005, pp. 2061–2064.
- [94] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. Cambridge, Massachusetts: MIT Press, 1986.
- [95] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
- [96] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, “Acoustic feedback suppression for long acoustic paths using a nonstationary source model,” in *Proc. 13th European Signal Process. Conf. (EUSIPCO '05)*, Antalya, Turkey, Sep. 2005.
- [97] —, “Acoustic feedback suppression for long acoustic paths using a nonstationary source model,” *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.

- [98] G. Rombouts, P. Verhoeve, K. Struyve, T. van Waterschoot, and M. Moonen, “Circuit and method for estimating a room impulse response,” European Patent Application EP1 675 374 A1, Jun., 2006.
- [99] G. Rombouts, T. van Waterschoot, and M. Moonen, “Robust and efficient implementation of the PEM-AFROW algorithm for acoustic feedback cancellation,” *J. Audio Eng. Soc.*, vol. 55, no. 11, pp. 955–966, Nov. 2007.
- [100] T. van Waterschoot, G. Rombouts, and M. Moonen, “Dually regularized recursive prediction error identification for acoustic feedback and echo cancellation,” in *Proc. 15th European Signal Process. Conf. (EUSIPCO '07)*, Poznań, Poland, Sep. 2007, pp. 1610–1614.
- [101] —, “Optimally regularized adaptive filtering algorithms for room acoustic signal enhancement,” *Signal Processing*, vol. 88, no. 3, pp. 594–611, Mar. 2008.
- [102] T. Haulick, G. U. Schmidt, and H. Lenhardt, “Feedback reduction in communication systems,” European Patent EP1 679 874, May, 2008.
- [103] H. Okumura and H. Fujita, “Adaptive howling canceller,” European Patent Application EP1 615 463 A2, Jan., 2006.
- [104] T. van Waterschoot and M. Moonen, “Adaptive feedback cancellation for audio signals using a warped all-pole near-end signal model,” in *Proc. 2008 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '08)*, Las Vegas, NV, USA, Apr. 2008, pp. 269–272.
- [105] —, “Adaptive feedback cancellation for audio applications,” *Signal Processing*, conditionally accepted for publication, Dec. 2008, ESAT-SISTA Technical Report TR 07-30, Katholieke Universiteit Leuven, Belgium. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/07-30.html>
- [106] T. A. C. M. Claasen and W. F. G. Mecklenbräuker, “On stationary linear time-varying systems,” *IEEE Trans. Circuits Syst.*, vol. CAS-29, no. 3, pp. 169–184, Mar. 1982.
- [107] A. Papoulis, *Signal Analysis*. New York: McGraw-Hill, 1977.
- [108] B. Boashash and A. P. Reilly, “Algorithms for time-frequency signal analysis,” in *Methods and Applications of Time-Frequency Signal Analysis*, B. Boashash, Ed. Melbourne: Longman Cheshire, 1992, ch. 7.
- [109] S. L. Marple, Jr., “Computing the discrete-time “analytic” signal via FFT,” *IEEE Trans. Signal Process.*, vol. 47, no. 9, pp. 2600–2603, Sep. 1999.

- [110] A. Reilly, G. Frazer, and B. Boashash, “Analytic signal generation – tips and traps,” *IEEE Trans. Signal Process.*, vol. 42, no. 11, pp. 3241–3245, Nov. 1994.
- [111] J. Dattorro, “Effect design – part 2: Delay-line modulation and chorus,” *J. Audio Eng. Soc.*, vol. 45, no. 10, pp. 764–788, Oct. 1997.
- [112] S. Disch and U. Zölzer, “Modulation and delay line based digital audio effects,” in *Proc. 2nd COST G-6 Workshop Digital Audio Effects (DAFX '99)*, Trondheim, Norway, Dec. 1999, pp. 5–8.
- [113] P. Dutilleux and U. Zölzer, “Delays,” in *DAFX: Digital Audio Effects*, U. Zölzer, Ed. New York: Wiley, 2002, ch. 3.
- [114] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine, “Splitting the unit delay: Tools for fractional delay filter design,” *IEEE Signal Process. Mag.*, vol. 13, no. 1, pp. 30–60, Jan. 1996.
- [115] D. Griesinger, “Improving room acoustics through time-variant synthetic reverberation,” in *Preprints AES 90th Convention*, Paris, France, Feb. 1991, AES Preprint 3014.
- [116] F. J. Harris, “On the use of windows for harmonic analysis with the discrete Fourier transform,” *Proc. IEEE*, vol. 66, no. 1, pp. 51–83, Jan. 1978.
- [117] J. O. Smith, *Mathematics of the Discrete Fourier Transform (DFT)*. <http://ccrma.stanford.edu/~jos/mdft/>, accessed Nov. 2008, online book.
- [118] A. V. Oppenheim, D. H. Johnson, and K. Steiglitz, “Computation of spectra with unequal resolution using the fast Fourier transform,” *Proc. IEEE*, vol. 59, no. 2, pp. 299–301, Feb. 1971.
- [119] T. van Waterschoot and M. Moonen, “A pole-zero placement technique for designing second-order IIR parametric equalizer filters,” *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 8, pp. 2561–2565, Nov. 2007.
- [120] J. A. Moorer, “The manifold joys of conformal mapping: Applications of digital filtering in the studio,” *J. Audio Eng. Soc.*, vol. 31, no. 11, pp. 826–841, Nov. 1983.
- [121] K. Hirano, S. Nishimura, and S. K. Mitra, “Design of digital notch filters,” *IEEE Trans. Commun.*, vol. COM-22, no. 7, pp. 964–970, Jul. 1974.
- [122] S. A. White, “Design of a digital biquadratic peaking or notch filter for digital audio equalization,” *J. Audio Eng. Soc.*, vol. 34, no. 6, pp. 479–483, Jun. 1986.

- [123] P. A. Regalia and S. K. Mitra, "Tunable digital frequency response equalization filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 1, pp. 118–120, Jan. 1987.
- [124] D. J. Shpak, "Analytical design of biquadratic filter sections for parametric filters," *J. Audio Eng. Soc.*, vol. 40, no. 11, pp. 876–885, Nov. 1992.
- [125] D. C. Massie, "An engineering study of the four-multiply normalized ladder filter," *J. Audio Eng. Soc.*, vol. 41, no. 7/8, pp. 564–582, July/August 1993.
- [126] R. Bristow-Johnson, "The equivalence of various methods of computing biquad coefficients for audio parametric equalizers," in *Preprints AES 97th Convention*, San Francisco, CA, USA, Nov. 1994, AES Preprint 3906.
- [127] S. J. Orfanidis, "Digital parametric equalizer design with prescribed Nyquist-frequency gain," *J. Audio Eng. Soc.*, vol. 45, no. 6, pp. 444–455, Jun. 1997.
- [128] Y. Haneda, S. Makino, and Y. Kaneda, "Common acoustical pole and zero modeling of room transfer functions," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 2, pp. 320–328, 1994.
- [129] C. P. Boner, "A procedure for controlling room-ring modes and feedback modes in sound systems with narrow-band filters," *J. Audio Eng. Soc.*, vol. 13, no. 4, pp. 297–299, 1965.
- [130] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Upper Saddle River, New Jersey: Prentice-Hall, 1993.
- [131] U. Forssell and L. Ljung, "Closed-loop identification revisited," *Automatica*, vol. 35, no. 7, pp. 1215–1241, Jul. 1999.
- [132] T. van Waterschoot, G. Rombouts, P. Verhoeve, and M. Moonen, "Double-talk-robust prediction error identification algorithms for acoustic echo cancellation," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 846–858, Mar. 2007.
- [133] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1996.
- [134] J. M. Cioffi and T. Kailath, "Fast recursive least squares transversal filters for adaptive processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, no. 2, pp. 304–337, Apr. 1984.
- [135] D. T. M. Slock and T. Kailath, "Numerically stable fast transversal filters for recursive least squares adaptive filtering," *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 92–114, Jan. 1991.

- [136] S. H. Jensen, “Acoustic echo canceller for hands-free mobile radiotelephony,” in *Proc. 6th European Signal Process. Conf. (EUSIPCO '92)*, Brussels, Belgium, Aug. 1992, pp. 1629–1632.
- [137] S. L. Gay, “Dynamically regularized fast RLS with application to echo cancellation,” in *Proc. 1996 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '96)*, vol. 2, Atlanta, Georgia, USA, May 1996, pp. 957–960.
- [138] J. Benesty, T. Gänslér, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*. Berlin: Springer-Verlag, 2001.
- [139] B. Baykal and A. G. Constantinides, “Underdetermined-order recursive least-squares adaptive filtering: The concept and algorithms,” *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 346–362, Feb. 1997.
- [140] M. Rupp, “A family of adaptive filter algorithms with decorrelating properties,” *IEEE Trans. Signal Process.*, vol. 46, no. 3, pp. 771–775, Mar. 1998.
- [141] S. L. Gay and S. Tavathia, “The fast affine projection algorithm,” in *Proc. 1995 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '95)*, vol. 5, Detroit, Michigan, USA, May 1995, pp. 3023–3026.
- [142] M. Tanaka, S. Makino, and J. Kojima, “A block exact fast affine projection algorithm,” *IEEE Trans. Speech Audio Process.*, vol. 7, no. 1, pp. 79–86, Jan. 1999.
- [143] G. Rombouts and M. Moonen, “A sparse block exact affine projection algorithm,” *IEEE Trans. Speech Audio Process.*, vol. 10, no. 2, pp. 100–108, Feb. 2002.
- [144] G. Rombouts, T. van Waterschoot, K. Struyve, P. Verhoeve, and M. Moonen, “Identification of undermodelled room impulse responses,” in *Proc. 2005 Int. Workshop Acoustic Echo Noise Control (IWAENC '05)*, Eindhoven, The Netherlands, Sep. 2005, pp. 153–156.
- [145] R. D. Poltmann, “Stochastic gradient algorithm for system identification using adaptive FIR-filters with too low number of coefficients,” *IEEE Trans. Circuits Syst.*, vol. 35, no. 2, pp. 247–250, Feb. 1988.
- [146] C. Paleologu, S. Ciochina, and J. Benesty, “Variable step-size NLMS algorithm for under-modeling acoustic echo cancellation,” *IEEE Signal Process. Lett.*, vol. 15, pp. 5–8, 2008.
- [147] M. M. Sondhi, D. R. Morgan, and J. L. Hall, “Stereophonic acoustic echo cancellation – an overview of the fundamental problem,” *IEEE Signal Process. Lett.*, vol. 2, no. 8, pp. 148–151, Aug. 1995.

- [148] Y. Qian, G. Chahine, and P. Kabal, "Pseudo-multi-tap pitch filters in a low bit-rate CELP speech coder," *Speech Commun.*, vol. 14, no. 4, pp. 339–358, Sep. 1994.
- [149] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, no. 4, pp. 561–578, Apr. 1975.
- [150] T. van Waterschoot and M. Moonen, "Comparison of linear prediction models for audio signals," *EURASIP J. Audio, Speech, Music Process.*, vol. 2008, Article ID 706935, 24 pages, 2008, doi:10.1155/2008/706935.
- [151] S. Gustafsson, R. Martin, and P. Vary, "Combined acoustic echo control and noise reduction for hands-free telephony," *Signal Processing*, vol. 64, no. 1, pp. 21–32, Jan. 1998, special Issue on Acoustic Echo and Noise Control.
- [152] E. Haensler and G. U. Schmidt, "Hands-free telephones – joint control of echo cancellation and postfiltering," *Signal Processing*, vol. 80, no. 11, pp. 2295–2305, Nov. 2000.
- [153] G. Enzner and P. Vary, "Frequency-domain adaptive Kalman filter for acoustic echo control in hands-free telephones," *Signal Processing*, vol. 86, no. 6, pp. 1140–1156, Jun. 2006, special Issue on Applied Speech and Audio Processing.
- [154] T. van Waterschoot, G. Rombouts, and M. Moonen, "MSE optimal regularization of APA and NLMS algorithms in room acoustic applications," in *Proc. 2006 Int. Workshop Acoustic Echo Noise Control (IWAENC '06)*, Paris, France, Sep. 2006.
- [155] A. Spriet, K. Eneman, M. Moonen, and J. Wouters, "Objective measures for real-time evaluation of adaptive feedback cancellation algorithms in hearing aids," in *Proc. 16th European Signal Process. Conf. (EUSIPCO '08)*, Lausanne, Switzerland, Aug. 2008.
- [156] ANSI S3.5-1997, "American national standard methods for calculation of the speech intelligibility index," American National Standards Institute: New York, 1997.
- [157] J. Pongsiri, P. Amin, and C. Thompson, "Modeling the acoustic transfer function of a room," in *Proc. 12th Int. Conf. Mathematical Comput. Modelling Scientific Comput. (ICMCMSC '99)*, Chicago, Illinois, USA, Aug. 1999.
- [158] T. Gustafsson, J. Vance, H. R. Pota, B. D. Rao, and M. M. Trivedi, "Estimation of acoustical room transfer functions," in *Proc. 39th IEEE Conf. Decision Control (CDC '00)*, Sydney, Australia, Dec. 2000, pp. 5184–5189.

- [159] S. Gudvangen and S. J. Flockton, "Comparison of pole-zero and all-zero modelling of acoustic transfer functions," *IEE Electronics Lett.*, vol. 28, no. 21, pp. 1976–1978, 1992.
- [160] A. P. Liavas and P. A. Regalia, "Acoustic echo cancellation: Do IIR models offer better modeling capabilities than their FIR counterparts," *IEEE Trans. Signal Process.*, vol. 46, no. 9, pp. 2499–2504, Sep. 1998.
- [161] G. W. Davidson and D. D. Falconer, "Reduced complexity echo cancellation using orthonormal functions," *IEEE Trans. Circuits Syst.*, vol. 38, no. 1, pp. 20–28, Jan. 1991.
- [162] L. S. H. Ngia, "Recursive identification of acoustic echo systems using orthonormal basis functions," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 3, pp. 278–293, May 2003.
- [163] J. Benesty, D. R. Morgan, and M. M. Sondhi, "A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 2, pp. 156–165, Mar. 1998.





## Part II

# Acoustic Echo Cancellation



## Chapter 3

# Double-Talk-Robust AEC

Double-talk-robust prediction error identification  
algorithms for acoustic echo cancellation

Toon van Waterschoot, Geert Rombouts, Piet Verhoeve,  
and Marc Moonen

Published in *IEEE Trans. Signal Process.*, vol. 55, no. 3, Mar.  
2007, pp. 846–858.

©2007 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

### Contributions of first author

- literature study
- co-development of concept of achieving double-talk-robustness by minimum-variance estimation
- co-development of formal relation between minimum-variance estimation and recursive prediction error identification
- co-development of recursive prediction error identification algorithm
- co-development of algorithm implementation and complexity analysis
- co-design of evaluation experiments
- software implementation and computer simulations
- co-interpretation of simulation results
- co-formulation of conclusion
- text redaction and editing

## Abstract

The performance of an acoustic echo canceller may be severely degraded by the presence of a near-end signal. In such a double-talk situation, the variance of the echo path estimate typically increases, resulting in slow convergence or even divergence of the adaptive filter. This problem is usually tackled by equipping the echo canceller with a double-talk detector that freezes adaptation during near-end activity. Nevertheless there is a need for more robust adaptive algorithms since the adaptive filter's convergence may be affected considerably in the time interval needed to detect double-talk. Moreover, in some applications, near-end noise may be continuously present and then the use of a double-talk detector becomes futile. Robustness to double-talk may be established by taking into account the near-end signal characteristics, which are however unknown and time-varying. In this paper we show how concurrent estimation of the echo path and an autoregressive near-end signal model can be performed using prediction error identification techniques. We develop a general recursive prediction error identification algorithm and compare it to three existing algorithms from adaptive feedback cancellation. The potential benefit of the algorithms in a double-talk situation is illustrated by means of computer simulations. It appears that especially in the stochastic gradient case a huge improvement in convergence behaviour can be obtained.

## 3.1 Introduction

Acoustic echo cancellation (AEC) has been a popular research topic in acoustic signal processing, motivated mainly by the increasing demand for hands-free speech communication. A general AEC scenario is shown in Fig. 3.1. A speech signal  $u(t)$  from the far-end side is broadcast in an acoustic enclosure (the "room") by means of a loudspeaker. A microphone is present in the room for recording a local signal  $v(t)$  (the "near-end signal") which is to be transmitted back to the far-end side. An acoustic echo path exists between the loudspeaker and the microphone such that the recorded microphone signal  $y(t) = x(t) + v(t)$  contains an undesired echo component  $x(t)$  in addition to the near-end signal component  $v(t)$ . If the echo path transfer function is modelled as a finite impulse response (FIR) filter  $F(q, t)$ , then the echo component can be considered as a filtered version of the loudspeaker signal:  $x(t) = F(q, t)u(t)$ . Here  $q$  denotes the time shift operator, e.g.,  $q^{-k}u(t) = u(t - k)$ . The main objective in AEC is to identify the unknown room impulse response (RIR)  $F(q, t)$  and hence to subtract an estimate of the echo component from the microphone signal. In this way, an echo-compensated signal  $d(t) = y(t) - \hat{F}(q, t)u(t)$  is sent to the far-end side, with  $\hat{F}(q, t)$  an estimate of  $F(q, t)$ .

Since  $F(q, t)$  may be time-varying (e.g., due to people moving around the room),

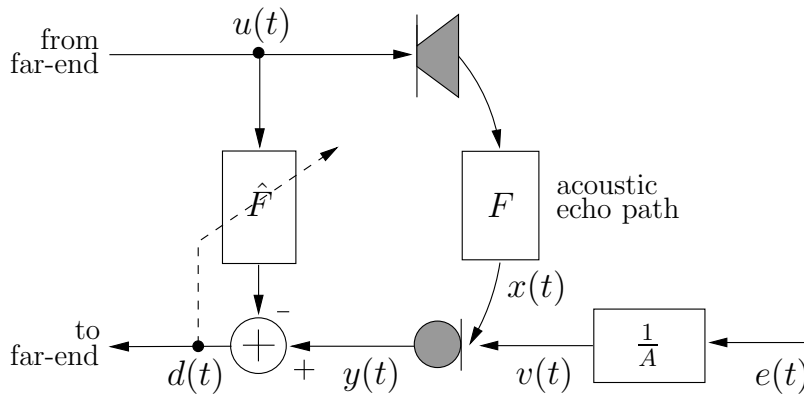


Figure 3.1: AEC scenario with autoregressive modelling of the near-end signal.

an adaptive algorithm is usually applied for the estimation of the RIR. It is well known that the convergence speed and hence the tracking capabilities of standard adaptive algorithms [like recursive least squares (RLS), normalized least mean squares (NLMS), and the affine projection algorithm (APA)] may decrease severely when a near-end signal is present (“double-talk” periods). Therefore, efficient double-talk detectors (DTDs) have been developed, which are used to slow down or switch off the adaptation during double-talk periods. Typically, in a DTD, some decision statistic is calculated using the available signals and signal estimates, and compared to a threshold for deciding whether or not a double-talk situation occurs. Since the popular Geigel DTD, which was originally developed for line echo cancellation (LEC) [1], does not always perform satisfactorily in an AEC application, several alternative DTD algorithms have been proposed [2]-[5]. Probably the most widespread DTD for AEC is based on the cross-correlation method described in [2],[3]. Recently also some DTD algorithms have been developed that are specifically suited for subband [6] and stereo AEC [7]. It has been noted however that at the onset of a double-talk period, the adaptive algorithm may diverge considerably before double-talk is detected. Hence it is desirable to improve the robustness of the adaptive algorithm to compensate for the detection lag as well as for other DTD imperfections. In [8] the robustness of several adaptive algorithms w.r.t. so-called near-end noise bursts in LEC is improved by introducing a scaled nonlinearity in the LEC cost function. This method, inspired by the theory of robust statistics [9], is parametrized by a scale factor that governs the trade-off between robustness and convergence rate. When combined with a cross-correlation-based DTD, the method shows improved performance also in AEC double-talk situations, as compared to the standard adaptive algorithms combined with the same DTD [10, Ch. 6], [11]. Recently, the robust statistics approach has also proven successful in adaptive beamforming applications [12]. However, the increase in robustness comes at the expense of a decreased

convergence rate when no double-talk is present. Moreover the quality of the scale factor estimate depends heavily on the DTD performance.

In a normal telephone conversation, a double-talk situation will occur during approximately 20% of the time [13]. However, in some AEC scenarios, near-end noise will be continuously present and then the use of a DTD becomes futile. This may be the case for example in a noisy teleconferencing application, in an automatic gain adjustment system equipped with an echo canceller, or in adaptive feedback cancellation (AFC) [14],[15]. These applications have motivated the design of robust adaptive algorithms that allow for continuous adaptation during double-talk. A first class of these double-talk-robust algorithms is based on the interplay between the echo canceller and a postfilter that is added to suppress the residual echo which remains after cancellation [16]. Such a joint echo canceller and postfilter system may be extended even further with a microphone array [17] or a blind signal separation algorithm [18]. A second approach to double-talk-robust adaptation is to switch to a different adaptation strategy during double-talk. In [19] a so-called maximum-length correlation estimate replaces the stochastic gradient RIR estimate whenever a double-talk situation occurs. In [20],[21] an adaptive cross-spectral technique is applied instead of a standard adaptive algorithm, and it is claimed to be insensitive to double-talk. Finally, a third method to obtain double-talk robustness is to remove some correlation in the microphone signal or echo-compensated signal. In [22] an attempt is made to extract the pitch of the residual echo component in the echo-compensated signal, whereas in [23] the microphone signal is processed by an adaptive prediction filter to remove the near-end signal correlation during double-talk.

In this paper we aim at developing a recursive identification algorithm that behaves in a robust way in a continuous double-talk situation (without a DTD) as well as in a bursting double-talk scenario (with, or even without a DTD). In contrast to the method described in [10, Ch. 6], we investigate how robustness may be improved by linear modifications to the standard adaptive algorithms. In our classification of continuously adapting double-talk-robust algorithms, the proposed algorithm fits in the third class described above, since it involves prediction of the near-end signal component in the microphone signal. From linear estimation theory [24], we know that the best (i.e., minimum-variance) linear unbiased estimator (BLUE) for an unknown system depends on the characteristics of the noise acting upon the system. In the AEC context, it is the near-end signal that acts as a noise signal to the RIR identification. Therefore we expect that by using knowledge of the near-end signal characteristics, the convergence properties of the RIR identification algorithm can be improved. Moreover, in contrast to many existing double-talk-robust algorithms, the increase in robustness using the proposed method does not come at the expense of slower convergence in a single-talk situation. However, the near-end signal characteristics are typically unknown and time-varying. Therefore they have

to be estimated concurrently with the unknown RIR. Inspired by prediction error identification theory [25], we propose a new recursive prediction error (RPE) identification algorithm for concurrent estimation of the RIR and of an autoregressive model of the near-end signal.

The problem of concurrently estimating the RIR and the near-end signal characteristics has also been studied in the context of adaptive feedback cancellation (AFC) [14],[15],[26]. It appears that several algorithms proposed for AFC can actually be derived starting from the proposed RPE algorithm. We will explore three algorithms developed for AFC, namely the two-channel adaptive filtering (2ch-AF) algorithm and the prediction-error-method-based adaptive filtering (PEM-AF) algorithm, both developed for hearing aid applications [14], and finally the so-called PEM-AFROW algorithm, proposed in [15] as a PEM-AF variant with only row operations on the loudspeaker data matrix. In contrast to the RPE algorithm, the first two algorithms both rely on the assumption that the near-end signal characteristics do not change significantly on a time scale comparable to the echo path length. As pointed out in [26] this assumption does not hold for room acoustic applications, where the RIR length may be several orders of magnitude larger than the period over which a typical audio signal can be considered stationary. To tackle this particular problem, the PEM-AF algorithm has been modified, resulting in the PEM-AFROW algorithm [15] for room acoustic applications. Whereas in the RPE algorithm the autoregressive near-end signal model is identified recursively, the near-end signal model identification is performed non-recursively and in a frame-based manner in the PEM-AFROW algorithm. The frame-based representation is well suited for speech signals, but less appropriate for near-end signals that do not exhibit short-term stationary behaviour such as music or babble noise.

The outline of the paper is as follows. We start by defining the data model and reviewing some results from linear estimation theory in Section 3.2, which have motivated the work described in the rest of the paper. In Section 3.3 we describe four prediction error identification algorithms for concurrent estimation of a RIR and an autoregressive near-end signal model: the new RPE algorithm and the PEM-AF, PEM-AFROW, and 2ch-AF algorithms from adaptive feedback cancellation. Section 3.4 deals with some implementation issues and we compare the computational complexity of the prediction error identification algorithms with RLS and NLMS complexity. In Section 3.5, simulation results are shown in which the convergence properties of the proposed algorithms are compared with those of the standard RLS and NLMS algorithms, as well as the double-talk-robust algorithm described in [10, Ch. 6]. Finally, Section 3.6 concludes the paper.



## 3.2 Linear Estimation Background

We first review some results from linear estimation theory [24] to indicate how the variance of a RIR estimator can be decreased. Let us assume that a data record  $\{u(k), y(k)\}_{k=1}^t$  of microphone and loudspeaker samples is available, as well as initial conditions  $\{u(k)\}_{k=1-n_F-n_A}^0$  and  $\{y(k)\}_{k=1-n_A}^0$  (with model orders  $n_F$  and  $n_A$  to be defined later on). For now we also assume that  $F(q, t)$  is time-invariant over the observed time interval so no data windowing is applied (we will consider time-varying RIR's when we derive a recursive estimation procedure in Section 3.3). The data model can then be written as

$$\mathbf{y} = \mathbf{U}\mathbf{f} + \mathbf{v}, \quad (3.1)$$

where the data matrices are defined as follows:

$$\mathbf{U} = \begin{bmatrix} u(t) & u(t-1) & \dots & u(t-n_F) \\ u(t-1) & u(t-2) & \dots & u(t-n_F-1) \\ \vdots & \vdots & \ddots & \vdots \\ u(1) & u(0) & \dots & u(1-n_F) \end{bmatrix}, \quad (3.2)$$

$$\mathbf{y} = [y(t) \quad y(t-1) \quad \dots \quad y(1)]^T, \quad (3.3)$$

$$\mathbf{v} = [v(t) \quad v(t-1) \quad \dots \quad v(1)]^T, \quad (3.4)$$

and

$$\mathbf{f} = [f_0 \quad f_1 \quad \dots \quad f_{n_F}]^T \quad (3.5)$$

is the  $(n_F+1) \times 1$  parameter vector containing the true coefficients of  $F(q, t)$  that have to be estimated. The loudspeaker signal Hankel matrix  $\mathbf{U}$  is considered to be deterministic, whereas the near-end signal vector  $\mathbf{v}$  and hence also the microphone signal vector  $\mathbf{y}$  are random vectors. Note that no additional term is added for representing possible background noise, since we assume that all near-end disturbances are represented together with the useful near-end signal in  $\mathbf{v}$ .

Typically in AEC, an estimator for the RIR  $F(q, t)$  is based on an ordinary least squares (LS) criterion

$$\min_{\hat{\mathbf{f}}} V_{\text{LS}}(t, \hat{\mathbf{f}}) = \min_{\hat{\mathbf{f}}} (\mathbf{y} - \mathbf{U}\hat{\mathbf{f}})^T (\mathbf{y} - \mathbf{U}\hat{\mathbf{f}}) \quad (3.6)$$

which leads to the well-known estimator

$$\boxed{\hat{\mathbf{f}}_{\text{LS}} = (\mathbf{U}^T \mathbf{U})^{-1} \mathbf{U}^T \mathbf{y}.} \quad (3.7)$$

The LS estimator  $\hat{\mathbf{f}}_{\text{LS}}$  is unbiased, i.e.,  $E\{\hat{\mathbf{f}}_{\text{LS}}\} = \mathbf{f}$  (with  $E$  the expectation operator), if the near-end signal  $v(t)$  is uncorrelated with the loudspeaker signal

$u(t)$ , i.e., if

$$E\{\mathbf{U}^T \mathbf{v}\} = \mathbf{0}_{(n_F+1) \times 1}. \quad (3.8)$$

In AEC, this means that the signal  $d(t)$  sent to the far-end side is not reflected back to the near-end side due to an acoustic echo path in the far-end room. In the sequel we will assume (3.8) to hold. Except when the near-end signal  $v(t)$  is a white noise signal, the least squares estimator (3.7) is suboptimal. Indeed, a linear unbiased estimator with a smaller variance can be obtained by taking into account the characteristics of the near-end signal.

Minimizing the variance  $E(\hat{\mathbf{f}} - E\{\hat{\mathbf{f}}\})(\hat{\mathbf{f}} - E\{\hat{\mathbf{f}}\})^T$  of a linear estimator  $\hat{\mathbf{f}}$  under an unbiasedness constraint yields the best linear unbiased estimator (BLUE),

$$\boxed{\hat{\mathbf{f}}_{\text{BLUE}} = (\mathbf{U}^T \mathbf{R}^{-1} \mathbf{U})^{-1} \mathbf{U}^T \mathbf{R}^{-1} \mathbf{y}}, \quad (3.9)$$

which depends on the near-end signal covariance matrix  $\mathbf{R}$ , defined as

$$\mathbf{R} \triangleq E\{(\mathbf{v} - E\{\mathbf{v}\})(\mathbf{v} - E\{\mathbf{v}\})^T\} = E\{\mathbf{v}\mathbf{v}^T\}, \quad (3.10)$$

which is usually unknown. Nevertheless, this estimator will be of great importance in the sequel since in Section 3.3.2 we will derive a recursive algorithm with which the RIR and the near-end signal characteristics can be estimated concurrently.

### 3.3 Prediction Error Identification Algorithms

Obviously, the BLUE as presented in Section 3.2 is unrealizable in an AEC scenario because the near-end signal characteristics (and hence  $\mathbf{R}^{-1}$ ) are unknown. Moreover the estimators from the previous section are non-recursive and do not allow for tracking of time-varying RIRs. In this section, we will derive a recursive algorithm, which incorporates a RIR estimator that approaches the BLUE, by concurrently estimating the near-end signal characteristics.

We will start this section by establishing a formal link between the BLUE and the prediction error (PE) estimate of the RIR  $F(q, t)$ . We will then propose a new recursive prediction error (RPE) identification algorithm for concurrent estimation of the RIR and an autoregressive model of the near-end signal. Finally, we will review three prediction error identification algorithms (the PEM-AF, PEM-AFROW, and 2ch-AF algorithms), that were proposed in an adaptive feedback cancellation (AFC) framework, and show how they can be applied to the AEC problem.

### 3.3.1 Prediction Error Criterion

Throughout this article, we will employ a particular model structure to the system under study:

$$y(t) = F(q, t)u(t) + \frac{1}{A(q, t)}e(t) \quad (3.11)$$

with  $F(q, t)$  and  $A(q, t)$  representing finite-order polynomials in  $q$  with possibly time-varying coefficients:

$$F(q, t) = f_0(t) + f_1(t)q^{-1} + \dots + f_{n_F}(t)q^{-n_F} \quad (3.12)$$

$$A(q, t) = 1 + a_1(t)q^{-1} + \dots + a_{n_A}(t)q^{-n_A}. \quad (3.13)$$

The above choice of model structure will allow us to derive a truly recursive prediction error identification algorithm, which would not have been possible if either the room model were an infinite impulse response (IIR) filter or the near-end signal model would contain a moving average (MA) part (see Section 3.3.3). The signal  $e(t)$  represents the excitation signal generating the near-end signal, and is assumed to be a white noise signal with a time-dependent variance  $\sigma_t^2$ :

$$E\{e(t)e(t-k)\} = \sigma_t^2\delta(k). \quad (3.14)$$

The best one-step-ahead predictor for  $y(t)$  is given by [27, Ch. 7]

$$\hat{y}[t|\boldsymbol{\theta}(t)] = [1 - A(q, t)]y(t) + A(q, t)F(q, t)u(t), \quad (3.15)$$

which depends on the unknown coefficients of the polynomials  $F(q, t)$  and  $A(q, t)$  that are collected in the  $(n_F + n_A + 1) \times 1$  parameter vector  $\boldsymbol{\theta}(t)$ , defined as

$$\boldsymbol{\theta}(t) \triangleq \begin{bmatrix} \mathbf{f}(t) \\ \mathbf{a}(t) \end{bmatrix} \quad (3.16)$$

with

$$\mathbf{f}(t) \triangleq [f_0(t) \quad f_1(t) \quad \dots \quad f_{n_F}(t)]^T, \quad (3.17)$$

$$\mathbf{a}(t) \triangleq [a_1(t) \quad a_2(t) \quad \dots \quad a_{n_A}(t)]^T. \quad (3.18)$$

The prediction error, defined as

$$\varepsilon[t, \boldsymbol{\theta}(t)] \triangleq y(t) - \hat{y}[t|\boldsymbol{\theta}(t)], \quad (3.19)$$

can be calculated with (3.15) as

$$\varepsilon[t, \boldsymbol{\theta}(t)] = A(q, t)y(t) - A(q, t)F(q, t)u(t) \quad (3.20)$$

$$= A(q, t)[y(t) - F(q, t)u(t)]. \quad (3.21)$$

Hence a prediction error criterion for estimating  $\boldsymbol{\theta}(t)$  is given by [27, Ch. 7]

$$\min_{\hat{\boldsymbol{\theta}}(t)} V_{\text{PE}}[t, \hat{\boldsymbol{\theta}}(t)] = \min_{\hat{\boldsymbol{\theta}}(t)} \frac{1}{2} E\{\varepsilon^2[t, \hat{\boldsymbol{\theta}}(t)]\}, \quad (3.22)$$

with  $\hat{\boldsymbol{\theta}}(t)$  denoting an estimate of  $\boldsymbol{\theta}(t)$ . When the expectation is replaced by a summation over the entire time interval on which data are available and appropriate weighting is included, the criterion can be rewritten, using (3.21), as

$$\min_{\hat{\boldsymbol{\theta}}(t)} \frac{1}{2N} \sum_{k=1}^t \frac{\lambda^{t-k}}{\hat{\sigma}_k^2} \{ \hat{A}(q, t)[y(k) - \hat{F}(q, t)u(k)] \}^2, \quad (3.23)$$

where  $\hat{A}(q, t)$  and  $\hat{F}(q, t)$  represent estimates of  $A(q, t)$  and  $F(q, t)$ , respectively, and the weighting factors  $\hat{\sigma}_k^{-2}$  and  $\lambda^{t-k}$  are defined as follows. The weighting with the inverse of the estimated prediction error variance  $\hat{\sigma}_k^2$  accounts for energy variations in the near-end excitation signal  $e(k)$ . On the other hand, an exponential forgetting profile with forgetting factor  $\lambda$  is included to discount old data and hence to allow for tracking of  $F(q, t)$  and  $A(q, t)$ . The effective window length  $N = 1/(1 - \lambda)$  is then determined by the forgetting factor  $\lambda$ . The rest of this section will be devoted to the recursive minimization of the prediction error criterion (3.23).

A formal relationship between the BLUE and the PE estimate of the room impulse response  $F(q, t)$  can be derived only if the near-end signal characteristics are assumed to be known. In [28] it was shown that minimization of the prediction error criterion (3.23) with respect to  $\hat{\mathbf{f}}(t)$  only, and with  $\lambda = 1$ , yields the best (minimum-variance) linear unbiased estimate of  $\mathbf{f}(t)$  if the near-end signal  $v(k)$  can at each time instant  $k \in [1, t]$  be modeled as an AR process of order  $n_A$ , if the prefilter  $\hat{A}(q, k)$  contains at each time instant the true AR coefficients  $a_i(k)$ ,  $i = 1, \dots, n_A$ , and if the weight  $\hat{\sigma}_k^{-2}$  is at each time instant equal to the inverse variance of the near-end excitation signal  $e(k)$ , i.e.,  $\hat{\sigma}_k^2 = \sigma_k^2$ .

### 3.3.2 Recursive Prediction Error (RPE) Identification Algorithm

We now derive a recursive prediction error (RPE) identification algorithm for estimating the parameter vector  $\boldsymbol{\theta}(t)$  defined in (3.16), which contains the coefficients of  $F(q, t)$  and  $A(q, t)$ . A general update equation for recursive minimization of the prediction error criterion in (3.22) with respect to  $\boldsymbol{\theta}(t)$  is given by [25, Ch. 3]

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \frac{N}{\hat{\sigma}_t^2} \gamma(t) \mathbf{R}^{-1}(t) \boldsymbol{\psi}[t, \hat{\boldsymbol{\theta}}(t-1)] \varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)], \quad (3.24)$$

where the terms are defined as follows. In the update term on the right hand side of (3.24),  $\mathbf{R}(t)$  should approximate the Hessian  $\partial^2 / \partial \boldsymbol{\theta}^2(t) V_{\text{PE}}[t, \boldsymbol{\theta}(t)]$  of the

criterion (3.22), evaluated in  $\boldsymbol{\theta}(t) = \hat{\boldsymbol{\theta}}(t-1)$ . The  $(n_F + n_A + 1) \times 1$  gradient vector  $\boldsymbol{\psi}[t, \boldsymbol{\theta}(t)]$  is defined as

$$\boldsymbol{\psi}[t, \boldsymbol{\theta}(t)] \triangleq -\frac{\partial}{\partial \boldsymbol{\theta}(t)} \varepsilon[t, \boldsymbol{\theta}(t)] = \frac{\partial}{\partial \boldsymbol{\theta}(t)} \hat{y}[t|\boldsymbol{\theta}(t)], \quad (3.25)$$

where the second equality is obtained by use of (3.19), in which the term  $y(t)$  is independent of  $\boldsymbol{\theta}(t)$ . The gain sequence  $\gamma(t)$  depends on the type of data windowing that is applied. In the sequel, we will choose  $\gamma(t) = 1/N$  to compensate for the factor  $N$  in (3.24), consistent with the definition of  $\mathbf{R}(t)$  in (3.32) below.

As pointed out in [26], it is highly desirable to identify  $F(q, t)$  and  $A(q, t)$  on data windows of different sizes. The RIR  $F(q, t)$  will typically be a high-order polynomial requiring a large data window for accurate estimation. On the other hand, the near-end signal AR polynomial  $A(q, t)$  may be highly time-varying, hence a small data window is desirable for estimation of the AR coefficients. It will be shown below that such a decoupling of the data windows for estimating  $F(q, t)$  and  $A(q, t)$  can be achieved by block-diagonalizing the matrix  $\mathbf{R}(t)$ . First however we explore the calculation of the gradient vector  $\boldsymbol{\psi}[t, \boldsymbol{\theta}(t)]$ .

#### Calculation of Gradient Vector $\boldsymbol{\psi}[t, \boldsymbol{\theta}(t)]$

The gradient vector  $\boldsymbol{\psi}[t, \hat{\boldsymbol{\theta}}(t-1)]$ , as it appears in (3.24), can be calculated by differentiating the one-step-ahead predictor (3.15) with respect to the coefficients of  $F(q, t)$  and  $A(q, t)$ , and replacing  $A(q, t)$  and  $F(q, t)$  by their estimates  $\hat{A}(q, t-1)$  and  $\hat{F}(q, t-1)$  at time  $t-1$ , i.e.,

$$\boldsymbol{\psi}[t, \hat{\boldsymbol{\theta}}(t-1)] = \begin{bmatrix} \hat{A}(q, t-1)u(t) \\ \vdots \\ \hat{A}(q, t-1)u(t-n_F) \\ \hat{F}(q, t-1)u(t-1) - y(t-1) \\ \vdots \\ \hat{F}(q, t-1)u(t-n_A) - y(t-n_A) \end{bmatrix}, \quad (3.26)$$

or, in partitioned matrix notation,

$$\boldsymbol{\psi}[t, \hat{\boldsymbol{\theta}}(t-1)] \triangleq \begin{bmatrix} \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] \\ \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \end{bmatrix} \quad (3.27)$$

$$= \begin{bmatrix} \mathbf{0}_{(n_F+1) \times (n_F+1)} & \mathbf{U}(t) \\ \mathbf{U}^T(t) & \mathbf{0}_{n_A \times n_A} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{f}}(t-1) \\ \hat{\mathbf{a}}(t-1) \end{bmatrix} + \begin{bmatrix} \mathbf{u}(t) \\ -\mathbf{y}(t) \end{bmatrix} \quad (3.28)$$

with

$$\mathbf{y}(t) \triangleq [y(t-1) \quad \dots \quad y(t-n_A)]_{n_A \times 1}^T, \quad (3.29)$$

$$\mathbf{u}(t) \triangleq [u(t) \quad \dots \quad u(t - n_F)]_{(n_F+1) \times 1}^T, \quad (3.30)$$

$$\mathbf{U}(t) \triangleq \begin{bmatrix} u(t-1) & \dots & u(t-n_A) \\ \vdots & \ddots & \vdots \\ u(t-n_F-1) & \dots & u(t-n_F-n_A) \end{bmatrix}_{(n_F+1) \times n_A} \quad (3.31)$$

and  $\hat{\mathbf{f}}(t-1)$  and  $\hat{\mathbf{a}}(t-1)$  denoting the most recent estimates of parameter vectors  $\mathbf{f}(t)$  and  $\mathbf{a}(t)$  defined in (3.17) and (3.18), respectively.

### Block-Diagonal Approximate Gauss-Newton Update of $\mathbf{R}(t)$

The matrix  $\mathbf{R}(t)$  determines the search direction of the recursive identification algorithm (3.24). It should be chosen positive definite and ideally, i.e., to obtain a Newton update direction, equal to the Hessian of the prediction error criterion (3.22), evaluated at the most recent parameter vector estimate. A common approximation is to retain only the term made up of gradient vector outer products in the calculation of the Hessian, which results in a Gauss-Newton update instead of a Newton update [25, Ch. 3]. Defining

$$\mathbf{R}(t) \triangleq N \left\{ \frac{\partial^2}{\partial \boldsymbol{\theta}^2(t)} V_{\text{PE}}[t, \boldsymbol{\theta}(t)] \right\} \Big|_{\boldsymbol{\theta}(t) = \hat{\boldsymbol{\theta}}(t-1)}, \quad (3.32)$$

a Gauss-Newton update equation for  $\mathbf{R}(t)$  can be obtained:

$$\mathbf{R}(t) = \lambda \mathbf{R}(t-1) + \left\{ \frac{1}{\hat{\sigma}_t^2} \boldsymbol{\psi}[t, \boldsymbol{\theta}(t)] \boldsymbol{\psi}^T[t, \boldsymbol{\theta}(t)] \right\} \Big|_{\boldsymbol{\theta}(t) = \hat{\boldsymbol{\theta}}(t-1)}. \quad (3.33)$$

As suggested in [14], decoupling of the data windows on which  $\mathbf{f}(t)$  and  $\mathbf{a}(t)$  are estimated can be achieved by forcing the off-diagonal blocks of  $\mathbf{R}(t)$  to zero and introducing two different exponential forgetting factors  $\lambda_F$  and  $\lambda_A$ , resulting in

$$\begin{bmatrix} \mathbf{R}_F(t) & \mathbf{0}_{(n_F+1) \times n_A} \\ \mathbf{0}_{n_A \times (n_F+1)} & \mathbf{R}_A(t) \end{bmatrix} = \begin{bmatrix} \lambda_F \mathbf{R}_F(t-1) & \mathbf{0}_{(n_F+1) \times n_A} \\ \mathbf{0}_{n_A \times (n_F+1)} & \lambda_A \mathbf{R}_A(t-1) \end{bmatrix} + \frac{1}{\hat{\sigma}_t^2} \begin{bmatrix} \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] \boldsymbol{\psi}_F^T[t, \hat{\mathbf{a}}(t-1)] & \mathbf{0}_{(n_F+1) \times n_A} \\ \mathbf{0}_{n_A \times (n_F+1)} & \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \boldsymbol{\psi}_A^T[t, \hat{\mathbf{f}}(t-1)] \end{bmatrix}. \quad (3.34)$$

When applied to the update equation (3.24), this block-diagonalization of  $\mathbf{R}(t)$  results in an approximate Gauss-Newton update direction for  $\hat{\boldsymbol{\theta}}(t)$ . The block-diagonalization was also suggested earlier for the recursive generalized least squares (RGLS) algorithm described in [29] and [30], which is a recursive implementation of the GLS estimator proposed in [31]. Setting the off-diagonal blocks to zero may also be justified by noting that if  $\hat{\mathbf{f}}(t-1)$  and  $\hat{\mathbf{a}}(t-1)$  have converged to their true values  $\mathbf{f}(t)$  and  $\mathbf{a}(t)$ , respectively, then

$$E \left\{ \boldsymbol{\psi}_F[t, \mathbf{a}(t)] \boldsymbol{\psi}_A^T[t, \mathbf{f}(t)] \right\} = E \left\{ \boldsymbol{\psi}_F[t, \mathbf{a}(t)] [v(t-1) \quad \dots \quad v(t-n_A)] \right\}$$

$$= \mathbf{0}_{(n_F+1) \times n_A}, \quad (3.35)$$

since  $\boldsymbol{\psi}_F[t, \mathbf{a}(t)]$  depends only on the loudspeaker signal  $u(t)$ , which was assumed to be uncorrelated with the near-end signal  $v(t)$  in (3.8), and

$$E\left\{\varepsilon[t, \boldsymbol{\theta}(t)]\mathbf{U}(t)\right\} = E\left\{e(t)\mathbf{U}(t)\right\} = \mathbf{0}_{(n_F+1) \times n_A}, \quad (3.36)$$

since  $e(t)$  is a white noise signal uncorrelated with the loudspeaker signal  $u(t)$ .

Finally, note that the block-diagonal approximation (3.34) of  $\mathbf{R}(t)$  is positive semi-definite by construction and will hence always lead to a recursive estimator  $\hat{\boldsymbol{\theta}}(t)$  that converges to a local minimum of the prediction error criterion (3.23), but not necessarily to the global minimum. The proposed recursive prediction error (RPE) identification algorithm is summarized in Table 3.1, making use of (3.21), (3.24), (3.28), and (3.34), and explicitly decoupling all expressions for estimation of  $\mathbf{f}(t)$  and  $\mathbf{a}(t)$ . Note that the prediction error variance  $\hat{\sigma}_\varepsilon^2$  is estimated by means of a single update equation and on the same data window as the AR polynomial  $A(q, t)$ , since both depend on the near-end signal characteristics.

### 3.3.3 Prediction Error Identification Algorithms from AFC

#### Prediction-Error-Method-Based Adaptive Filtering (PEM-AF) Algorithm

A common approximation in recursive system identification [25, Ch. 3] consists of replacing the estimates  $\hat{A}(q, t-1)$  and  $\hat{F}(q, t-1)$  in (3.26) by previously obtained estimates, e.g.,

$$\boldsymbol{\psi}(t) = \begin{bmatrix} \hat{A}(q, t-1)u(t) \\ \vdots \\ \hat{A}(q, t-1-n_F)u(t-n_F) \\ \hat{F}(q, t-2)u(t-1) - y(t-1) \\ \vdots \\ \hat{F}(q, t-1-n_A)u(t-n_A) - y(t-n_A) \end{bmatrix}. \quad (3.37)$$

The motivation behind this approximation is that in a generic recursive identification scenario, the system model is an IIR filter and the noise model contains a moving average (MA) part:

$$y(t) = \frac{F(q, t)}{D(q, t)}u(t) + \frac{C(q, t)}{A(q, t)}e(t). \quad (3.38)$$

If either one of the polynomials  $D(q, t)$  and  $C(q, t)$  is included in the model structure, the one-step-ahead predictor  $\hat{y}[t|\boldsymbol{\theta}(t)]$  and hence the gradient vector

Table 3.1: Summary of the RPE algorithm

---

**RPE - Recursive prediction error identification algorithm**


---

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \frac{1}{\hat{\sigma}_t^2} \mathbf{R}_F^{-1}(t) \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] \varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]$$

$$\hat{\mathbf{a}}(t) = \hat{\mathbf{a}}(t-1) + \frac{1}{\hat{\sigma}_t^2} \mathbf{R}_A^{-1}(t) \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]$$

$$\varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)] = \begin{bmatrix} 1 & \hat{\mathbf{a}}^T(t-1) \end{bmatrix} \left( \begin{bmatrix} y(t) \\ \mathbf{y}(t) \end{bmatrix} - \begin{bmatrix} \mathbf{u}^T(t) \\ \mathbf{U}^T(t) \end{bmatrix} \hat{\mathbf{f}}(t-1) \right)$$

$$\hat{\sigma}_t^2 = \lambda_A \hat{\sigma}_{t-1}^2 + (1 - \lambda_A) \varepsilon^2[t, \hat{\boldsymbol{\theta}}(t-1)]$$

$$\boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] = \begin{bmatrix} \mathbf{u}(t) & \mathbf{U}(t) \end{bmatrix} \begin{bmatrix} 1 \\ \hat{\mathbf{a}}(t-1) \end{bmatrix}$$

$$\boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] = \mathbf{U}^T(t) \hat{\mathbf{f}}(t-1) - \mathbf{y}(t)$$

$$\mathbf{R}_F(t) = \lambda_F \mathbf{R}_F(t-1) + \frac{1}{\hat{\sigma}_t^2} \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] \boldsymbol{\psi}_F^T[t, \hat{\mathbf{a}}(t-1)]$$

$$\mathbf{R}_A(t) = \lambda_A \mathbf{R}_A(t-1) + \frac{1}{\hat{\sigma}_t^2} \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \boldsymbol{\psi}_A^T[t, \hat{\mathbf{f}}(t-1)]$$

Refer to (3.16), (3.17), (3.18), (3.29), (3.30), and (3.31) for definitions of the parameter vectors and data matrices.

---



$\psi[t, \boldsymbol{\theta}(t)]$  will be made up of data filtered by IIR filters. An exact calculation of  $\psi[t, \hat{\boldsymbol{\theta}}(t-1)]$  as in (3.26) would then require the complete data record to be filtered with the most recent estimates  $\hat{D}(q, t-1)$  and/or  $\hat{C}(q, t-1)$ , and hence would not allow for a truly recursive identification algorithm. However, this problem does not occur for the model structure (3.11) under study, as pointed out earlier in [32].

Nevertheless, even with the model structure (3.11), the approximation in (3.37) is appealing from a complexity point of view. As pointed out in [14], using (3.37) instead of (3.26) saves  $n_F + n_A - 1$  filtering operations in each recursion step, leading to a cheaper, so-called PEM-AF algorithm. However, this approximation is fully justified only when

$$\begin{cases} \hat{A}(q, t-1) = \hat{A}(q, t-1-i), & i \in [0, n_F], \\ \hat{F}(q, t-1) = \hat{F}(q, t-1-j), & j \in [1, n_A]. \end{cases} \quad (3.39)$$

$$(3.40)$$

As mentioned before, in an AEC scenario,  $F(q, t)$  will typically be of very high order and so  $A(q, t)$  may be highly time-varying over a data window of length  $n_F$ . Hence the assumption in (3.39) will not hold for the current application, in contrast to the hearing aid AFC application considered in [14]. Also, in the case of a time-varying RIR, the assumption in (3.40) may not hold.

### Prediction-Error-Method-Based Adaptive Filtering Algorithm with Row Operations (PEM-AFROW)

Instead of identifying the near-end signal AR coefficients recursively, as in the RPE and PEM-AF algorithms described above, the AR model can also be identified non-recursively on a batch of loudspeaker and microphone data. This may be interesting from a computational point of view if the near-end signal exhibits short-term stationary behaviour, because then the AR model does not have to be recalculated at each time instant. This is the idea behind the PEM-AFROW algorithm, which was originally developed in an AFC framework [15] and applied to a continuous double-talk AEC scenario in [28]. This algorithm, like the RPE algorithm, performs only row operations on the loudspeaker data matrix, hence the name PEM-AFROW. The PEM-AFROW algorithm is best suited for applications involving a near-end speech signal, since speech exhibits short-term stationary behaviour.

### Two-Channel Adaptive Filtering (2ch-AF) Algorithm

All three prediction error identification algorithms described above aim at direct minimization of the criterion in (3.23), which is non-linear in the coefficients of  $F(q, t)$  and  $A(q, t)$ ; hence they may suffer from convergence to local minima. In the two-channel adaptive filtering (2ch-AF) algorithm proposed in [14], it is noted that expression (3.20) for the prediction error can be written

as

$$\varepsilon[t, \boldsymbol{\theta}_{2\text{ch}}(t)] = A(q, t)y(t) + B(q, t)u(t) \quad (3.41)$$

with

$$B(q, t) \triangleq -A(q, t)F(q, t) \quad (3.42)$$

and

$$\boldsymbol{\theta}_{2\text{ch}}(t) \triangleq [b_0(t) \quad \dots \quad b_{n_B}(t) \quad a_1(t) \quad \dots \quad a_{n_A}(t)]^T, \quad (3.43)$$

with  $n_B = n_A + n_F$ . This interpretation leads to a criterion that is quadratic in the parameter vector  $\boldsymbol{\theta}_{2\text{ch}}(t)$ , so that an ordinary recursive least squares (RLS) algorithm can be applied.

The main drawback of the 2ch-AF algorithm is that the coefficients of  $A(q, t)$  and  $F(q, t)$ , which are combined into  $B(q, t)$ , are estimated on the same data window. As pointed out in [26] for the AFC case, this makes the algorithm, which was originally derived for hearing aid applications, unsuitable when dealing with room acoustics. Another drawback is that  $F(q, t)$  can only be obtained by deconvolution of  $A(q, t)$  and  $B(q, t)$ , which adds a complexity of  $O(n_F^2)$  to the algorithm. Finally, we note that the interpretation of the prediction error as in (3.41)-(3.42) was made earlier in [33] as an alternative to the generalized least squares (GLS) algorithm proposed in [31].

## 3.4 Implementation and Complexity

### 3.4.1 Implementation Issues

The RPE, PEM-AF, and PEM-AFROW algorithms described in Section 3.3 are implemented both with a Gauss-Newton update and a cheaper stochastic gradient update of the RIR estimate  $\hat{\mathbf{f}}(t)$ . The same is done for the update of the coefficients of  $B(q, t)$  in the 2ch-AF algorithm. The estimate  $\hat{\mathbf{a}}(t)$  is always updated according to the Gauss-Newton method in the RPE, PEM-AF, and 2ch-AF algorithms. In the PEM-AFROW algorithm,  $\hat{\mathbf{a}}(t)$  and  $\hat{\sigma}_t^2$  are calculated using the Levinson-Durbin recursion [34, Ch. 6].

#### Gauss-Newton Algorithms

For computational efficiency, all Gauss-Newton updates are performed by using the Kalman gain vectors and applying the matrix inversion lemma [34, Ch. 13]. Typically, the inverse correlation matrix will be initialized as  $\mathbf{R}_F^{-1}(0) = 10^6 \cdot \mathbf{I}_{n_F+1}$  and the RIR estimate as  $\hat{\mathbf{f}}(0) = \mathbf{0}_{(n_F+1) \times 1}$ .

### Stochastic Gradient Algorithms

The stochastic gradient implementation of the algorithms features a normalization with estimates of both the loudspeaker signal variance (as in the NLMS algorithm) and the near-end signal variance (as in the algorithms described in Section 3.3). Simulations have shown that the sum method proposed in [35] for combining these two normalization factors yields the best convergence behaviour. As an example, a stochastic gradient implementation of the RPE update equations in Table 3.1 can be formulated as

$$\begin{aligned} \hat{\mathbf{f}}(t) &= \hat{\mathbf{f}}(t-1) \\ &+ \mu_F \frac{\boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)]\varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]}{\delta_F + (n_F + 1)\hat{\sigma}_t^2 + \boldsymbol{\psi}_F^T[t, \hat{\mathbf{a}}(t-1)]\boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)}} \end{aligned} \quad (3.44)$$

where  $\mu_F$  denotes the step size and  $\delta_F$  the regularization parameter. Note that the weighted sum method proposed in [35] was applied in [36] to improve double-talk-robustness in an acoustic echo canceller.

### Choice of Near-End Model Order

Simulations have shown that the choice of the near-end model order  $n_A$  has a profound influence on the performance of the prediction error identification algorithms described in Section 3.3. The algorithms were derived under a sufficient-order assumption, yet in practice this assumption will not be fulfilled. Due to the non-stationary nature of speech and audio signals, the true model order is not well-defined, let alone that it would be known to the user. Moreover, the larger  $n_A$  is chosen, the larger the number of local minima in the prediction error cost function (3.23).

We have evaluated four different choices of  $n_A$  by computer simulations at  $f_s = 8$  kHz: a high-order model ( $n_A = 55$ ) to capture all near-end signal dynamics, a low-order model ( $n_A = 12$ ) as used for formant prediction in speech coding, a low-order model ( $n_A = 12$ ) cascaded with a 3-tap pitch predictor as used in speech coding [37], and a very-low-order model ( $n_A = 1$ ) as used in the AFC algorithm proposed in [38]. It appears that the very-low-order model ( $n_A = 1$ ) yields the best performance in all scenarios that were tested. In other words, even if the near-end signal component  $v(t)$  contained in the microphone signal  $y(t)$  is undermodeled, convergence speed is gained due to the more convex shape of the prediction error cost function when  $n_A$  is small.

### 3.4.2 Computational Complexity

To compare the computational complexity of the prediction error identification algorithms described in Section 3.3 with the complexity of standard RLS and NLMS adaptive algorithms, we show the number of floating point operations

Table 3.2: Complexity Comparison by Number of FLOPS per Recursion

Algorithm	Number of FLOPS per recursion	
Gauss-Newton algorithms		
RLS	$5L_F^2 + 5L_F$	5005000
RPE	$5L_F^2 + (4n_A + 5)L_F + 5n_A^2 + 5n_A + 6$	5009016
PEM-AF	$5L_F^2 + 5L_F + 5n_A^2 + 7n_A + 6$	5005018
PEM-AFROW	$5L_F^2 + \left(5 + \frac{2M + 2n_A + 1}{P}\right)L_F + \frac{1}{P}n_A^2 + \left(4 + \frac{2M + 2}{P}\right)n_A + \frac{2M}{P} + \frac{P - 1}{P} + 2$	5007000
2ch-AF	$6L_F^2 + (20n_A + 4)L_F + 20n_A^2 + 10n_A + 5$	6024035
Stochastic gradient algorithms		
NLMS	$6L_F + 2$	6002
RPE	$(4n_A + 6)L_F + 5n_A^2 + 5n_A + 10$	10020
PEM-AF	$6L_F + 5n_A^2 + 7n_A + 10$	6022
PEM-AFROW	$\left(6 + \frac{2M + 2n_A + 1}{P}\right)L_F + \frac{1}{P}n_A^2 + \left(4 + \frac{2M + 2}{P}\right)n_A + \frac{2M}{P} + \frac{P - 1}{P} + 6$	8046
2ch-AF	$L_F^2 + 5L_F + 5n_A^2 + 11n_A + 10$	1005026

(FLOPS) per recursion step in Table 3.2. The complexity equations in Table 3.2 are evaluated with the values of the parameters used in the simulations described in Section 3.5:  $L_F = n_F + 1 = 1000$ ,  $n_A = 1$ ,  $P = 160$  and  $M = P + n_A = 161$ .

Note that the Gauss-Newton algorithms are of comparable complexity if  $n_A$  is small, except for the 2ch-AF algorithm, which requires additional computations for the deconvolution of the filters  $A(q, t)$  and  $B(q, t)$ . The stochastic gradient PEM-AF algorithm has a complexity comparable to the NLMS algorithm, whereas the RPE and PEM-AFROW algorithms require somewhat more computations due to the exact calculation of the prediction error and gradient vectors (in contrast to the approximate calculation in the PEM-AF algorithm). The stochastic gradient 2ch-AF algorithm is computationally not interesting due to the deconvolution with complexity  $O(L_F^2)$ . Since no assumptions on the short-term stationarity of the near-end signal are made in the RPE algorithm, its complexity rises somewhat faster than that of the PEM-AF and PEM-AFROW algorithms as  $n_A$  is increased.

### 3.5 Simulation Results

MATLAB simulations were performed to compare the convergence behaviour of the four prediction error algorithms described in Section 3.3 with some reference algorithms in different scenarios. The Gauss-Newton reference algorithms are the standard RLS algorithm and the double-talk-robust non-linear recursive least squares (NL-RLS) algorithm proposed in [10, Ch. 6]. The stochastic gradient reference algorithms are the standard NLMS algorithm and a stochastic gradient implementation of the NL-RLS algorithm. It should be noted that the NL-RLS algorithm was designed to be used in combination with a DTD, in contrast to the prediction error identification algorithms. To focus on the double-talk-robustness of the adaptive algorithms, we have chosen not to include a DTD in any of the simulations because the results would then depend heavily on the DTD quality.

All simulations are performed at a sampling rate  $f_s = 8$  kHz. The room impulse response  $F(q, t)$  of length  $L_F = n_F + 1 = 1000$  (corresponding to 125 ms) was measured in a typical office, and is plotted in Fig. 3.2. The microphone and loudspeaker are assumed to be linear devices with a flat frequency response. In all simulations, some low-level ( $-50$  dB) Gaussian white noise is added to the near-end signal to account for background noise, which is present in any realistic setup. The near-end signal model order is set to  $n_A = 1$  in most of the experiments, as motivated in Section 3.4.1. The exponential forgetting factors for estimation of  $\mathbf{f}(t)$  in the Gauss-Newton RPE, PEM-AF, and PEM-AFROW algorithms, and of  $\boldsymbol{\theta}_{2\text{ch}}(t)$  in the 2ch-AF algorithm, are chosen such that the

effective data window length is 3 times the parameter vector length, i.e.,  $\lambda_F = \lambda_{2\text{ch}} = 0.9997$ . The exponential forgetting factor for recursive estimation of  $\mathbf{a}(t)$  in the RPE, PEM-AF, and stochastic gradient 2ch-AF algorithms, and of  $\sigma_t^2$  in the RPE, PEM-AF, and 2ch-AF algorithms is  $\lambda_A = 0.9938$ , such that the effective data window length is 160 samples or 20 ms (which is the average window length in which a speech signal can be considered stationary). The PEM-AFROW window hop size is set to  $P = 160$  samples and the window length to  $M = P + n_A$ , as suggested in [15]. The step size for all stochastic gradient algorithms is equal to  $\mu_F = 0.5$  (which was found to give the best convergence behaviour on average) and the regularization parameter is  $\delta_F = 10^{-6}$ . In all the different scenarios, the near-end signal is scaled such that the echo-to-near-end ratio (ENR) equals 10 dB:

$$\text{ENR} \triangleq 10 \log_{10} \frac{\sum_{k=1}^N |x(k)|^2}{\sum_{k=1}^N |v(k)|^2} = 10 \text{ dB}, \quad (3.45)$$

with  $N$  the number of data points used. The simulation length  $N$  varies for different simulations, and will typically be much smaller for the Gauss-Newton simulations than for the stochastic gradient simulations. The parameters of the NL-RLS algorithm are chosen as suggested in [10, Ch. 6]: the saturation level is  $k_0 = 1.5$  and the exponential forgetting factor for estimation of the scale factor is  $\lambda_s = 0.9975$ . All algorithms, impulse responses, and sound signals are available for download [39].

The performance measure for evaluating the convergence behaviour is the misalignment, defined as the logarithmic normalized Euclidian distance between the true and estimated impulse response at each time instant:

$$\text{misalignment (dB)} = 20 \log_{10} \frac{\|\hat{\mathbf{f}}(t) - \mathbf{f}(t)\|}{\|\mathbf{f}(t)\|}. \quad (3.46)$$

In a first scenario, double-talk is continuously present. The far-end signal is a male speech signal plotted in Figs. 3.3(a) and 3.4(a) for the Gauss-Newton and stochastic gradient experiments, respectively. Note that the signals are drawn from the same speech fragment, but on different time scales. With the continuous double-talk applications described in Section 3.1 in mind, the near-end signal is a babble noise signal plotted in Figs. 3.3(b) and 3.4(b). Continuous double-talk simulations with a near-end speech signal are described in [28] and [40]. The Gauss-Newton and stochastic gradient convergence curves are drawn in Figs. 3.3(c) and 3.4(c), respectively, for a low-order near-end signal model ( $n_A = 12$ ) and in Figs. 3.3(d) and 3.4(d), respectively, for a very-low-order near-end signal model ( $n_A = 1$ ). A first, somewhat surprising, observation

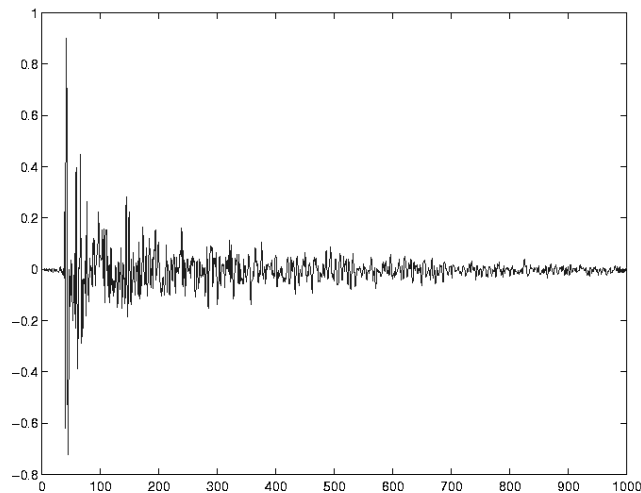


Figure 3.2: Room impulse response measured in a typical office.

is that the standard RLS algorithm performs in a relatively robust way in a continuous double-talk situation: the RLS algorithm outperforms all other algorithms. The standard NLMS algorithm on the other hand appears to be extremely sensitive to double-talk for  $\mu = 0.5$ , in contrast to the stochastic gradient RPE, PEM-AF, and PEM-AFROW algorithms. All prediction error identification algorithms exhibit improved convergence and more stable behaviour with the very-low-order near-end signal model ( $n_A = 1$ ). Finally, in all simulations, the NL-RLS algorithm stalls at an estimate far from the true impulse response. This is because the NL-RLS scale factor converges to zero during double-talk, thereby slowing down and eventually freezing the adaptation. This effect can be overcome by limiting the scale factor to a minimum value during double-talk, which is however impossible when no DTD is applied.

In a second scenario, double-talk only occurs after the adaptive algorithms have initially converged to some extent. The same male far-end speech signals as in the first scenario are used, whereas the near-end signal is now a short female speech signal, occurring in the middle of the experiment, as shown in Figs. 3.5(a) and 3.6(a). In this scenario we only use a very-low-order near-end signal model ( $n_A = 1$ ), which yields the convergence curves in Figs. 3.5(b) and 3.6(b). The Gauss-Newton RPE, PEM-AFROW, and 2ch-AF algorithms show approximately the same initial convergence during single-talk as the standard RLS algorithm, while the Gauss-Newton PEM-AF and NL-RLS algorithms perform worse. At the onset of double-talk (around  $t/T_s = 3000$  samples) the 2ch-AF and standard RLS algorithms diverge considerably. The PEM-AFROW and RPE algorithms behave remarkably robust, although they show some divergence when the near-end signal power increases. The standard RLS

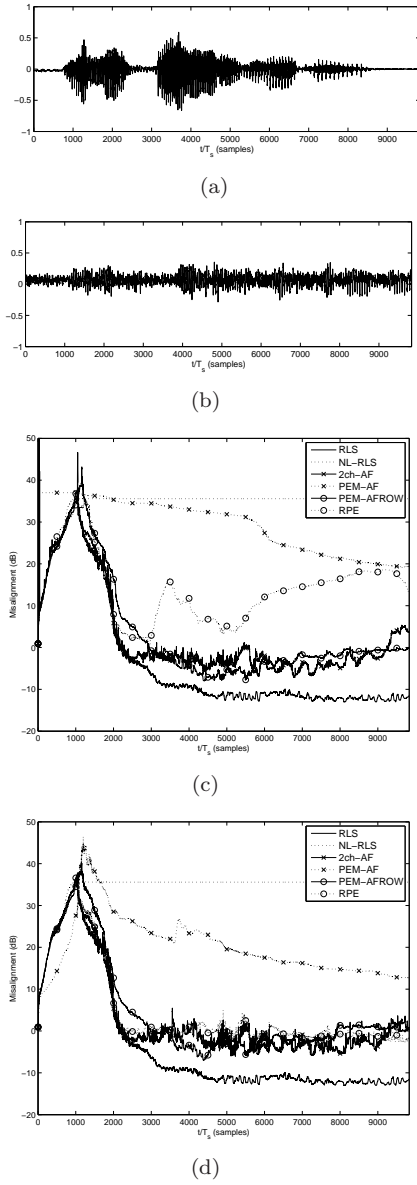


Figure 3.3: Continuous double-talk scenario: Gauss-Newton simulations. (a) Far-end speech signal. (b) Near-end babble noise signal. (c) Convergence curves for  $n_A = 12$ . (d) Convergence curves for  $n_A = 1$ .

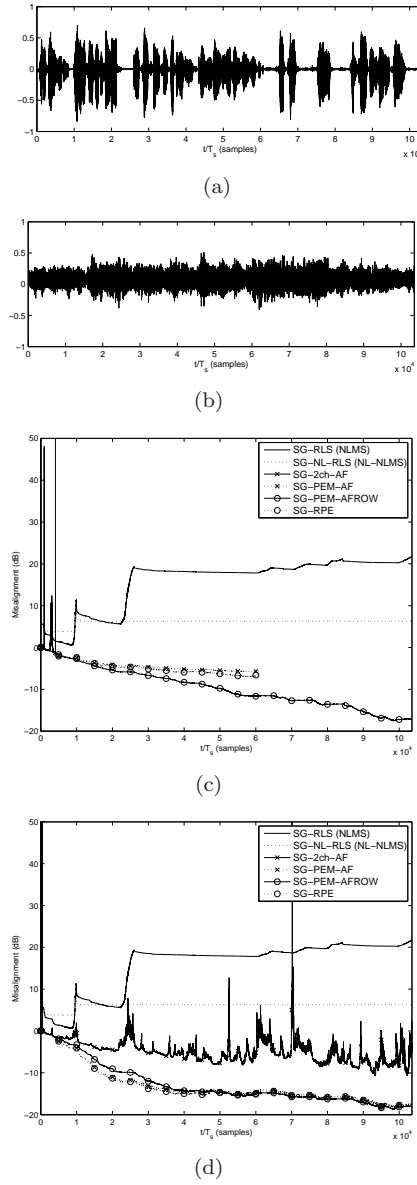


Figure 3.4: Continuous double-talk scenario: stochastic gradient simulations. (a) Far-end speech signal. (b) Near-end babble noise signal. (c) Convergence curves for  $n_A = 12$ . (d) Convergence curves for  $n_A = 1$ .



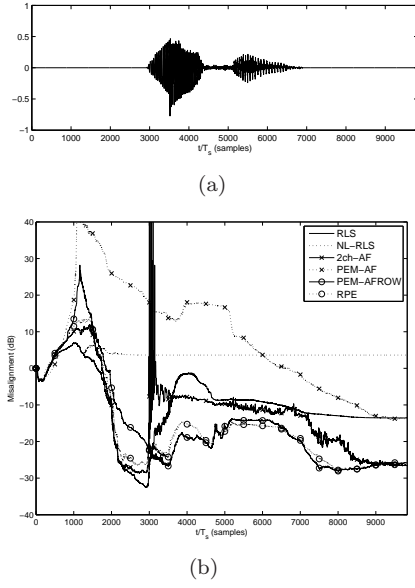


Figure 3.5: Bursting double-talk scenario: Gauss-Newton simulations. (a) Near-end speech signal. (b) Convergence curves for  $n_A = 1$ .

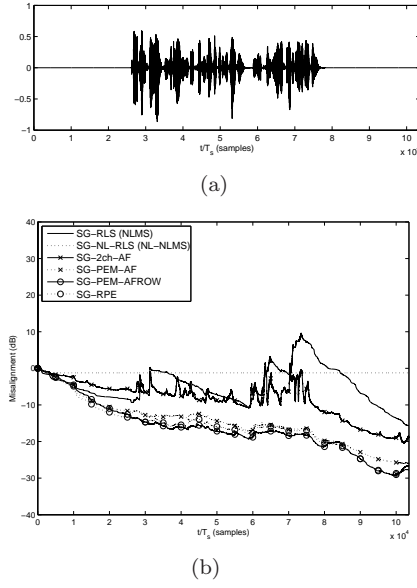


Figure 3.6: Bursting double-talk scenario: stochastic gradient simulations. (a) Near-end speech signal. (b) Convergence curves for  $n_A = 1$ .

algorithm clearly reacts in a less robust way to a sudden near-end signal onset than to the continuous double-talk situation in the first scenario. None of the Gauss-Newton algorithms returns to its initial convergence speed after the double-talk situation has passed, which could however be solved by resetting the adaptive filter memory at the end of such a double-talk period. The stochastic gradient RPE, PEM-AFROW, and PEM-AF algorithms clearly outperform all other stochastic gradient algorithms, and they do not even diverge at the onset of double-talk (around  $t/T_s = 27000$  samples).

In a third and fourth scenario, we investigate the behaviour of the algorithms when an echo path change occurs. In the middle of the experiment, the echo path gain is decreased by 6 dB, i.e., all RIR coefficients are halved. The far-end signals are the same as before. In the third scenario the echo path change occurs during single-talk. The convergence curves with  $n_A = 1$  are shown in Figs. 3.7 and 3.8. The Gauss-Newton algorithms are obviously non-robust to echo path changes. Memory resetting after an echo path change is necessary, and hence these algorithms should be equipped with an echo path change detector (as proposed, e.g., in [5], [41]-[43]). On the other hand, the stochastic gradient algorithms converge after an echo path change as fast as in the initial convergence phase. In the fourth scenario, the echo path change occurs during

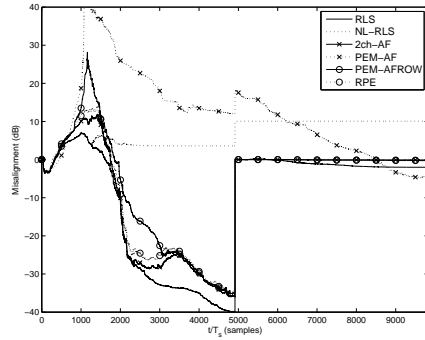


Figure 3.7: Echo path change in single-talk: Gauss-Newton simulations. Convergence curves for  $n_A = 1$ .

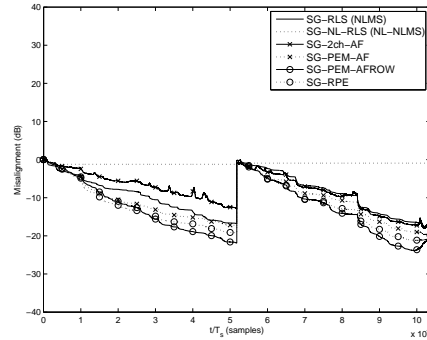


Figure 3.8: Echo path change in single-talk: stochastic gradient simulations. Convergence curves for  $n_A = 1$ .

double-talk (a worst-case scenario). The near-end signals are the same as in the bursting double-talk scenario. The convergence curves with  $n_A = 1$  are drawn in Figs. 3.9 and 3.10. The behaviour of the Gauss-Newton algorithms after the echo path change is comparable to the third scenario. However, the stochastic gradient RPE, PEM-AF, and PEM-AFROW algorithms converge remarkably faster after the echo path change than the other stochastic gradient algorithms.

### 3.6 Conclusions

We have proposed a new approach to improve double-talk-robustness in adaptive algorithms for acoustic echo cancellation. Minimization of a prediction error criterion that includes a model of the near-end signal leads to a minimum-variance type, fast-converging recursive echo path estimate. We have applied a new recursive prediction error (RPE) identification algorithm and three prediction error identification algorithms from adaptive feedback cancellation (PEM-AF, PEM-AFROW, and 2ch-AF) to different double-talk scenarios that may occur in echo cancellation. Especially in the stochastic gradient case, the prediction error identification algorithms outperform the standard NLMS algorithm, which is highly sensitive to double-talk. In most double-talk scenarios, the RPE and PEM-AFROW algorithms tend to show the fastest convergence behaviour with only a few additional computations and without sacrificing convergence speed during single-talk.

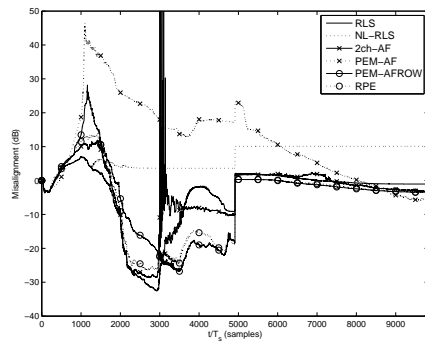


Figure 3.9: Echo path change in double-talk: Gauss-Newton simulations. Convergence curves for  $n_A = 1$ .

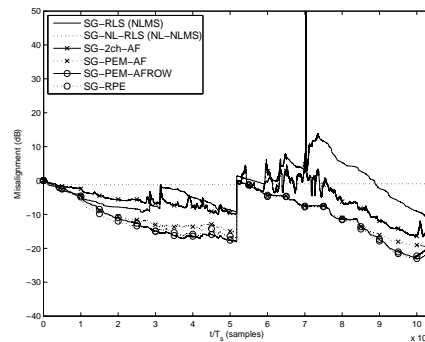


Figure 3.10: Echo path change in double-talk: stochastic gradient simulations. Convergence curves for  $n_A = 1$ .

## Bibliography

- [1] D. L. Duttweiler, "A twelve-channel digital echo canceler," *IEEE Trans. Commun.*, vol. 26, no. 5, pp. 647–653, May 1978.
- [2] J. H. Cho, D. R. Morgan, and J. Benesty, "An objective technique for evaluating doubletalk detectors in acoustic echo cancelers," *IEEE Trans. Speech Audio Process.*, vol. 7, no. 6, pp. 718–724, Nov. 1999.
- [3] J. Benesty, D. R. Morgan, and J. H. Cho, "A new class of doubletalk detectors based on cross-correlation," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 2, pp. 168–172, Mar. 2000.
- [4] K. Ghose and V. U. Reddy, "A double-talk detector for acoustic echo cancellation applications," *Signal Processing*, vol. 80, no. 8, pp. 1459–1467, Aug. 2000.
- [5] J. C. Jenq and S. F. Hsieh, "Decision of double-talk and time-variant echo path for acoustic echo cancelation," *IEEE Signal Process. Lett.*, vol. 10, no. 11, pp. 317–319, Nov. 2003.
- [6] T. Jia, Y. Jia, J. Li, and Y. Hu, "Subband doubletalk detector for acoustic echo cancellation systems," in *Proc. 2003 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '03)*, vol. 5, Apr. 2003, pp. 604–607.
- [7] M. Kallinger, A. Mertins, and K.-D. Kammeyer, "Enhanced double-talk detection based on pseudo-coherence in stereo," in *Proc. 2005 Int. Workshop Acoustic Echo Noise Control (IWAENC '05)*, Eindhoven, The Netherlands, Sep. 2005, pp. 177–180.

- [8] T. Gänsler, J. Benesty, S. L. Gay, and M. M. Sondhi, "A robust proportionate affine projection algorithm for network echo cancellation," in *Proc. 2000 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '00)*, vol. 2, Istanbul, Turkey, Jun. 2000, pp. 793–796.
- [9] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.
- [10] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*. Berlin: Springer-Verlag, 2001.
- [11] H. Buchner, J. Benesty, T. Gänsler, and W. Kellermann, "An outlier-robust extended multidelay filter with application to acoustic echo cancellation," in *Proc. 2003 Int. Workshop Acoustic Echo Noise Control (IWAENC '03)*, Kyoto, Japan, Sep. 2003, pp. 19–22.
- [12] W. Herbordt, H. Buchner, S. Nakamura, and W. Kellermann, "Application of a double-talk resilient DFT-domain adaptive filter for bin-wise stepsize controls to adaptive beamforming," in *Proc. 2005 IEEE-EURASIP Workshop Nonlinear Signal Image Process.*, Sapporo, Japan, May 2005.
- [13] M. M. Sondhi and D. A. Berkley, "Silencing echoes on the telephone network," *Proc. IEEE*, vol. 68, pp. 948–963, Aug. 1980.
- [14] A. Spriet, I. Proudler, M. Moonen, and J. Wouters, "Adaptive feedback cancellation in hearing aids with linear prediction of the desired signal," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3749–3763, Oct. 2005.
- [15] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, "Acoustic feedback suppression for long acoustic paths using a nonstationary source model," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.
- [16] G. Enzner and P. Vary, "Robust and elegant, purely statistical adaptation of acoustic echo canceler and postfilter," in *Proc. 2003 Int. Workshop Acoustic Echo Noise Control (IWAENC '03)*, Kyoto, Japan, Sep. 2003, pp. 43–46.
- [17] M. Kallinger, K.-D. Kammeyer, and J. Bitzer, "Multi-microphone residual echo estimation," in *Proc. 2003 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '03)*, vol. 5, Apr. 2003, pp. 584–587.
- [18] S. Y. Low and S. Nordholm, "A blind approach to joint noise and acoustic echo cancellation," in *Proc. 2005 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '05)*, vol. 3, Philadelphia, PA, USA, Mar. 2005, pp. 69–72.
- [19] J. F. Doherty and R. Porayath, "A robust echo canceler for acoustic environments," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 44, no. 5, pp. 389–396, May 1997.

- [20] T. Okuno, M. Fukushima, and M. Tohyama, “Adaptive cross-spectral technique for acoustic echo cancellation,” *IEICE Trans. Fundamentals*, vol. E82-A, no. 4, pp. 634–639, Apr. 1999.
- [21] X. Lu and B. Champagne, “Acoustic echo cancellation over a non-linear channel,” in *Proc. 2001 Int. Workshop Acoustic Echo Noise Control (IWAENC '01)*, Darmstadt, Germany, Sep. 2001, pp. 139–142.
- [22] —, “Pitch analysis-based acoustic echo cancellation over a nonlinear channel,” in *Proc. 11th European Signal Process. Conf. (EUSIPCO '02)*, Toulouse, France, Sep. 2002, pp. 159–162.
- [23] X. Hu, A.-Q. Hu, Y. Chen, and X.-H. Zeng, “An adaptive acoustic echo cancellation without double-talk detection,” in *Proc. 2003 Int. Conf. Machine Learning Cybern. (ICMLC '03)*, Xi'an, China, Nov. 2003, pp. 972–975.
- [24] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Upper Saddle River, New Jersey: Prentice-Hall, 1993.
- [25] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. Cambridge, Massachusetts: MIT Press, 1986.
- [26] T. van Waterschoot, G. Rombouts, and M. Moonen, “On the performance of decorrelation by prefiltering for adaptive feedback cancellation in public address systems,” in *Proc. 4th IEEE Benelux Signal Process. Symp. (SPS '04)*, Hilvarenbeek, The Netherlands, Apr. 2004, pp. 167–170. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/04-24.html>
- [27] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
- [28] T. van Waterschoot and M. Moonen, “Double-talk robust acoustic echo cancellation with continuous near-end activity,” in *Proc. 13th European Signal Process. Conf. (EUSIPCO '05)*, Antalya, Turkey, Sep. 2005.
- [29] R. Hastings-James and M. W. Sage, “Recursive generalised-least-squares procedure for online identification of process parameters,” *Proc. IEE*, vol. 116, no. 12, pp. 2057–2062, Dec. 1969.
- [30] T. Söderström, L. Ljung, and I. Gustavsson, “A theoretical analysis of recursive identification methods,” *Automatica*, vol. 14, no. 3, pp. 231–244, May 1978.
- [31] D. W. Clarke, “Generalized least squares estimation of the parameters of a dynamic model,” in *Proc. 1st IFAC Symp. Identification Autom. Control Syst.*, Prague, Czechoslovakia, 1967.

- [32] J. Gertler and C. Bányász, “A recursive (on-line) maximum likelihood identification method,” *IEEE Trans. Autom. Control*, vol. AC-19, no. 6, pp. 816–820, Dec. 1974.
- [33] T. Söderström, “Convergence properties of the generalized least squares identification method,” *Automatica*, vol. 10, no. 6, pp. 617–626, Dec. 1974.
- [34] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1996.
- [35] J. E. Greenberg, “Modified LMS algorithms for speech processing with an adaptive noise canceller,” *IEEE Trans. Speech Audio Process.*, vol. 6, no. 4, pp. 338–351, Jul. 2000.
- [36] X. Cheng, J. Xu, X. Li, and J. Tian, “Double-talk protection of acoustic echo cancellation based on weighted-sum NLMS algorithms,” in *Proc. 6th Int. Conf. Signal Process. (ICSP '02)*, vol. 1, Beijing, China, Aug. 2002, pp. 297–300.
- [37] R. P. Ramachandran and P. Kabal, “Pitch prediction filters in speech coding,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-37, no. 4, pp. 467–478, Apr. 1989.
- [38] R. Leber and A. Schaub, “Circuit and method for the adaptive suppression of an acoustic feedback,” U.S. Patent 6,611,600, Aug., 2003.
- [39] T. van Waterschoot, G. Rombouts, and M. Moonen, “Software for double-talk-robust acoustic echo cancellation,” Katholieke Universiteit Leuven, Belgium, Tech. Rep. ESAT-SISTA TR 05-162, Oct. 2005. [Online]. Available: <ftp://ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/05-162.html>
- [40] T. van Waterschoot, G. Rombouts, K. Struyve, and M. Moonen, “Acoustic echo cancellation in the presence of continuous double-talk,” in *Proc. 1st Annual IEEE Benelux/DSP Valley Signal Process. Symp. (SPS-DARTS '05)*, Antwerp, Belgium, Apr. 2005, pp. 53–57. [Online]. Available: <ftp://ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/04-149.html>
- [41] J. Huo, S. Nordholm, and Z. Zang, “A method for detecting echo path variation,” in *Proc. 2003 Int. Workshop Acoustic Echo Noise Control (IWAENC '03)*, Kyoto, Japan, Sep. 2003, pp. 71–74.
- [42] J. Liu, “Efficient and robust cancellation of echoes with long echo path delay,” *IEEE Trans. Commun.*, vol. 52, no. 8, pp. 1288–1291, Aug. 2004.
- [43] H. K. Jung, N. S. Kim, and T. Kim, “A new double-talk detector using echo path estimation,” *Speech Commun.*, vol. 45, no. 1, pp. 41–48, Jan. 2005.

## Addendum

In the last paragraph of Section 3.4.1, the term “more convex shape of the prediction error cost function” might be somewhat misleading and requires some more explanation. With the term “more convex shape”, we aim to denote that the number of local minima in the cost function is smaller. Generally, the prediction error cost function is non-convex and has multiple local minima, yet the number of local minima has been found to decrease as the near-end signal model order  $n_A$  is decreased. As a result, the probability of finding the global minimum using the prediction error identification approach increases as  $n_A$  is decreased, and hence convergence speed is gained.





## Chapter 4

# Acoustic Echo Path Undermodeling

Identification of undermodeled room impulse  
responses

Geert Rombouts, Toon van Waterschoot, Kris Struyve,  
Piet Verhoeve, and Marc Moonen

Published in *Proc. 2005 Int. Workshop Acoustic Echo Noise  
Control (IWAENC '05)*, Eindhoven, The Netherlands, Sept. 2005,  
pp. 153–156.

### Contributions of second author

- co-analysis of undermodeling problem
- co-development of concept of tackling undermodeling using prediction error identification
- co-analysis of resulting bias and variance
- co-design of evaluation experiments
- co-interpretation of simulation results
- co-formulation of conclusion
- text co-editing and co-revision

## Abstract

In several scenarios it is desired to obtain an estimate of only the first part of the room impulse response, e.g., due to computing power restrictions. Room impulse response estimation is also often required in continuous double-talk situations. In this paper we show that the PEM-AFROW algorithm which has recently been proposed for acoustic feedback cancellation, can be used in these situations to provide a low-variance estimate with only a small bias.

## 4.1 Introduction

In this paper, we will focus on scenarios in which it is desired to provide an estimate of the first part (undermodeling) of a room impulse response (RIR), while continuous double-talk is present. An example is an acoustic echo canceller followed by a postprocessor which can remove the residual echo due to the last part of the RIR (typically less energy).

Standard adaptive filtering algorithms will provide a biased and large-variance estimate of such a truncated impulse response. In this paper we will show that the most important of both is the large variance, and that by using the PEM-AFROW algorithm [1],[2] which was derived in the context of acoustic feedback cancellation, bias and variance can be reduced. In Section 4.2 a problem statement is given, and in Section 4.3 the PEM-AFROW-based approach is introduced. Section 4.4 provides the simulation results.

## 4.2 Undermodeled Room Impulse Response

Fig. 4.1 shows a RIR identification scheme. We assume that the RIR  $\mathbf{f}(t) \in \mathbb{R}^{n_F+1}$  varies slowly compared to the statistics of the signals. We assume that the signals involved will be speech signals, and it is known that they can be modeled as time-varying AR processes (TVAR). On the other hand, especially in undermodeling scenarios, speech segments which are stationary for a longer period than the length of the modeled part of the impulse response may occur. Hence in the simulations, we use both stationary and time-variant AR models. The near-end signal  $v(t)$  and the far-end signal  $u(t)$  are assumed to be independent.

In order to describe the undermodeling case, we define

$$\mathbf{f}(t) = \begin{bmatrix} \mathbf{f}_1(t) \\ \mathbf{f}_2(t) \end{bmatrix} \quad (4.1)$$

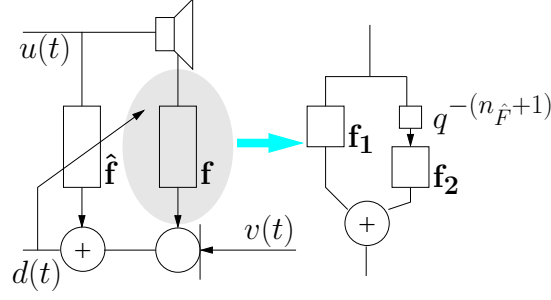


Figure 4.1: A traditional room impulse response identification scheme fails in the undermodeling case.

and we constrain

$$\hat{\mathbf{f}}(t) = \begin{bmatrix} \hat{\mathbf{f}}_1(t) \\ \mathbf{0} \end{bmatrix} \quad (4.2)$$

with  $\mathbf{f}_1(t), \hat{\mathbf{f}}_1(t) \in \mathbb{R}^{n_{\hat{F}}+1}$ . The MMSE criterion which is solved in a traditional echo canceller can then be specified as

$$\begin{aligned} \min E \left\{ \left[ \mathbf{u}^T(t) \begin{pmatrix} \mathbf{f}_1(t) \\ \mathbf{f}_2(t) \end{pmatrix} - \begin{bmatrix} \hat{\mathbf{f}}_1(t) \\ \mathbf{0} \end{bmatrix} + v(t) \right]^2 \right\} \\ = \min E \left\{ \left[ \mathbf{u}_1^T(t) \mathbf{f}_1(t) + \mathbf{u}_2^T(t) \mathbf{f}_2(t) - \mathbf{u}_1^T(t) \hat{\mathbf{f}}_1(t) + v(t) \right]^2 \right\} \end{aligned} \quad (4.3)$$

with  $\mathbf{u}_1(t)$  a vector containing the first  $n_{\hat{F}} + 1$  elements of  $\mathbf{u}(t)$ , and  $\mathbf{u}_2(t)$  the last  $n_F - n_{\hat{F}}$ . Define the Hankel matrices

$$\mathbf{U}_i(t) = [\mathbf{u}_i(t)^T \quad \dots \quad \mathbf{u}_i(1)^T]^T, \quad i = 1, 2 \quad (4.4)$$

The microphone signal consists of the signal  $\mathbf{U}_1(t)\mathbf{f}_1(t)$ , and what could be described as “noise”  $n(t)$  for the identification process. The noise consists of the signal  $\mathbf{U}_2(t)\mathbf{f}_2(t)$  which has a component  $n_b(t)$  in the column space of  $\mathbf{U}_1(t)$ , and a component *orthogonal* to the column space of  $\mathbf{U}_1(t)$ . The sum of the latter signal and the near-end signal  $v(t)$  will be called  $n_v(t)$ , and it will lead to variance on the estimate  $\hat{\mathbf{f}}_1(t)$  of  $\mathbf{f}_1(t)$ , while  $n_b(t)$  will lead to a bias on the estimate,

$$\mathbf{n}(t) = \underbrace{[\mathbf{U}_2(t)\mathbf{f}_2(t)]^{//}}_{\mathbf{n}_b(t): \text{Leads to bias}} + \underbrace{[\mathbf{U}_2(t)\mathbf{f}_2(t)]^{\perp} + \mathbf{v}(t)}_{\mathbf{n}_v(t): \text{Leads to variance}}. \quad (4.5)$$

The bias can be expressed by setting the derivative to  $\hat{\mathbf{f}}_1(t)$  of (4.3) to zero:

$$-2E\{\mathbf{u}_1(t)\mathbf{u}_1^T(t)\mathbf{f}_1(t) + \mathbf{u}_1(t)\mathbf{u}_2^T(t)\mathbf{f}_2(t) - \mathbf{u}_1(t)\mathbf{u}_1^T(t)\hat{\mathbf{f}}_1(t)\} + \underbrace{E\{\mathbf{u}_1(t)v(t)\}}_{=\mathbf{0}} = 0. \quad (4.6)$$

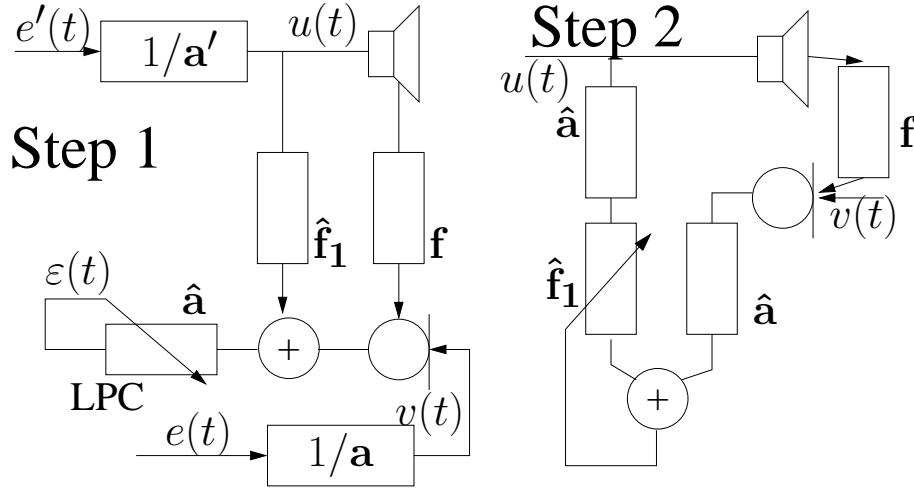


Figure 4.2: PEM-AFROW applied for identification of an undermodeled room impulse response.

Now define  $\mathbf{R}_{11} = E\{\mathbf{u}_1(t)\mathbf{u}_1^T(t)\}$  and  $\mathbf{R}_{12} = E\{\mathbf{u}_1(t)\mathbf{u}_2^T(t)\}$ . We then have for the bias

$$\hat{\mathbf{f}}_1(t) - \mathbf{f}_1(t) = \mathbf{R}_{11}^{-1}\mathbf{R}_{12}\mathbf{f}_2(t). \quad (4.7)$$

Note that the bias is zero when  $u(t)$  is white noise.

### 4.3 PEM-AFROW-Based Approach

In Fig. 4.2, the PEM-AFROW scheme applied to undermodeled RIR identification is shown. The PEM-AFROW scheme [1],[2] is a prediction error method [3], specifically applied to nonstationary TVAR signals under the assumption that the plant (the RIR) changes slower than the statistics of the signals. It can be used both in open loop [4],[5] and in closed loop [1],[2].

For the simulations which we will perform in this paper, we assume a near-end noise signal

$$v(t) = -a_1(t)v(t-1) - \dots - a_{n_A}(t)v(t-n_A) + e(t), \quad (4.8)$$

with  $e(t)$  a white noise sequence, and similarly, a far-end signal

$$u(t) = -a'_1(t)u(t-1) - \dots - a'_{n_A}(t)u(t-n_A) + e'(t). \quad (4.9)$$

Define

$$\mathbf{a}(t) = [1 \quad a_1(t) \quad \dots \quad a_{n_A}(t)]^T \quad (4.10)$$

and  $\mathbf{a}'(t)$  similarly. The coefficients  $\mathbf{a}(t)$  and  $\mathbf{a}'(t)$  will either be fixed, or changing every 20 ms.

The PEM-AFROW algorithm is based on an MSE cost function given as

$$\min_{\hat{\mathbf{a}}(t), \hat{\mathbf{f}}_1(t)} E \left\{ \left[ \hat{\mathbf{a}}^T(t) (\mathbf{U}_1(t) [\mathbf{f}_1(t) - \hat{\mathbf{f}}_1(t)] + \mathbf{U}_2(t) \mathbf{f}_2(t) + \mathbf{v}(t)) \right]^2 \right\}. \quad (4.11)$$

In this criterion,  $\hat{\mathbf{a}}(t)$  and  $\hat{\mathbf{f}}_1(t)$  are estimated in an alternating fashion, and on a frame-by-frame basis. In a first step,  $\hat{\mathbf{f}}_1(t)$  is assumed “correct” and kept fixed, and  $\hat{\mathbf{a}}(t)$  will be estimated by linear prediction such that the residual energy is minimized for a frame of data, in which the signals can be assumed to be stationary (20 ms for speech). This means that linear prediction is performed on *the combination*  $n(t)$  of the AR process  $v(t)$  and the ARMA process  $\mathbf{U}_2(t) \mathbf{f}_2(t)$ . In a second step, for the same frame of data, the far-end and the microphone signal are prefiltered by the linear prediction error filter  $\hat{\mathbf{a}}(t)$ , and from these prefiltered signals, the estimate  $\hat{\mathbf{f}}_1(t)$  is updated.

It should be noted that in case of a white far-end signal  $u(t)$ , in which a conventional echo canceller in an undermodeling setup would perform bias-free, inserting the prediction error filter  $\hat{\mathbf{a}}(t)$  of order  $n_A$ , would lead to a bias  $\mathbf{Q}_1^{-1} \mathbf{Q}_2 \mathbf{f}_2(t)$ , where  $\mathbf{Q}_1$  is a band-diagonal matrix with  $2n_A - 1$  non-zero diagonals, and  $\mathbf{Q}_2$  a matrix with only non-zero elements on a triangle in the first  $n_A$  columns and on the last  $n_A$  rows. If the AR models used are stable,  $\mathbf{Q}_1^{-1}$  can be approximated as a band-diagonal matrix too, and the bias will mainly occur in the last  $n_A$  elements of  $\hat{\mathbf{f}}_1(t)$ . Since  $n_A$  is usually small compared to  $n_{\hat{F}}$ , these  $n_A$  elements can be discarded. On the other hand, the structure of the scheme in Fig. 4.2 allows for  $\hat{\mathbf{a}}(t)$  to form an inverse model for the AR process generating the far-end signal, and hence it can also *reduce* the bias. In this setup,  $\hat{\mathbf{a}}(t)$  will minimize the linear prediction error energy of  $n(t)$ , and its effect will depend on the relative energies of the different components of  $n(t)$ .

The PEM-AFROW-based method will –as we will show in Section 4.4 below– effectively reduce the variance on the estimate  $\hat{\mathbf{f}}_1(t)$ , since the energy of the orthogonal part  $\mathbf{n}_v(t)$  of  $\mathbf{n}(t)$  is reduced after prefiltering in Step 2.

## 4.4 Simulations

In the simulations, we use an artificial RIR with 800 taps (as shown in Fig. 4.3), and in each of the experiments, only the first 250 taps of this impulse response will be modeled. From the figure it is clear that a significant amount of energy resides in tap 251 to 800, and (as will become clear in the simulations) without precautions, estimates will be useless. All experiments were performed with least squares identification (batch solution of a least squares system), and

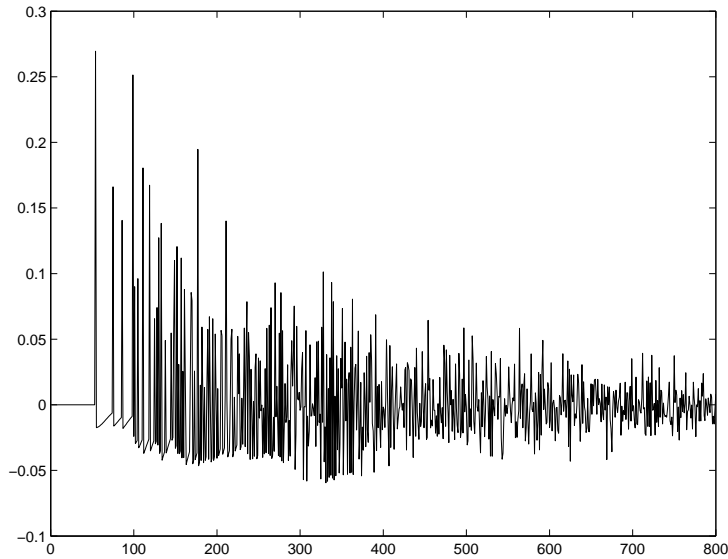


Figure 4.3: The room impulse response (800 taps) of which only 250 taps are modeled in the experiments.

*not* with stochastic gradient algorithms (NLMS). In NLMS-type algorithms the variance on the estimate would seem larger because of the excess mismatch which occurs in these algorithms due to the presence of near-end signals. This effectively means that the proposed technique even provides a larger improvement on NLMS-type algorithms than on the LS solutions in the simulations. For the simulations, 500 trials were run, and then the bias and variance of the estimate of the modeled part of the RIR was calculated. Instead of the variance, we plot the square root of the variance (the standard deviation), because this can directly be compared to the amplitude of the RIR.

We first consider a stationary AR far-end signal. While speech is of course nonstationary, this scenario is relevant in case of strong undermodeling (less than 20 ms of the impulse response). Fig. 4.4 shows a simulation where no near-end signal is present. In the upper figure the square root of the variance (standard deviation) and the bias on the estimate of the first 250 taps are shown when direct identification (DI) is applied by solving a least squares system, and in the lower figure the result is shown when PEM-AFROW is applied. Without PEM-AFROW the standard deviation is about 0.5, to be compared to the amplitude of the RIR, which peaks to only 0.3, see Fig. 4.3. The bias is concentrated in the last taps of the modeled part of the RIR, because due to the (stable) AR process for the far-end signal, the unmodeled part is less correlated with the new process samples than with the (older) process samples which correspond to the last taps of the modeled part. When PEM-AFROW

is applied (lower part of the figure), the standard deviation drops spectacularly to 0.01, and the bias is lowered, but still concentrated in the last taps. This is an interesting result, since these last taps can easily be discarded.

In Fig. 4.5, a stationary near-end signal is added. The energy of the stimulus of this signal,  $e(t)$  is chosen from a uniform random distribution between -5dB and 0 dB compared to the energy of the stimulus  $e'(t)$  of the far-end signal. Similar results are obtained: the bias is (both with and without application of PEM-AFROW) negligible compared to the variance, and concentrated around the last taps of the modeled part. The variance drops because the linear prediction error filter reduces the energy in the uncorrelated part of the microphone signal.

In Fig. 4.6, the simulation is repeated for a time-variant AR (TVAR) signal for the far-end. The AR coefficients are chosen randomly with a pole radius of 0.83, and kept stationary for about 20 ms (as in speech). In this experiment, no near-end signal is added, and only the undermodeling performance is evaluated.

In Fig. 4.7, the approach is validated on a real speech signal, with a real-time implementation using NLMS adaptive filters. The impulse response was measured independently in silence in order to have a reference. The first 1000 taps of the 5000 taps impulse response are modeled and shown, together with their estimate performed by an NLMS adaptive filter (upper plot) and by PEM-AFROW with NLMS as its adaptive filter (lower plot; contrarily to the experiments above, the plots only show one realization). The estimate by the PEM-AFROW with NLMS algorithm is observed to be much better than the estimate provided by NLMS (direct identification). This is most easily observed in the first 500 taps where the impulse response is zero.

## 4.5 Conclusion

We have experimentally shown that the PEM-AFROW algorithm can be used to provide estimates of an undermodeled room impulse response with both a low variance and a bias which is concentrated in a few filter taps, which can easily be discarded.

## Bibliography

- [1] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, "Acoustic feedback suppression for long acoustic paths using a nonstationary source model," in *Proc. 13th European Signal Process. Conf. (EUSIPCO '05)*, Antalya, Turkey, Sep. 2005.



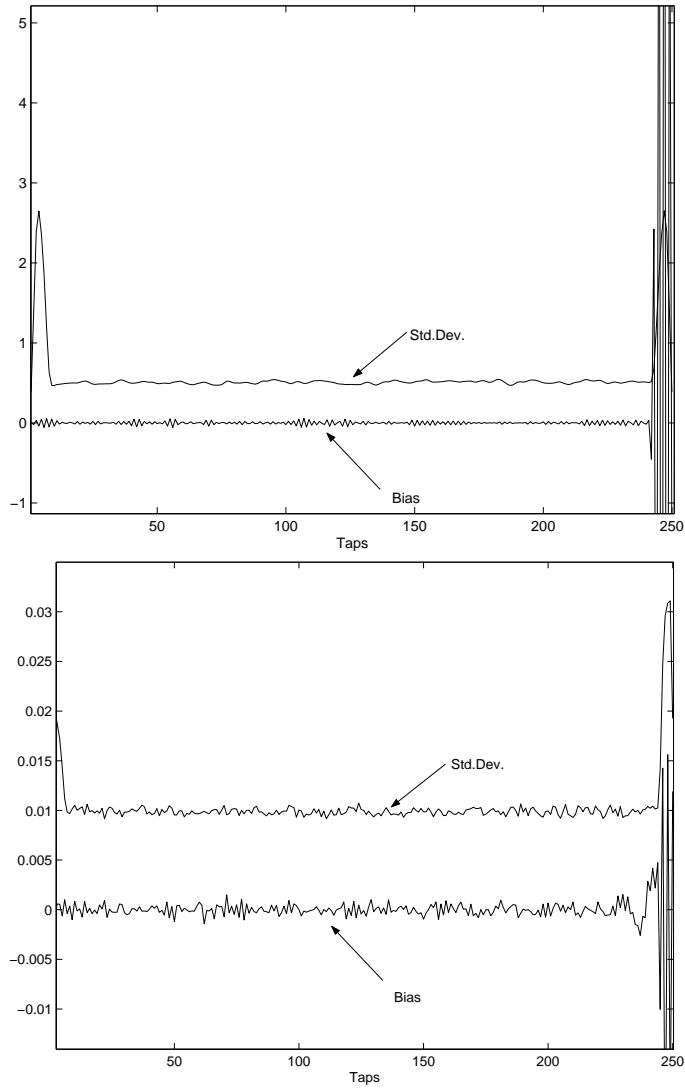


Figure 4.4: Stationary AR far-end, no near-end. Upper: direct identification (DI), lower: PEM-AFROW.

- [2] —, “Acoustic feedback suppression for long acoustic paths using a non-stationary source model,” *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.
- [3] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
- [4] T. van Waterschoot, G. Rombouts, and M. Moonen, “On the performance of decorrelation by prefiltering for adaptive feedback cancellation in public address systems,” in *Proc. 4th IEEE Benelux Signal Process. Symp. (SPS '04)*, Hilvarenbeek, The Netherlands, Apr. 2004, pp. 167–170. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/04-24.html>
- [5] T. van Waterschoot, G. Rombouts, K. Struyve, and M. Moonen, “Acoustic echo cancellation in the presence of continuous double-talk,” in *Proc. 1st Annual IEEE Benelux/DSP Valley Signal Process. Symp. (SPS-DARTS '05)*, Antwerp, Belgium, Apr. 2005, pp. 53–57. [Online]. Available: <ftp://ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/04-149.html>

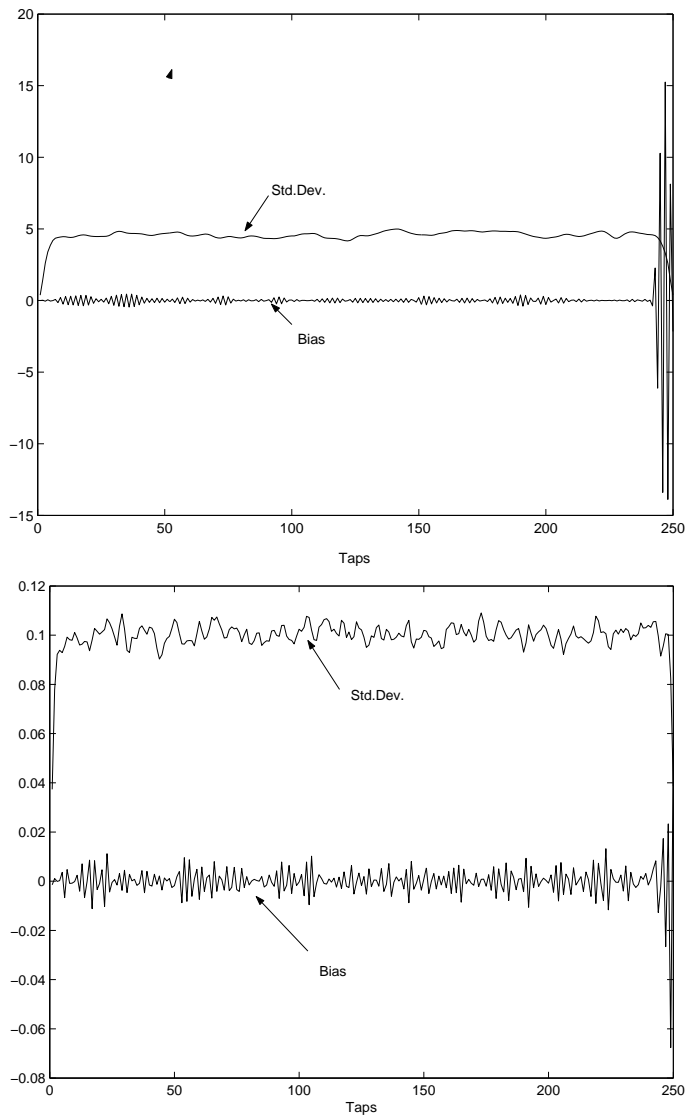


Figure 4.5: Stationary AR far-end and near-end. Upper: DI, lower: PEM-AFROW.

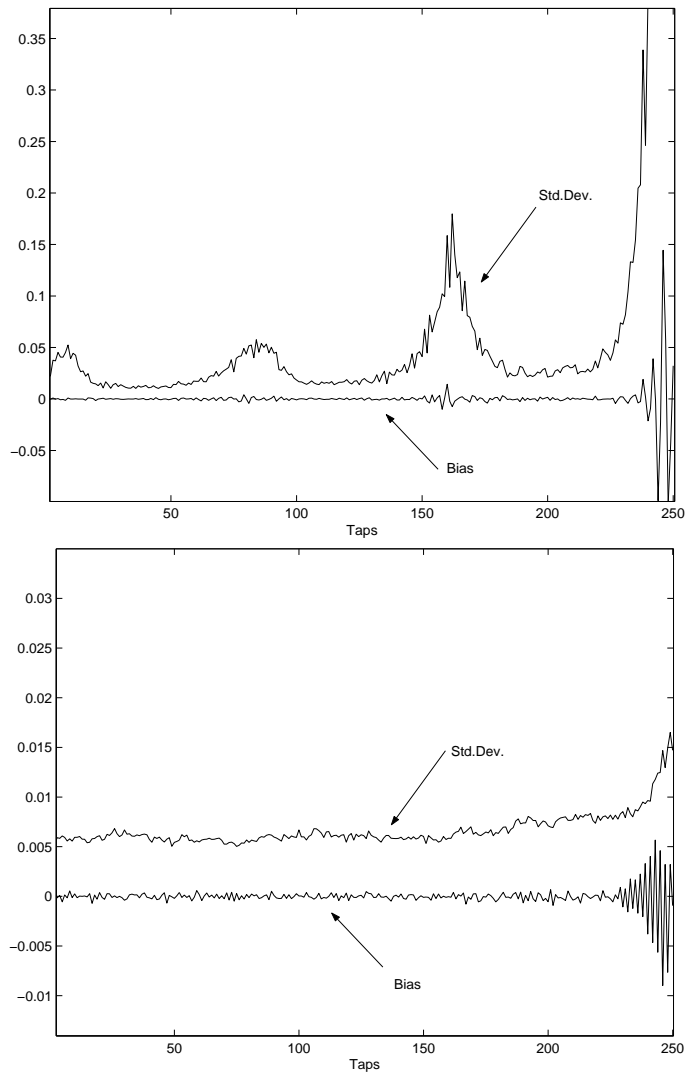


Figure 4.6: TVAR far-end, no near-end. Upper: DI, lower: PEM-AFROW.

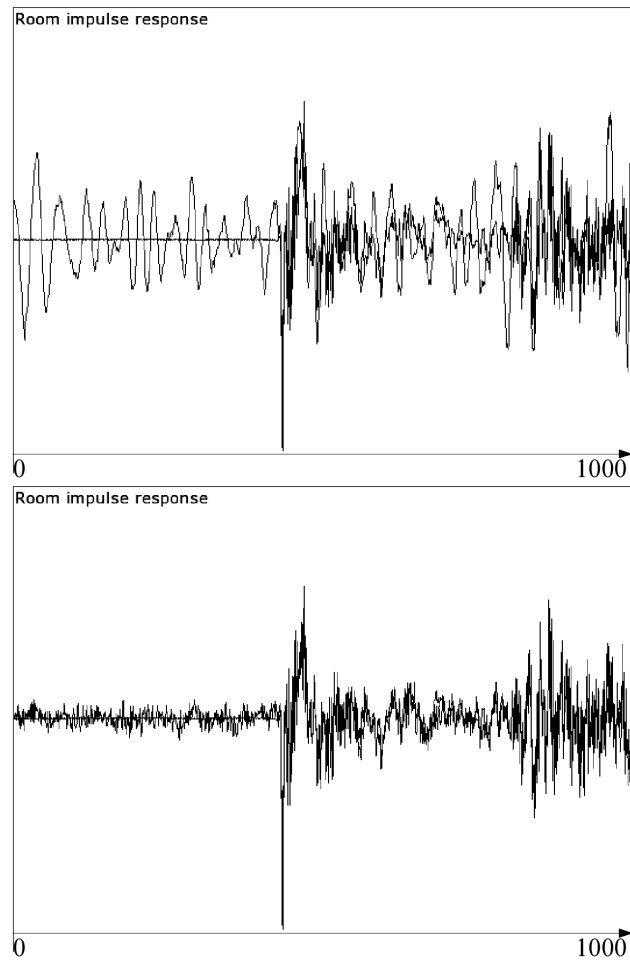


Figure 4.7: Upper: undermodeled (1000 out of 5000 taps) identification with NLMS. Lower: with PEM-AFROW.



## Chapter 5

# Optimal Regularization for AEC/AFC

Optimally regularized adaptive filtering  
algorithms for room acoustic signal enhancement

Toon van Waterschoot, Geert Rombouts, and Marc Moonen

Published in *Signal Processing*, vol. 88, no. 3, Mar. 2008,  
pp. 594–611.

### Contributions of first author

- literature study
- co-development of optimal regularization concept and unifying regularization framework
- co-development of formal relation between regularization and Bayesian MMSE estimation
- co-development of prior knowledge gathering techniques
- co-development of optimally regularized adaptive filtering algorithms
- co-development of relation between optimal regularization and proportionate adaptation
- co-design of evaluation experiments
- software implementation and computer simulations
- co-interpretation of simulation results
- co-formulation of conclusion
- text redaction and editing



## Abstract

In many room acoustic signal processing applications, a room impulse response identification is needed to eliminate undesired effects such as echo, feedback, or reverberation. This is typically done using an adaptive filter driven by a speech or audio input signal. However, such signals exhibit poor excitation properties, which cause standard adaptive filtering algorithms to be very sensitive to disturbing signals, especially in the underdetermined case. A popular remedy is regularization, which is usually implemented with a scaled identity regularization matrix. This type of regularization is governed by a single regularization parameter, the value of which is often chosen in an arbitrary way. We propose to regularize the adaptive filter using a non-identity regularization matrix, in which prior knowledge on the unknown room impulse response may be incorporated. When knowledge of the disturbing signal is also used to add prefiltering and weighting in the adaptation, a new family of regularized adaptive filtering algorithms is obtained, which is shown to be optimal in a mean square error sense. Existing regularized algorithms can then be obtained as special cases, assuming limited or no prior knowledge is available. When combined with a recently proposed method of extracting prior knowledge from the acoustic setup, our algorithms exhibit superior convergence behavior compared to existing algorithms in different simulation scenarios, while the additional computational cost is small.

## 5.1 Introduction

Several room acoustic signal processing applications require the identification of a room impulse response (RIR). The most well-known application is probably acoustic echo cancellation (AEC) [1], in which a RIR estimate is needed for the prediction of an echo component that is present in a microphone signal. Other applications include adaptive feedback cancellation [2], which is similar to AEC but involves a closed signal loop, and active noise control [3], where both the so-called primary and secondary signal path have to be identified.

We will focus on the AEC scenario for illustrating the general RIR identification problem, see Fig. 5.1. We assume that the true RIR coefficients of the echo path between the loudspeaker and the microphone are collected in the parameter vector

$$\mathbf{f} \triangleq [f_0 \quad f_1 \quad \dots \quad f_{n_F}]^T, \quad (5.1)$$

of known length  $n_F + 1$ . The loudspeaker produces a signal  $u(t)$  (often referred to as the “far-end” signal), which generates an echo signal  $x(t) = \mathbf{u}^T(t)\mathbf{f}$  at the microphone position, with the loudspeaker signal vector defined as

$$\mathbf{u}(t) \triangleq [u(t) \quad u(t-1) \quad \dots \quad u(t-n_F)]^T. \quad (5.2)$$

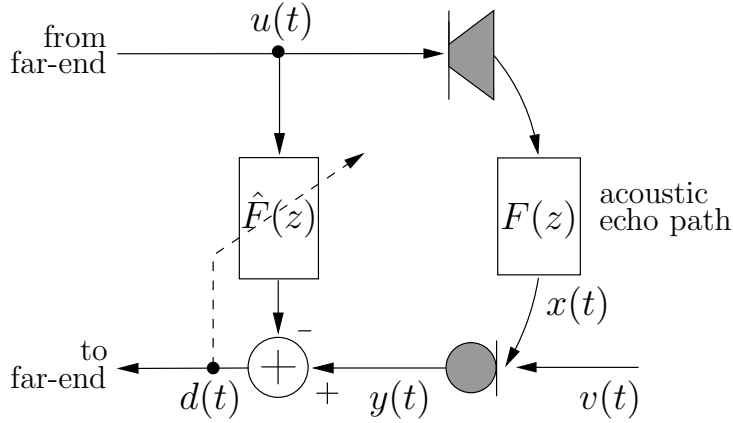


Figure 5.1: Room impulse response identification in an AEC scenario.  $F(z)$  and  $\hat{F}(z)$  represent the transfer functions associated with the filter coefficients in  $\mathbf{f}$  and  $\hat{\mathbf{f}}(t)$ , respectively.

The echo signal  $x(t)$  is picked up by the microphone in addition to a local signal  $v(t)$ , which is defined as the source signal (often referred to as the “near-end” signal), hence the microphone signal can be written as  $y(t) = x(t) + v(t)$ . It is important to note that the source signal is in most applications the signal of interest, however, from a system identification point of view,  $v(t)$  is a disturbance to the estimation of the RIR. If at time  $t$  a RIR estimate  $\hat{\mathbf{f}}(t)$  is available, then an echo-compensated signal can be calculated as  $d(t) = y(t) - \mathbf{u}^T(t)\hat{\mathbf{f}}(t)$ , which approximates the source signal  $v(t)$ . Note that, throughout this paper, we assume that the true RIR order  $n_F$  is known, and equal to the adaptive filter order. Methods for dealing with the insufficient-order case can be found in, e.g., [4],[5].

Nearly all linear adaptive filtering algorithms are related to the well-known least squares (LS) estimate of the RIR, given by

$$\hat{\mathbf{f}}(t) = (\mathbf{U}^T \mathbf{U})^{-1} \mathbf{U}^T \mathbf{y}, \quad (5.3)$$

with the data model for the discrete time range  $[1, t]$  defined as

$$\mathbf{y} = \mathbf{U}\mathbf{f} + \mathbf{v}, \quad (5.4)$$

and

$$\mathbf{U} \triangleq [\mathbf{u}(t) \quad \mathbf{u}(t-1) \quad \dots \quad \mathbf{u}(1)]^T, \quad (5.5)$$

$$\mathbf{y} \triangleq [y(t) \quad y(t-1) \quad \dots \quad y(1)]^T, \quad (5.6)$$

$$\mathbf{v} \triangleq [v(t) \quad v(t-1) \quad \dots \quad v(1)]^T. \quad (5.7)$$

A common problem in room acoustic applications is that the matrix  $\mathbf{U}^T\mathbf{U}$ , which is inverted in the LS estimate (5.3), is ill-conditioned or even singular due to poor excitation. Indeed, the identifiability of the RIR  $\mathbf{f}$  is only guaranteed if the loudspeaker signal  $u(t)$  is persistently exciting of order  $n_F$  [6]. However, the dynamics of a typical RIR can often only be captured with several thousands of coefficients. On the other hand, the loudspeaker signal is usually a speech or audio signal, and may exhibit a nearly harmonic spectrum, such that its excitation order is far below  $n_F$ . If such an ill-conditioned situation occurs, there will typically be a large variance on the resulting RIR estimate when the source signal  $v(t)$  is non-zero, as can be seen from the LS estimate covariance matrix:

$$\text{cov}\{\hat{\mathbf{f}}(t)\} = (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{R}_v\mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}, \quad (5.8)$$

with the (zero-mean) source signal covariance matrix defined as

$$\mathbf{R}_v = E\{\mathbf{v}\mathbf{v}^T\}, \quad (5.9)$$

and with  $E\{\cdot\}$  and  $\text{cov}\{\cdot\}$  the expectation and covariance operators, respectively.

Regularization is a standard technique to turn such an ill-posed problem into a well-posed one, namely by improving the condition of the matrix  $\mathbf{U}^T\mathbf{U}$  in the LS estimate (5.3). The regularization method that is most often adopted in LS estimation is the one proposed by Tikhonov et al. [7] (and reformulated in a linear algebra setting by Neumaier [8]). The Tikhonov regularization method applied to the LS estimate, consists in adding a scaled identity matrix to the matrix  $\mathbf{U}^T\mathbf{U}$ , i.e.,

$$\hat{\mathbf{f}}(t) = (\mathbf{U}^T\mathbf{U} + \alpha\mathbf{I})^{-1}\mathbf{U}^T\mathbf{y}, \quad (5.10)$$

where  $\alpha$  denotes the regularization parameter. This is in fact the minimizing estimate of a modified LS criterion in which the squared Euclidian norm of the RIR estimate is added to the sum of squared errors, and weighted with the regularization parameter  $\alpha$ :

$$\min_{\hat{\mathbf{f}}(t)} \left\{ [\mathbf{y} - \mathbf{U}\hat{\mathbf{f}}(t)]^T [\mathbf{y} - \mathbf{U}\hat{\mathbf{f}}(t)] + \alpha \|\hat{\mathbf{f}}(t)\|_2^2 \right\}. \quad (5.11)$$

From this criterion, it can be seen that a large RIR estimate is penalized by the regularization term, leading to an estimate that is biased towards zero. However, allowing this bias in particular results in a reduction of the estimation variance.

The Tikhonov regularized LS estimate in (5.10) may be calculated recursively by initializing the adaptive filter input correlation matrix as  $\mathbf{R}(0) = \alpha\mathbf{I}$ , and applying the standard recursive least squares (RLS) algorithm:

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mathbf{R}^{-1}(t)\mathbf{u}(t)\varepsilon(t), \quad (5.12)$$

$$\mathbf{R}(t) = \mathbf{R}(t-1) + \mathbf{u}(t)\mathbf{u}^T(t), \quad (5.13)$$

$$\varepsilon(t) = y(t) - \mathbf{u}^T(t)\hat{\mathbf{f}}(t-1). \quad (5.14)$$

Note that the a priori residual  $\varepsilon(t)$  in (5.14) differs from the a posteriori residual  $d(t)$ , which is sent to the far-end side, in that it depends on the previous RIR estimate  $\hat{\mathbf{f}}(t-1)$ . If, for tracking purposes, an exponential window is applied in the above RLS algorithm, then the regularization matrix  $\alpha\mathbf{I}$  will be exponentially weighted, along with older data (which results in a technique known as soft-constrained initialization [9]). If we wish to maintain the regularization throughout the complete time range  $[1, t]$ , an additional term should be added to the correlation matrix update in order to compensate for the weighting of the regularization matrix. Moreover, with this modification, the RLS algorithm would not converge to the (exponentially weighted) LS estimate, unless a correction term is also subtracted from the RIR weight update. The *Tikhonov regularized RLS algorithm* with exponential weighting factor  $\lambda$  is then given by:

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mathbf{R}^{-1}(t)[\mathbf{u}(t)\varepsilon(t) - (1-\lambda)\alpha\hat{\mathbf{f}}(t-1)], \quad (5.15)$$

$$\mathbf{R}(t) = \lambda\mathbf{R}(t-1) + \mathbf{u}(t)\mathbf{u}^T(t) + (1-\lambda)\alpha\mathbf{I}, \quad (5.16)$$

$$\varepsilon(t) = y(t) - \mathbf{u}^T(t)\hat{\mathbf{f}}(t-1). \quad (5.17)$$

This algorithm was proposed by Horita et al. [10], and was called the Leaky RLS algorithm because of its similarity with the Leaky Least Mean Squares (LMS) algorithm [11] if the RIR weight update in (5.15) is rewritten as

$$\hat{\mathbf{f}}(t) = [\mathbf{I} - (1-\lambda)\alpha\mathbf{R}^{-1}(t)]\hat{\mathbf{f}}(t-1) + \mathbf{R}^{-1}(t)\mathbf{u}(t)\varepsilon(t). \quad (5.18)$$

A different regularization technique, which is more often applied to RLS adaptive filtering algorithms, is known as Levenberg-Marquardt regularization [12]. This method is very similar to Tikhonov regularization, however no correction term is subtracted from the RIR weight update. The *Levenberg-Marquardt regularized RLS algorithm* is given by:

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mathbf{R}^{-1}(t)\mathbf{u}(t)\varepsilon(t), \quad (5.19)$$

$$\mathbf{R}(t) = \lambda\mathbf{R}(t-1) + \mathbf{u}(t)\mathbf{u}^T(t) + (1-\lambda)\alpha\mathbf{I}, \quad (5.20)$$

$$\varepsilon(t) = y(t) - \mathbf{u}^T(t)\hat{\mathbf{f}}(t-1). \quad (5.21)$$

The Levenberg-Marquardt method was initially developed by Levenberg [13] and Marquardt [14] as a hybrid technique for nonlinear optimization. To achieve an optimal trade-off between convergence speed and robustness, the Gauss-Newton method [with update term  $\mathbf{u}(t)\mathbf{u}^T(t)$ ] was combined with the steepest descent method [with update term  $(1-\lambda)\mathbf{I}$ ] in the correlation matrix update (5.20), by means of a steering factor  $\alpha$ . The interpretation of the Levenberg-Marquardt method as a regularization technique was first given by Ljung and Söderström [12], and found its way into acoustic signal processing through the work of Gay [15] and Benesty et al. [1], among others. However,

as already mentioned, due to the absence of the correction term in the RIR weight update [compare (5.19) to (5.15)], the Levenberg-Marquardt regularized RLS algorithm does not converge to the Tikhonov regularized LS estimate in (5.10).

In room acoustic applications, the underdetermined recursive least squares (URLS) family [16], of which the normalized least mean squares (NLMS) algorithm and the affine projection algorithm (APA) are the most well-known members, is much more appealing from a computational point of view. However, due to their underdetermined nature, these algorithms are even more susceptible to convergence problems resulting from poor excitation. Therefore, regularization is also included in nearly every algorithm from the URLS family. The *regularized APA* is similar to the Levenberg-Marquardt regularized RLS algorithm, in that a scaled identity matrix is added to the adaptive filter input correlation matrix before inversion:

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mu \mathbf{U}_M(t) [\mathbf{U}_M^T(t) \mathbf{U}_M(t) + \alpha \mathbf{I}]^{-1} \boldsymbol{\varepsilon}_M(t), \quad (5.22)$$

$$\boldsymbol{\varepsilon}_M(t) = \mathbf{y}_M(t) - \mathbf{U}_M^T(t) \hat{\mathbf{f}}(t-1), \quad (5.23)$$

where now the identity matrix is of dimension  $M \times M$ , with  $M$  the projection order,  $\mu$  represents the step size, and

$$\mathbf{U}_M(t) \triangleq [\mathbf{u}(t) \quad \dots \quad \mathbf{u}(t-M+1)], \quad (5.24)$$

$$\mathbf{y}_M(t) \triangleq [y(t) \quad \dots \quad y(t-M+1)]^T. \quad (5.25)$$

The well-known *NLMS algorithm* can then be obtained from the APA by setting  $M = 1$ :

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mu \frac{\mathbf{u}(t) \boldsymbol{\varepsilon}(t)}{\mathbf{u}^T(t) \mathbf{u}(t) + \alpha}, \quad (5.26)$$

$$\boldsymbol{\varepsilon}(t) = y(t) - \mathbf{u}^T(t) \hat{\mathbf{f}}(t-1). \quad (5.27)$$

The aim of this paper is twofold. First of all, we will show how the different regularized algorithms fit into a common framework, and point out the relationship between regularization and some other concepts in adaptive filtering such as leakage and proportionate adaptation. Second, we raise the question whether these existing regularization techniques are optimal in some sense, and to which extent they may be improved by using more than just one regularization parameter. Many choices for the regularization parameter  $\alpha$  in the above algorithms have been proposed, ranging from any small number [1],[16]-[23], to a parameter incorporating prior knowledge on the near-end signal [8],[15],[24]-[26], or on both the near-end signal and the unknown RIR [27],[28]. In many acoustic signal enhancement applications, the regularization parameter is an a priori estimate of the long-term source signal power, as suggested in [15]. Also, some algorithms have been proposed in which the regularization parameter can

be estimated online [29]-[31]. The use of multiple regularization parameters has previously been proposed for frequency domain and subband adaptive filtering, by assigning a different regularization parameter to each frequency band, see, e.g., [32]. Also, in neural networks [33, Ch. 5], it is customary to increase the number of regularization parameters by adding a more general penalty term  $\alpha\|\mathbf{L}\hat{\mathbf{f}}(t)\|_2^2$  instead of  $\alpha\|\hat{\mathbf{f}}(t)\|_2^2$  in the Tikhonov regularized LS criterion (5.11), as suggested in [34],[35], with  $\mathbf{L}$  typically chosen as a discrete approximation to some derivative operator for imposing smoothness on the RIR estimate.

In this paper, the main motivation for using more than one regularization parameter is related to the availability of prior knowledge. There exists a well-known equivalence between Tikhonov regularization and Bayesian maximum a posteriori (MAP) or minimum mean square error (MMSE) estimation. In a Bayesian framework, prior knowledge on the unknown RIR parameters can be incorporated in a non-identity regularization matrix. This approach is most often pointed to as ‘‘Bayesian regularization’’ (e.g., in the neural networks literature [36],[37]), or also as ‘‘Generalized Tikhonov regularization’’ (in the inverse problems literature [38]). In this paper, these concepts are extrapolated to linear adaptive filter theory, and applied to room acoustic problems. Recently, the RIR estimation problem was also tackled using Bayesian regularization in [39], where an iterative optimization algorithm was applied using both nonnegativity constraints and  $L1$ -norm sparsity regularization. As a consequence, this method relies on nonnegativity and sparseness of the RIR coefficients, which may be valid assumptions for synthetic RIRs generated using the image source method [40] (as used in the simulations in [39]), but not for realistic RIRs.

In Section 5.2, we review some results from Bayesian linear minimum mean square error estimation [41], which are essential for deriving the proposed adaptive filtering algorithms. We will formulate a non-recursive RIR estimate that is optimal in the sense of minimizing the mean square deviation of the RIR estimate from the true RIR in a Bayesian framework. This linear minimum mean square error (LMMSE) RIR estimate depends on the statistical assumptions on the source signal  $v(t)$ , as well as on the true RIR  $\mathbf{f}$ , and can also be obtained as the minimizing estimate of a weighted and regularized LS criterion. This observation has then motivated us to derive from the latter criterion a set of regularized adaptive filtering algorithms, which contains the existing regularized algorithms as special cases.

The MSE optimally regularized algorithms rely on knowledge of the source signal and true RIR statistics, which are unknown in most applications. In a previous paper, we have proposed adaptive filtering algorithms based on the prediction error method [12], with which the source signal statistics can be estimated concurrently with the unknown RIR (both for acoustic echo cancellation [42],[43] and for adaptive feedback cancellation [2],[44]). Also, we have proposed a method for constructing the true RIR statistics from prior knowledge on the acoustic setup [45]. In Section 5.3, these techniques are briefly

reviewed, and we comment on how they can provide the necessary statistical information for the algorithms proposed in Sections 5.4 and 5.5.

In Section 5.4, we focus on MSE optimal regularization of recursive least squares (RLS) algorithms. We will propose both a Tikhonov regularized (TR-RLS) algorithm and a Levenberg-Marquardt regularized (LMR-RLS) algorithm, with a regularization matrix which may contain prior knowledge on the true RIR. It is shown that the TR-RLS and LMR-RLS algorithms may in fact be obtained by minimization of the same criterion, and only differ in the statistical assumptions on the true RIR. Furthermore, we point out the increase in computational complexity that is inherent in regularizing the RLS algorithm, as because of the regularization, the matrix inversion lemma cannot be applied anymore to the adaptive filter input correlation matrix update [which can also be seen from the update equations (5.16) and (5.20)]. We discuss two methods for avoiding the explicit inversion of the RLS input correlation matrix, and hence reducing the complexity from  $O(n_F^3)$  to  $O(n_F^2)$  computations per iteration, in particular by: 1) approximating the regularization matrix by a sum of rank-one matrices [12] (also known as dynamic regularization [18]), and 2) applying a rectangular instead of an exponential data window [46]. Another technique for obtaining  $O(n_F^2)$  complexity, namely by indirect updating of the input correlation matrix through its eigenvalue decomposition [10],[29], is only feasible if the regularization matrix is a scaled identity matrix.

Regularization of underdetermined recursive least squares algorithms is tackled in Section 5.5, where in particular we discuss the regularized NLMS and APA algorithms. Minimization of the weighted and regularized LS criterion that led to the TR-RLS and LMR-RLS algorithms, yields in the underdetermined case an algorithm that exhibits features known as leakage and proportionate adaptation. However, from extensive simulations, we have learned that leakage (even in the simple case of the leaky LMS algorithm [11]) may significantly deteriorate the adaptive filter performance in room acoustic applications. Adopting the same statistical assumptions on the true RIR that led to the LMR-RLS algorithm, now leads to a set of regularized NLMS and APA algorithms, without leakage. Moreover, the proposed LMR-NLMS and LMR-APA algorithms are formally equivalent to the proportionate NLMS (PNLMS) [23],[47] and proportionate APA (PAPA) [21] algorithms. This equivalence leads to a new interpretation of the proportionate adaptation concept, and hence to an MSE optimal choice of the proportionate adaptation parameters.

The proposed regularized adaptive filtering algorithms have been tested by means of computer simulations of different scenarios and applications. Simulation results are discussed in Section 5.6. Finally, Section 5.7 concludes the paper.

## 5.2 MSE Optimal Weighting and Regularization

As discussed in Section 5.1, most regularized linear adaptive filtering algorithms described earlier, are based on the Tikhonov regularized LS estimate in (5.10) and are tuned by a single regularization parameter  $\alpha$ . A more general approach towards regularization is to replace the scaled Euclidian norm  $\alpha\|\hat{\mathbf{f}}(t)\|_2^2$  in the Tikhonov regularized LS criterion in (5.11) by the weighted Euclidian norm of the deviation  $[\hat{\mathbf{f}}(t) - \boldsymbol{\xi}]$  of the RIR estimate  $\hat{\mathbf{f}}(t)$  from some reference value  $\boldsymbol{\xi}$ , with an  $(n_F + 1) \times (n_F + 1)$  weighting matrix  $\boldsymbol{\Phi}$ . We will also include a  $t \times t$  weighting matrix  $\mathbf{W}$  in the LS term, which will be useful further on:

$$\min_{\hat{\mathbf{f}}(t)} \left\{ [\mathbf{y} - \mathbf{U}\hat{\mathbf{f}}(t)]^T \mathbf{W} [\mathbf{y} - \mathbf{U}\hat{\mathbf{f}}(t)] + [\hat{\mathbf{f}}(t) - \boldsymbol{\xi}]^T \boldsymbol{\Phi} [\hat{\mathbf{f}}(t) - \boldsymbol{\xi}] \right\}. \quad (5.28)$$

Minimizing the above criterion leads to a weighted LS estimate that features a general regularization matrix  $\boldsymbol{\Phi}$ , not necessarily equal to a scaled identity matrix:

$$\hat{\mathbf{f}}(t) = \boldsymbol{\xi} + (\mathbf{U}^T \mathbf{W} \mathbf{U} + \boldsymbol{\Phi})^{-1} \mathbf{U}^T \mathbf{W} (\mathbf{y} - \mathbf{U}\boldsymbol{\xi}). \quad (5.29)$$

The properties of the above estimate will depend on the choice of the weighting matrices  $\mathbf{W}$  and  $\boldsymbol{\Phi}$ , and of the reference value  $\boldsymbol{\xi}$ . Note that by choosing  $\mathbf{W} = \sigma \mathbf{I}$  and  $\boldsymbol{\Phi} = \nu \mathbf{I}$  such that  $\sigma^{-1}\nu = \alpha$ , and  $\boldsymbol{\xi} = \mathbf{0}$ , we obtain the traditional Tikhonov regularized LS estimate, given in (5.10). A desirable property of a linear estimate is that it has minimum variance [41]. However, for biased estimates, such as the estimate in (5.29), both the bias and the variance should be minimized. A straightforward choice is to minimize the mean square error (MSE) between the estimated and true RIR, i.e., with  $\text{tr}\{\cdot\}$  the trace operator,

$$\min_{\hat{\mathbf{f}}(t)} E \left\{ [\hat{\mathbf{f}}(t) - \mathbf{f}]^T [\hat{\mathbf{f}}(t) - \mathbf{f}] \right\} = \min_{\hat{\mathbf{f}}(t)} E \left\{ \text{tr} \left\{ [\hat{\mathbf{f}}(t) - \mathbf{f}] [\hat{\mathbf{f}}(t) - \mathbf{f}]^T \right\} \right\}, \quad (5.30)$$

since then the variance and the squared bias are weighted equally [41]. Minimization of the above criterion in a deterministic framework leads to an estimate that depends on the unknown RIR  $\mathbf{f}$ . It is more useful to minimize (5.30) in a Bayesian framework, where not only the source signal vector  $\mathbf{v}$ , but also the true RIR  $\mathbf{f}$  is viewed as one particular realization of a stochastic vector process. If we assume a Gaussian prior for both vector processes, with first and second order moments given by

$$\begin{cases} E\{\mathbf{v}\} = \mathbf{0}, \\ \text{cov}\{\mathbf{v}\} = E\{\mathbf{v}\mathbf{v}^T\} = \mathbf{R}_{\mathbf{v}}, \end{cases} \quad (5.31)$$

$$\text{cov}\{\mathbf{v}\} = E\{\mathbf{v}\mathbf{v}^T\} = \mathbf{R}_{\mathbf{v}}, \quad (5.32)$$

and

$$\begin{cases} E\{\mathbf{f}\} = \mathbf{f}_0, \\ \text{cov}\{\mathbf{f}\} = E\{(\mathbf{f} - \mathbf{f}_0)(\mathbf{f} - \mathbf{f}_0)^T\} = \mathbf{R}_{\mathbf{f}}, \end{cases} \quad (5.33)$$

$$\text{cov}\{\mathbf{f}\} = E\{(\mathbf{f} - \mathbf{f}_0)(\mathbf{f} - \mathbf{f}_0)^T\} = \mathbf{R}_{\mathbf{f}}, \quad (5.34)$$



then the minimizing estimate of the MSE criterion in (5.30) is given by (see, e.g., [41], [48, Ch. 4])

$$\hat{\mathbf{f}}(t) = \mathbf{f}_0 + (\mathbf{U}^T \mathbf{R}_v^{-1} \mathbf{U} + \mathbf{R}_f^{-1})^{-1} \mathbf{U}^T \mathbf{R}_v^{-1} (\mathbf{y} - \mathbf{U} \mathbf{f}_0). \quad (5.35)$$

Comparing this estimate with the weighted and regularized LS estimate in (5.29), we conclude that the MSE optimal choice for the weighting matrices  $\mathbf{W}$  and  $\Phi$ , and for the reference value  $\xi$ , is

$$\mathbf{W} = \mathbf{R}_v^{-1}, \quad (5.36)$$

$$\Phi = \mathbf{R}_f^{-1}, \quad (5.37)$$

$$\xi = \mathbf{f}_0. \quad (5.38)$$

Hence the criterion for deriving MSE optimally weighted and regularized LS-based adaptive filtering algorithms can be formulated as (see, e.g., [49, Ch. 5])

$$\min_{\hat{\mathbf{f}}(t)} \left\{ [\mathbf{y} - \mathbf{U} \hat{\mathbf{f}}(t)]^T \mathbf{R}_v^{-1} [\mathbf{y} - \mathbf{U} \hat{\mathbf{f}}(t)] + [\hat{\mathbf{f}}(t) - \mathbf{f}_0]^T \mathbf{R}_f^{-1} [\hat{\mathbf{f}}(t) - \mathbf{f}_0] \right\}. \quad (5.39)$$

If no prior knowledge on the true RIR is available (i.e.,  $\|\mathbf{R}_f\|_F \rightarrow \infty$ , with  $\|\cdot\|_F$  the Frobenius norm), then effectively no regularization is applied. Also, when the source signal power is small, then the data will be more reliable and the impact of the regularization term will decrease. On the other hand, if the source signal power increases, the signal-to-noise ratio (SNR) decreases, and then regularization starts playing a more important role.

## 5.3 Source Signal and True RIR Statistics

The approach of Section 5.2 is only useful when reliable information on the source signal and on the true RIR is available through the mean  $\mathbf{f}_0$  and covariance matrices  $\mathbf{R}_v$  and  $\mathbf{R}_f$ . In Section 5.4, it will become clear that two particular choices for  $\mathbf{f}_0$  are of special interest: choosing  $\mathbf{f}_0 = \mathbf{0}$  leads to a Tikhonov type of regularization, whereas choosing  $\mathbf{f}_0 = \hat{\mathbf{f}}(t-1)$  yields a Levenberg-Marquardt type of regularization. The choice of the covariance matrices  $\mathbf{R}_v$  and  $\mathbf{R}_f$  is discussed below.

### 5.3.1 Source Signal Covariance Matrix $\mathbf{R}_v$

In room acoustic applications, the source signal will typically be a speech or audio signal. This is then a nonstationary signal, i.e., having time-varying characteristics, and hence it may be desirable to choose  $\mathbf{R}_v$  realization-dependent.

To this end, we factorize the inverse covariance matrix  $\mathbf{R}_v^{-1}$  as follows [6, App. II]:

$$\mathbf{R}_v^{-1} = \mathbf{A}^T \mathbf{\Sigma}^{-1} \mathbf{A}, \quad (5.40)$$

with  $\mathbf{A}$  a unit upper triangular matrix, i.e.,

$$\mathbf{A} \triangleq \begin{bmatrix} 1 & a_{12} & a_{13} & \dots & a_{1t} \\ 0 & 1 & a_{23} & \dots & a_{2t} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}, \quad (5.41)$$

and  $\mathbf{\Sigma}$  a diagonal matrix. It is now convenient, both from a physical point of view, as well as for computational reasons, to model the source signal  $v(t)$  as an autoregressive (AR) process of order  $n_A$ , with time-varying AR coefficients  $a_i(t)$ , and residual signal  $r(t)$  [43], i.e.,

$$v(t) = - \sum_{i=1}^{n_A} a_i(t) v(t-i) + r(t). \quad (5.42)$$

The matrices  $\mathbf{A}$  and  $\mathbf{\Sigma}$  can then be rewritten as

$$\mathbf{A} = \begin{bmatrix} 1 & a_1(t) & a_2(t) & \dots & a_{n_A}(t) & 0 & \dots & 0 \\ 0 & 1 & a_1(t-1) & \dots & a_{n_A-1}(t-1) & a_{n_A}(t-1) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & \dots & 1 \end{bmatrix}, \quad (5.43)$$

and

$$\mathbf{\Sigma} = \text{diag}\{\sigma_r^2(t), \dots, \sigma_r^2(1)\} \triangleq E \left\{ \begin{bmatrix} r(t) \\ \vdots \\ r(1) \end{bmatrix} \begin{bmatrix} r(t) & \dots & r(1) \end{bmatrix} \right\}. \quad (5.44)$$

Now, with (5.40), (5.43), and (5.44), the criterion in (5.39) can be rewritten by shifting the prefiltering matrix  $\mathbf{A}$  into the data term  $[\mathbf{y} - \mathbf{U}\hat{\mathbf{f}}(t)]$  as follows,

$$\min_{\hat{\mathbf{f}}(t)} \left\{ [\tilde{\mathbf{y}} - \tilde{\mathbf{U}}\hat{\mathbf{f}}(t)]^T \mathbf{\Sigma}^{-1} [\tilde{\mathbf{y}} - \tilde{\mathbf{U}}\hat{\mathbf{f}}(t)] + [\hat{\mathbf{f}}(t) - \mathbf{f}_0]^T \mathbf{R}_f^{-1} [\hat{\mathbf{f}}(t) - \mathbf{f}_0] \right\}, \quad (5.45)$$

with the prefiltered loudspeaker and microphone signals defined as

$$\tilde{\mathbf{U}} = [\tilde{\mathbf{u}}(t) \quad \tilde{\mathbf{u}}(t-1) \quad \dots \quad \tilde{\mathbf{u}}(1)]^T \triangleq \mathbf{A}\mathbf{U}, \quad (5.46)$$

$$\tilde{\mathbf{y}} = [\tilde{y}(t) \quad \tilde{y}(t-1) \quad \dots \quad \tilde{y}(1)]^T \triangleq \mathbf{A}\mathbf{y}. \quad (5.47)$$

The AR coefficients  $a_i(t)$  and the residual signal variances  $\sigma_r^2(t)$  can be estimated using any of the prediction error identification algorithms described in

[43], e.g., the prediction-error-method-based adaptive filtering algorithm with row operations (PEM-AFROW), which is used in the computer simulations described in Section 5.6. In any case, this estimation procedure, as well as the prefiltering of the loudspeaker and microphone signals as in (5.46) and (5.47), can be entirely decoupled from the RIR identification algorithm, and can be appended to the unregularized algorithms described in [43], as well as to the regularized algorithms proposed in this paper. Therefore, in the sequel, we make use of the prefiltered loudspeaker and microphone signals and the residual signal variance, making abstraction of how these quantities are calculated.

### 5.3.2 True RIR Covariance Matrix $\mathbf{R}_f$

Taking into account that, in the sequel, we will focus on two particular choices of the mean [ $\mathbf{f}_0 = \mathbf{0}$  and  $\mathbf{f}_0 = \hat{\mathbf{f}}(t-1)$ ], we should also consider two expressions for the true RIR covariance matrix:

$$\mathbf{R}_f = E\{\mathbf{f}\mathbf{f}^T\}, \quad (5.48)$$

and

$$\mathbf{R}_f = E\left\{[\mathbf{f} - \hat{\mathbf{f}}(t-1)][\mathbf{f} - \hat{\mathbf{f}}(t-1)]^T\right\}. \quad (5.49)$$

Starting from the expression in (5.48), we will suggest two methods for constructing a diagonal estimate of the true RIR covariance matrix  $\mathbf{R}_f$ . The first method is based on an initial RIR measurement, and the second method on the 3-parameter RIR model proposed in [45]. The motivation for restricting the estimates to be diagonal matrices is twofold. First of all, with a diagonal estimate for  $\mathbf{R}_f$ , the regularization is governed by  $n_F + 1$  parameters, which seems to be a fair compromise between the existing regularization techniques (using one regularization parameter) and the general case of using a non-diagonal estimate for  $\mathbf{R}_f$  [with  $(n_F + 1)^2$  regularization parameters]. The second reason is that the matrix-vector products involving  $\mathbf{R}_f$  and  $\mathbf{R}_f^{-1}$ , as they appear in the algorithms proposed in Sections 5.4 and 5.5, can be computed with  $O(n_F)$  operations if diagonal estimates are used, in contrast to the  $O(n_F^2)$  operations that are needed in the non-diagonal case.

#### Initial RIR Measurement

If the application at hand allows so, probably the most reliable way of constructing prior knowledge on the true RIR, is to perform a batch RIR identification during the initialization of the signal enhancement system. Suppose that in this way an initial RIR estimate

$$\bar{\mathbf{f}} = [\bar{f}_0 \quad \bar{f}_1 \quad \dots \quad \bar{f}_{n_F}]^T \quad (5.50)$$

is obtained, then a diagonal estimate of the true RIR covariance matrix may be constructed as follows:

$$\hat{\mathbf{R}}_{\mathbf{f},\text{init}} = \text{diag}\{\bar{f}_0^2, \bar{f}_1^2, \dots, \bar{f}_{n_F}^2\}. \quad (5.51)$$

The main disadvantage of this method is that it is non-robust to RIR changes, since the estimate of  $\mathbf{R}_{\mathbf{f}}$  is based on one particular measurement. Alternatively, several initial measurements under different conditions (e.g., different microphone or loudspeaker positions) could be averaged before constructing  $\hat{\mathbf{R}}_{\mathbf{f},\text{init}}$ , or, if permitted by the application, the initial measurement could be updated once in a while.

### 3-parameter RIR Model

In [45], we have proposed a very simple RIR model, based on the observation that RIRs have a typical form, which may be characterized by three parameters, as illustrated in Fig. 5.2:

- the *initial delay*  $d$ , which corresponds to the time needed for the loudspeaker sound wave to reach the microphone through a direct path (i.e., without reflections),
- the *direct path attenuation*  $A$ , which determines the peak response in the RIR, and
- the *exponential decay time constant*  $\tau$ , which models the envelope of the reverberant tail of the RIR.

These three parameters may be estimated from the acoustic setup (distance between loudspeaker and microphone, acoustic absorption of the walls, room volume, etc.), e.g., using Sabine's reverberation formulas [50]. Hence they can be considered as prior knowledge. If these three parameters are taken into account, a diagonal estimate of the true RIR covariance matrix may be constructed as

$$\hat{\mathbf{R}}_{\mathbf{f},\mathbf{3}} = A \cdot \text{diag}\left\{\underbrace{\beta, \dots, \beta}_d, 1, \underbrace{e^{-\frac{2}{\tau}}, \dots, e^{-2\frac{n_F-d}{\tau}}}_{n_F-d}\right\}, \quad (5.52)$$

where  $\beta$  is a small number. As an example, the RIR in Fig. 5.2 may be approximated by choosing  $d = 38$ ,  $A = 0.9$ ,  $\tau = 100$  and  $\beta = 10^{-6}$ . The advantage of this method is that the parameter  $\tau$  is invariant to RIR changes due to an arbitrary microphone or loudspeaker movement. Moreover, the other two parameters,  $d$  and  $A$ , are invariant to microphone or loudspeaker movements as long as the distance between the loudspeaker and the microphone remains constant. Hence the 3-parameter model is found to be more robust to RIR changes than the model based on an initial RIR measurement.

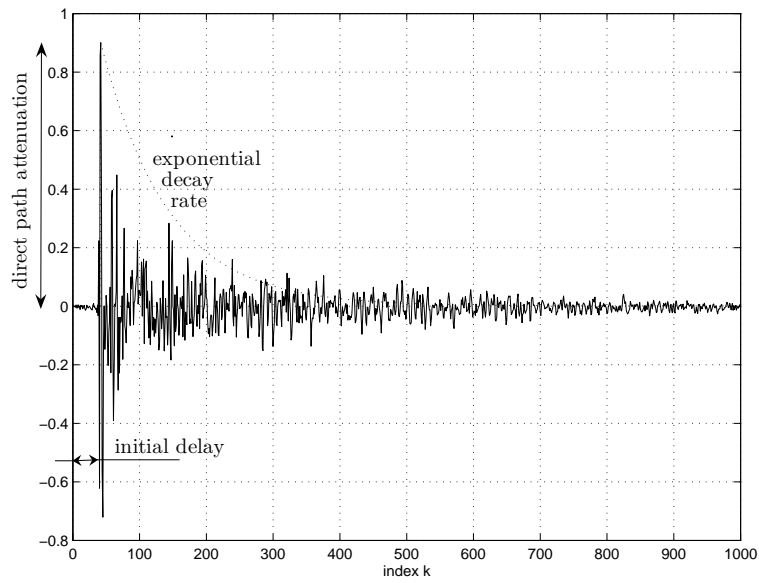


Figure 5.2: Room impulse response characterized by three parameters.

## 5.4 Regularized RLS Algorithms

An MSE optimally weighted and regularized RLS algorithm with an exponential weighting factor  $\lambda$  can be derived from criterion (5.45), namely,

$$\min_{\hat{\mathbf{f}}(t)} \left\{ [\tilde{\mathbf{y}} - \tilde{\mathbf{U}}\hat{\mathbf{f}}(t)]^T \Sigma_\lambda^{-1} [\tilde{\mathbf{y}} - \tilde{\mathbf{U}}\hat{\mathbf{f}}(t)] + [\hat{\mathbf{f}}(t) - \mathbf{f}_0]^T \mathbf{R}_f^{-1} [\hat{\mathbf{f}}(t) - \mathbf{f}_0] \right\}, \quad (5.53)$$

where the exponential data weighting is combined with the MSE optimal weighting in the diagonal matrix  $\Sigma_\lambda^{-1}$ :

$$\Sigma_\lambda^{-1} \triangleq \text{diag} \left\{ \frac{1}{\sigma_r^2(t)}, \frac{\lambda}{\sigma_r^2(t-1)}, \dots, \frac{\lambda^{t-1}}{\sigma_r^2(1)} \right\}. \quad (5.54)$$

The derivation of the RLS algorithm corresponding to (5.53)-(5.54) is omitted here, since it is quite similar to deriving the standard RLS algorithm (see, e.g., [9]) or the leaky RLS algorithm [10]. The resulting algorithm is given by

$$\hat{\mathbf{f}}(t) = \mathbf{f}_0 + [\mathbf{I} - (1-\lambda)\mathbf{R}^{-1}(t)\mathbf{R}_f^{-1}][\hat{\mathbf{f}}(t-1) - \mathbf{f}_0] + \frac{1}{\sigma_r^2(t)}\mathbf{R}^{-1}(t)\tilde{\mathbf{u}}(t)\tilde{\varepsilon}(t), \quad (5.55)$$

$$\mathbf{R}(t) = \lambda\mathbf{R}(t-1) + \frac{1}{\sigma_r^2(t)}\tilde{\mathbf{u}}(t)\tilde{\mathbf{u}}^T(t) + (1-\lambda)\mathbf{R}_f^{-1}, \quad (5.56)$$

$$\tilde{\varepsilon}(t) = \tilde{y}(t) - \tilde{\mathbf{u}}^T(t)\hat{\mathbf{f}}(t-1), \quad (5.57)$$

where  $\tilde{\mathbf{u}}(t)$ ,  $\tilde{y}(t)$ , and  $\tilde{\varepsilon}(t)$  represent the loudspeaker signal vector, microphone signal, and a priori residual signal, respectively, after prefiltering with the prediction error filter in the top row of  $\mathbf{A}$  in (5.43), and  $\sigma_r^2(t)$  represents the residual signal variance from the source signal linear prediction (see Section 5.3.1 and [43]).

To point out the relationship of the above algorithm with the existing regularized RLS algorithms, described in Section 5.1, we will focus on two particular choices for the mean  $\mathbf{f}_0$  of the true RIR distribution. Choosing  $\mathbf{f}_0 = \mathbf{0}$ , leads to the MSE optimal *Tikhonov regularized RLS (TR-RLS) algorithm*, shown in the upper part of Table 5.1, which is a generalization of the traditional Tikhonov regularized RLS algorithm (or leaky RLS algorithm) in (5.15)-(5.17). In Section 5.1, the traditional Levenberg-Marquardt regularized RLS algorithm was considered as an approximation to the traditional Tikhonov regularized RLS algorithm by omitting the correction term in the RIR weight update. However, we now observe that an MSE optimal *Levenberg-Marquardt regularized RLS (LMR-RLS) algorithm* can also be obtained from the algorithm in (5.55)-(5.57) by setting  $\mathbf{f}_0 = \hat{\mathbf{f}}(t-1)$ . The resulting LMR-RLS algorithm is shown in the lower part of Table 5.1.

A severe drawback of all the regularized RLS algorithms, either with the traditional scaled identity matrix, or with a regularization matrix incorporating

Table 5.1: MSE Optimally Regularized RLS Algorithms

---

**TR-RLS - Tikhonov Regularized Recursive Least Squares**

$$\begin{aligned}\hat{\mathbf{f}}(t) &= \hat{\mathbf{f}}(t-1) + \mathbf{R}^{-1}(t) \left[ \frac{1}{\sigma_r^2(t)} \tilde{\mathbf{u}}(t) \tilde{\varepsilon}(t) - (1-\lambda) \mathbf{R}_f^{-1} \hat{\mathbf{f}}(t-1) \right], \\ \mathbf{R}(t) &= \lambda \mathbf{R}(t-1) + \frac{1}{\sigma_r^2(t)} \tilde{\mathbf{u}}(t) \tilde{\mathbf{u}}^T(t) + (1-\lambda) \mathbf{R}_f^{-1}, \\ \tilde{\varepsilon}(t) &= \tilde{y}(t) - \tilde{\mathbf{u}}^T(t) \hat{\mathbf{f}}(t-1).\end{aligned}$$

**LMR-RLS - Levenberg-Marquardt Regularized Recursive Least Squares**

$$\begin{aligned}\hat{\mathbf{f}}(t) &= \hat{\mathbf{f}}(t-1) + \frac{1}{\sigma_r^2(t)} \mathbf{R}^{-1}(t) \tilde{\mathbf{u}}(t) \tilde{\varepsilon}(t), \\ \mathbf{R}(t) &= \lambda \mathbf{R}(t-1) + \frac{1}{\sigma_r^2(t)} \tilde{\mathbf{u}}(t) \tilde{\mathbf{u}}^T(t) + (1-\lambda) \mathbf{R}_f^{-1}, \\ \tilde{\varepsilon}(t) &= \tilde{y}(t) - \tilde{\mathbf{u}}^T(t) \hat{\mathbf{f}}(t-1).\end{aligned}$$


---

prior knowledge, is that the inversion of the regularized correlation matrix  $\mathbf{R}(t)$  cannot be performed cheaply, i.e., in  $O(n_F^2)$  operations, based on the matrix inversion lemma (MIL). While in the standard RLS algorithm the correlation matrix is updated with a rank-one matrix in each iteration, in the regularized algorithms a full-rank regularization matrix is added in each iteration. Nevertheless, some procedures have been proposed to reduce the complexity of RLS algorithms regularized with a scaled identity matrix, to  $O(n_F^2)$  operations per iteration. We illustrate how two such procedures can also be applied in the case of an MSE optimal regularization matrix  $\mathbf{R}_f^{-1}$  that is not necessarily a scaled identity matrix. In fact, these procedures can be applied to arbitrary, non-diagonal regularization matrices, and obviously also to the diagonal matrices  $\hat{\mathbf{R}}_{f,\text{init}}$  and  $\hat{\mathbf{R}}_{f,3}$  described in Section 5.3.

- A first procedure starts from the decomposition of the  $(n_F + 1) \times (n_F + 1)$  regularization matrix in a sum of  $n_F + 1$  rank-one matrices, or more specifically, a sum of  $n_F + 1$  outer vector products. This decomposition is easily constructed for a diagonal regularization matrix, e.g., for the inverse of the matrix  $\hat{\mathbf{R}}_{f,\text{init}}$  proposed in (5.51):

$$\hat{\mathbf{R}}_{f,\text{init}}^{-1} = \sum_{k=0}^{n_F} \phi_k \phi_k^T, \quad (5.58)$$

with

$$\phi_k^T \triangleq [0 \quad \dots \quad 0 \quad \bar{f}_k^{-1} \quad 0 \quad \dots \quad 0]. \quad (5.59)$$

The regularized RLS correlation matrix update can then be approximated by adding each outer product  $\phi_k \phi_k^T$  with appropriate scaling only once in  $n_F + 1$  iteration steps, instead of adding the full rank matrix  $(1 - \lambda)\mathbf{R}_f^{-1}$  in each iteration step:

$$\mathbf{R}(t) = \lambda \mathbf{R}(t-1) + \frac{1}{\sigma_r^2(t)} \tilde{\mathbf{u}}(t) \tilde{\mathbf{u}}^T(t) + (1 - \lambda)(n_F + 1) \phi_k \phi_k^T, \quad (5.60)$$

$$k = \text{mod}(t, n_F + 1).$$

Since this approximate update involves two rank-one matrix updates, applying the MIL to (5.60) yields an inverse correlation matrix update which requires the inversion of a  $2 \times 2$  matrix, and can hence be computed in  $O(n_F^2)$  operations. This method was first proposed in [12] and adapted later on to the so-called dynamic regularization method for acoustic echo cancellation [18].

- A second procedure was proposed in the context of regularization of fast recursive least squares algorithms [46], and consists in replacing the exponential data window with a rectangular window. If the adaptive filter input correlation matrix is initialized to the regularization matrix, i.e.,  $\mathbf{R}(0) = \mathbf{R}_f^{-1}$ , then no restoration of the regularization matrix needs to be performed during the operation of the algorithm, if a sliding window



approach is followed. Hence the correlation matrix update consists of two rank-one matrix updates:

$$\begin{aligned} \mathbf{R}(t) &= \mathbf{R}(t-1) \\ &+ \frac{1}{\sigma_r^2(t)} \tilde{\mathbf{u}}(t) \tilde{\mathbf{u}}^T(t) - \frac{1}{\sigma_r^2(t-n_F-1)} \tilde{\mathbf{u}}(t-n_F-1) \tilde{\mathbf{u}}^T(t-n_F-1), \end{aligned} \quad (5.61)$$

and applying the MIL leads to an  $O(n_F^2)$  update of the inverse correlation matrix, which involves the inversion of a  $2 \times 2$  matrix. However, this method is not very appealing from a numerical point of view, since a sliding rectangular data window approach suffers from linear round-off error build-up.

## 5.5 Regularized APA and NLMS Algorithms

Considering the affine projection algorithm (APA) and the normalized least mean squares (NLMS) algorithm as underdetermined recursive least squares (URLS) algorithms [16], it seems straightforward to derive regularized APA and NLMS algorithms from the same criterion that led to the MSE optimally weighted and regularized RLS algorithms in Section 5.4, and to exploit the underdetermined nature of the algorithms to reduce the dimension of the input correlation matrix that has to be inverted. Rewriting the criterion in (5.45) for the underdetermined case, leads to

$$\begin{aligned} \min_{\hat{\mathbf{f}}(t)} \left\{ [\tilde{\mathbf{y}}_M(t) - \tilde{\mathbf{U}}_M^T(t) \hat{\mathbf{f}}(t)]^T \boldsymbol{\Sigma}_M^{-1}(t) [\tilde{\mathbf{y}}_M(t) - \tilde{\mathbf{U}}_M^T(t) \hat{\mathbf{f}}(t)] \right. \\ \left. + [\hat{\mathbf{f}}(t) - \mathbf{f}_0]^T \mathbf{R}_f^{-1} [\hat{\mathbf{f}}(t) - \mathbf{f}_0] \right\}, \end{aligned} \quad (5.62)$$

with  $M$  the projection order, as defined earlier. Consistent with the notation in (5.22)-(5.23), we define the prefiltered data matrices  $\tilde{\mathbf{U}}_M(t)$  and  $\tilde{\mathbf{y}}_M(t)$  as

$$\tilde{\mathbf{U}}_M(t) \triangleq [\tilde{\mathbf{u}}(t) \quad \tilde{\mathbf{u}}(t-1) \quad \dots \quad \tilde{\mathbf{u}}(t-M+1)] \quad (5.63)$$

$$= [\mathbf{u}(t) \quad \mathbf{u}(t-1) \quad \dots \quad \mathbf{u}(t-M-n_A+1)] \mathbf{A}_M^T(t), \quad (5.64)$$

$$\tilde{\mathbf{y}}_M(t) \triangleq [\tilde{y}(t) \quad \tilde{y}(t-1) \quad \dots \quad \tilde{y}(t-M+1)]^T \quad (5.65)$$

$$= \mathbf{A}_M(t) [y(t) \quad y(t-1) \quad \dots \quad y(t-M-n_A+1)]^T, \quad (5.66)$$

where  $\mathbf{A}_M(t)$  is defined as the  $M \times (M+n_A)$  top left submatrix of  $\mathbf{A}$  in (5.43). The diagonal weighting matrix  $\boldsymbol{\Sigma}_M(t)$  appearing in (5.62) is defined as

$$\boldsymbol{\Sigma}_M(t) \triangleq \text{diag}\{\sigma_r^2(t), \dots, \sigma_r^2(t-M+1)\}. \quad (5.67)$$

Minimizing the criterion in (5.62), and subsequently applying the MIL to reduce the dimension of the  $(n_F+1) \times (n_F+1)$  inverse correlation matrix to  $M \times M$ ,

leads to the following underdetermined estimate:

$$\hat{\mathbf{f}}(t) = \mathbf{f}_0 + \mathbf{R}_f \tilde{\mathbf{U}}_M(t) [\tilde{\mathbf{U}}_M^T(t) \mathbf{R}_f \tilde{\mathbf{U}}_M(t) + \boldsymbol{\Sigma}_M(t)]^{-1} [\tilde{\mathbf{y}}_M(t) - \tilde{\mathbf{U}}_M^T(t) \mathbf{f}_0]. \quad (5.68)$$

Note that this estimate is non-recursive, i.e., it does not depend on the previous estimate  $\hat{\mathbf{f}}(t-1)$ . A recursive algorithm can be obtained by explicitly bringing in the dependency on  $\hat{\mathbf{f}}(t-1)$ , as follows:

$$\begin{aligned} \hat{\mathbf{f}}(t) &= \mathbf{f}_0 + \mathbf{R}_f \tilde{\mathbf{U}}_M(t) [\tilde{\mathbf{U}}_M^T(t) \mathbf{R}_f \tilde{\mathbf{U}}_M(t) + \boldsymbol{\Sigma}_M(t)]^{-1} \\ &\quad [\tilde{\mathbf{y}}_M(t) - \tilde{\mathbf{U}}_M^T(t) \hat{\mathbf{f}}(t-1) + \tilde{\mathbf{U}}_M^T(t) \hat{\mathbf{f}}(t-1) - \tilde{\mathbf{U}}_M^T(t) \mathbf{f}_0], \\ &= \mathbf{f}_0 + \mathbf{R}_f \tilde{\mathbf{U}}_M(t) [\tilde{\mathbf{U}}_M^T(t) \mathbf{R}_f \tilde{\mathbf{U}}_M(t) + \boldsymbol{\Sigma}_M(t)]^{-1} \mathbf{U}_M^T(t) [\hat{\mathbf{f}}(t-1) - \mathbf{f}_0] \\ &\quad + \mathbf{R}_f \tilde{\mathbf{U}}_M(t) [\tilde{\mathbf{U}}_M^T(t) \mathbf{R}_f \tilde{\mathbf{U}}_M(t) + \boldsymbol{\Sigma}_M(t)]^{-1} [\tilde{\mathbf{y}}_M(t) - \tilde{\mathbf{U}}_M^T(t) \hat{\mathbf{f}}(t-1)]. \end{aligned} \quad (5.69)$$

Finally, by again applying the MIL to the second term on the right hand side, we rewrite the estimate in (5.70) into an expression which is computationally less interesting, but more convenient to understand the estimator's properties:

$$\begin{aligned} \hat{\mathbf{f}}(t) &= \mathbf{f}_0 + \{ \mathbf{I} - [\tilde{\mathbf{U}}_M(t) \boldsymbol{\Sigma}_M^{-1}(t) \mathbf{U}_M^T(t) + \mathbf{R}_f^{-1}]^{-1} \mathbf{R}_f^{-1} \} [\hat{\mathbf{f}}(t-1) - \mathbf{f}_0] \\ &\quad + \mathbf{R}_f \tilde{\mathbf{U}}_M(t) [\tilde{\mathbf{U}}_M^T(t) \mathbf{R}_f \tilde{\mathbf{U}}_M(t) + \boldsymbol{\Sigma}_M(t)]^{-1} [\tilde{\mathbf{y}}_M(t) - \tilde{\mathbf{U}}_M^T(t) \hat{\mathbf{f}}(t-1)]. \end{aligned} \quad (5.71)$$

From (5.71), it can be seen that the minimizing estimate of the underdetermined criterion in (5.62), consists of three terms: the mean value  $\mathbf{f}_0$  of the true RIR distribution, a leakage term depending on the deviation  $[\hat{\mathbf{f}}(t-1) - \mathbf{f}_0]$  of the previous estimate from the mean value, and a proportionate adaptation term. However, from extensive simulations, we have concluded that leakage (even in the simple case of the leaky LMS algorithm [11]) does not have a beneficial effect on the convergence behaviour of URLS algorithms in room acoustic applications. We will therefore not consider Tikhonov regularized URLS algorithms here (even though their update formulae are readily derived with  $\mathbf{f}_0 = \mathbf{0}$ ). The leakage term in (5.71) disappears by choosing  $\mathbf{f}_0 = \hat{\mathbf{f}}(t-1)$ , and hence we end up with a Levenberg-Marquardt type of regularization. If we finally introduce a relaxation factor  $\mu$ , then the resulting algorithms are the *Levenberg-Marquardt regularized affine projection algorithm (LMR-APA)* and, when the projection order is set to  $M = 1$ , the *Levenberg-Marquardt regularized normalized least mean squares (LMR-NLMS) algorithm*, which are shown in Table 5.2. Note that the traditionally regularized APA and NLMS algorithms, described in Section 5.1, can be obtained as special cases of these algorithms, by omitting the prefiltering step, and by choosing  $\boldsymbol{\Sigma}_M(t) = \sigma \mathbf{I}$  and  $\mathbf{R}_f = \nu \mathbf{I}$  such that  $\sigma \nu^{-1} = \alpha$ .

The LMR-APA and LMR-NLMS algorithms are closely related to the proportionate APA (PAPA) [21] and proportionate NLMS (PNLMS) [47] algorithms. In the PNLMS RIR weight update,

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mu \frac{\mathbf{G}(t) \mathbf{u}(t) \varepsilon(t)}{\mathbf{u}^T(t) \mathbf{G}(t) \mathbf{u}(t) + \alpha}, \quad (5.72)$$

Table 5.2: MSE Optimally Regularized Underdetermined RLS Algorithms

**LMR-APA - Levenberg-Marquardt Regularized APA**

$$\begin{aligned}\hat{\mathbf{f}}(t) &= \hat{\mathbf{f}}(t-1) + \mu \mathbf{R}_f \tilde{\mathbf{U}}_M(t) [\tilde{\mathbf{U}}_M^T(t) \mathbf{R}_f \tilde{\mathbf{U}}_M(t) + \boldsymbol{\Sigma}_M(t)]^{-1} \tilde{\boldsymbol{\varepsilon}}_M(t), \\ \tilde{\boldsymbol{\varepsilon}}_M(t) &= \tilde{\mathbf{y}}_M(t) - \tilde{\mathbf{U}}_M^T(t) \hat{\mathbf{f}}(t-1).\end{aligned}$$

**LMR-NLMS - Levenberg-Marquardt Regularized NLMS**

$$\begin{aligned}\hat{\mathbf{f}}(t) &= \hat{\mathbf{f}}(t-1) + \mu \frac{\mathbf{R}_f \tilde{\mathbf{u}}(t) \tilde{\boldsymbol{\varepsilon}}(t)}{\tilde{\mathbf{u}}^T(t) \mathbf{R}_f \tilde{\mathbf{u}}(t) + \sigma_r^2(t)}, \\ \tilde{\boldsymbol{\varepsilon}}(t) &= \tilde{y}(t) - \tilde{\mathbf{u}}^T(t) \hat{\mathbf{f}}(t-1).\end{aligned}$$

the diagonal matrix  $\mathbf{G}(t)$  is constructed as follows:

$$\mathbf{G}(t) = \frac{1}{\bar{g}(t)} \text{diag}\{g_0(t), \dots, g_{n_F}(t)\}, \quad (5.73)$$

with, for  $k = 0, \dots, n_F$ ,

$$g_k(t) \triangleq \max\left\{\rho \cdot \max\{\delta, |\hat{f}_0(t-1)|, \dots, |\hat{f}_{n_F}(t-1)|\}, |\hat{f}_k(t-1)|\right\}, \quad (5.74)$$

$$\bar{g}(t) \triangleq \frac{1}{n_F + 1} \sum_{k=0}^{n_F} g_k(t), \quad (5.75)$$

where  $\rho$  and  $\delta$  are small positive parameters. This choice of  $\mathbf{G}(t)$  was made somewhat intuitively by Duttweiler [47], with the aim of allocating a larger portion of the available adaptation energy to larger adaptive filter weights, and hence to speed up the convergence. Later on, Chen et al. [23] provided an interpretation of the above choice of  $\mathbf{G}(t)$  in terms of Bayesian priors. Suppose that some prior knowledge is available on the true RIR through its probability density function (PDF)  $p(\mathbf{f})$ , then it is advantageous to choose the PNLMS parameters  $g_k(t)$  proportional to the negative logarithm of the prior distribution [23], i.e.,

$$g_k(t) \propto -\ln[p(f_k)], \quad k = 0, \dots, n_F. \quad (5.76)$$

It can be shown that, slightly deviating from the treatment in [23],<sup>1</sup>

$$-\ln[p(f_k)] \sim |f_k - f_{0,k}| \quad \text{for a Laplacian prior,} \quad (5.77)$$

$$\sim (f_k - f_{0,k})^2 \quad \text{for a Gaussian prior,} \quad (5.78)$$

where  $f_k$  and  $f_{0,k}$  represent the  $(k+1)$ -th elements of the vectors  $\mathbf{f}$  and  $\mathbf{f}_0$ , respectively. In the PNLMS algorithm, the negative logarithm of a Laplacian prior on the true RIR is approximated using a smoothed estimate of  $|f_k|$ , given in (5.74), which is obtained by taking the  $L_\infty$ -norm of  $|\hat{f}_k(t-1)|$ ,  $\rho|\hat{\mathbf{f}}(t-1)|$  and  $\rho\delta$ , hence implicitly assuming that the mean  $\mathbf{f}_0 = \mathbf{0}$ . In other words, the prior knowledge on the true RIR is extracted from the previous RIR estimate  $\hat{\mathbf{f}}(t-1)$ . Obviously, the prior knowledge could also be constructed using a Gaussian prior, in which case the matrix  $\mathbf{G}(t)$  is in fact a (normalized) estimate of the true RIR covariance matrix  $\mathbf{R}_f$  in (5.49). Also, the smoothed estimate could be calculated using an  $L_2$ -norm instead of an  $L_\infty$ -norm, which appears to have a beneficial effect on the PNLMS convergence.

From this point of view, the LMR-NLMS algorithm provides a new choice for the matrix  $\mathbf{G}(t)$  in the PNLMS algorithm, which is optimal in the sense of minimizing the underdetermined MSE optimally weighted and regularized criterion in (5.62). The above comparison of LMR-NLMS with PNLMS can be done similarly for the LMR-APA and PAPA algorithms. An important difference with the existing proportionate adaptation algorithms is that in the proposed LMR-APA and LMR-NLMS algorithms the regularization matrix  $\mathbf{R}_f$  is fixed. As a consequence, the proposed algorithms may respond somewhat slower to a RIR change, but on the other hand they are much less computationally demanding than the PAPA and PNLMS algorithms, in which  $\mathbf{G}(t)$  is recalculated with (5.74) and (5.75) in each iteration of the adaptive filter.

## 5.6 Simulation Results

We present the results of five computer simulations, in which a comparison of the proposed and existing adaptive filtering algorithms is made in a variety of RIR identification scenarios. The performance measure for comparing the different algorithms is the misadjustment, which is defined as the normalized Euclidian distance between the estimated and true RIR on a logarithmic scale:

$$\text{misadjustment (dB)} = 20 \log_{10} \frac{\|\hat{\mathbf{f}}(t) - \mathbf{f}\|}{\|\mathbf{f}\|}. \quad (5.79)$$

<sup>1</sup>Similarity between two scalar functions  $\zeta(f_k)$  and  $\eta(f_k)$  is denoted with the  $\sim$  symbol, and defined as follows:  $\zeta(f_k) \sim \eta(f_k) \Leftrightarrow d(\zeta(f_k^1), \zeta(f_k^2)) = r d(\eta(f_k^1), \eta(f_k^2))$ , where  $f_k^1$  and  $f_k^2$  represent different realizations of the stochastic variable  $f_k$ ,  $r$  is a scaling factor, and the distance operator  $d(x, y)$  is defined as the Minkowski or  $p$ -norm distance, which reduces to  $|x - y|$  in the scalar case, for any  $p \in \mathbb{N}$ .

In all simulations, the sampling frequency  $f_s = 8$  kHz.

### 5.6.1 Regularized RLS and PEM-AFROW Algorithms for Acoustic Echo Cancellation in a Stationary Environment

In these two simulations, acoustic echo cancellation is performed using regularized RLS and PEM-AFROW algorithms in a stationary echo path situation. The PEM-AFROW algorithm is a recursive prediction error identification algorithm, in which an AR model of the near-end signal is estimated for prefiltering the loudspeaker and microphone signals as in (5.46) and (5.47), see [2],[42],[43]. The echo path is equal to the RIR plotted in Fig. 5.2, which is of known length  $n_F + 1 = 1000$  and is taken from the measured RIR database in [51]. An exponential data window is applied with an effective length of three times the RIR length, i.e.,  $\lambda = 0.9997$ . The far-end signal is a 1.23 s male speech signal (equivalent to  $N = 9831$  samples), whereas the near-end signal is either a stationary Gaussian white noise (GWN) signal or a male speech signal, both with variance  $\sigma_v^2 = 10^{-3}$ . This results in a signal-to-noise ratio of 18 dB at the microphone, i.e.,

$$\text{SNR}_{\text{mic}} \triangleq 10 \log_{10} \frac{\sum_{t=1}^N |x(t)|^2}{\sum_{t=1}^N |v(t)|^2} = 18 \text{ dB}. \quad (5.80)$$

As reference algorithms, we consider the standard RLS algorithm and the traditional Tikhonov and Levenberg-Marquardt regularized RLS algorithms with a scaled identity regularization matrix and  $\alpha = \sigma_v^2$ , which we will further call the TR-RLS  $\alpha \mathbf{I}$  and LMR-RLS  $\alpha \mathbf{I}$  algorithms.

In a first simulation, with a near-end GWN signal, these reference algorithms are compared to the proposed TR-RLS  $\hat{\mathbf{R}}_{\mathbf{f},3}$  and LMR-RLS  $\hat{\mathbf{R}}_{\mathbf{f},3}$  algorithms, in which the 3-parameter model given in (5.52) is used for constructing a non-identity regularization matrix with  $d = 38$ ,  $A = 0.9$ ,  $\tau = 100$  and  $\beta = 10^{-6}$ . Finally, also the TR-RLS  $\hat{\mathbf{R}}_{\mathbf{f},\text{true}}$  and LMR-RLS  $\hat{\mathbf{R}}_{\mathbf{f},\text{true}}$  algorithms are simulated, in which regularization is performed using a diagonal matrix as in (5.51) with  $\bar{\mathbf{f}}$  equal to the true RIR  $\mathbf{f}$ . The latter algorithms illustrate the “best-case” performance that can be obtained with the proposed regularized RLS algorithms in a stationary environment, which could be realized only if an exact initial measurement of the unknown RIR were available. In all regularized algorithms, we assume exact knowledge of the near-end GWN signal variance  $\sigma_v^2$ . Since the near-end signal is a white noise signal, no prefiltering is applied to the loudspeaker and microphone signals.

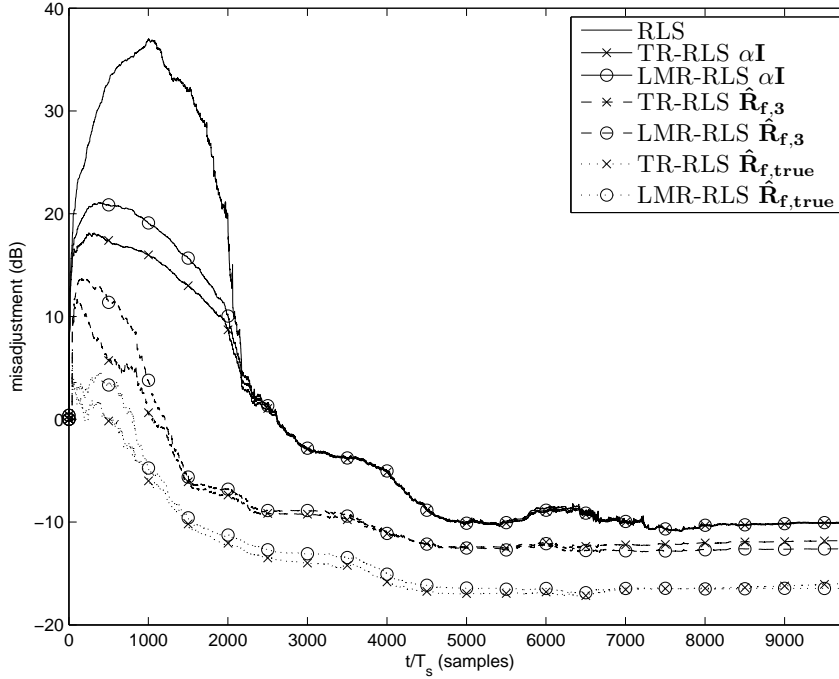


Figure 5.3: Convergence curves of regularized RLS algorithms for an AEC application with a near-end GWN signal in a stationary environment.

The convergence curves of the different RLS algorithms are shown in Fig. 5.3. The main observation is that, as expected, using more prior knowledge on the unknown RIR leads to faster convergence and to a smaller steady-state misadjustment. The initial divergence of the algorithms is due to the lack of excitation in the first 1000 recursion steps (the loudspeaker signal vector  $\mathbf{u}(t)$  contains  $n_F + 1 - t$  zeros if  $t < n_F + 1$ ). This effect is less pronounced for the regularized algorithms, which moreover recover faster when more prior knowledge is available. After the initial divergence effect has disappeared, it can be seen that the  $\alpha\mathbf{I}$  regularized algorithms do not perform better than the standard RLS algorithm. Finally, we note that the difference between Tikhonov and Levenberg-Marquardt regularization appears to be negligible.

In a second simulation, a near-end speech signal is applied, such that pre-filtering of the loudspeaker and microphone signals with the inverse near-end signal model is desirable. The regularized PEM-AFROW algorithms are implemented with a Gauss-Newton update, to allow for fair comparison with the regularized RLS algorithms, and the AR model order is set to 12. Since the difference between Tikhonov and Levenberg-Marquardt regularization is small,

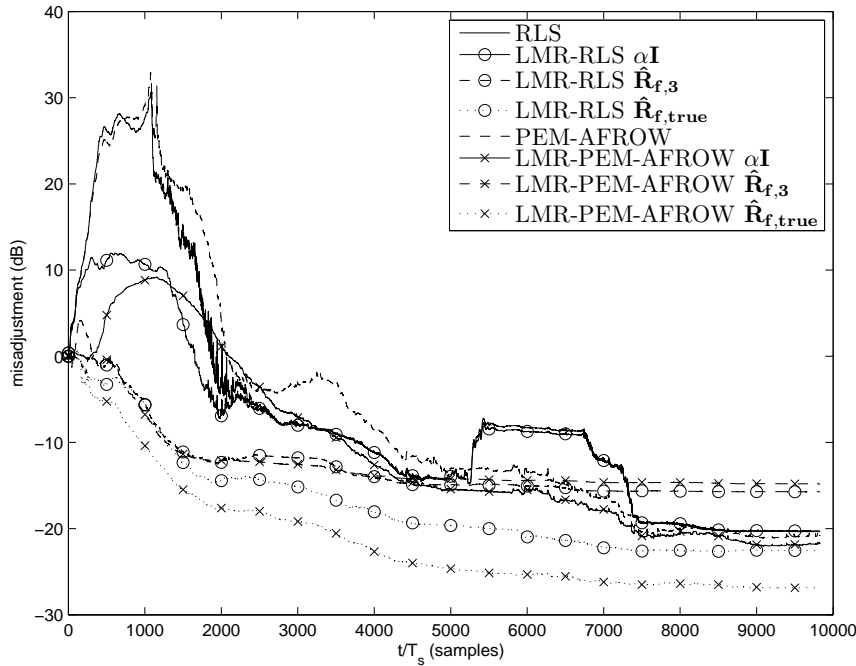


Figure 5.4: Convergence curves of regularized RLS and PEM-AFROW algorithms for an AEC application with a near-end speech signal in a stationary environment.

we have plotted in Fig. 5.4 only the convergence curves of the Levenberg-Marquardt regularized algorithms. A first observation is that the unregularized RLS algorithm shows a slightly better convergence than the unregularized PEM-AFROW algorithm, except in the time interval  $t/T_s \in [5250, 7250]$ , where a high-frequency artifact in the near-end signal causes divergence of the RLS algorithm, without affecting the double-talk-robust PEM-AFROW algorithm. Another issue that can be remarked from Fig. 5.4 is that the  $\hat{\mathbf{R}}_{f,3}$  regularized algorithms show better initial convergence than the unregularized and  $\alpha \mathbf{I}$  regularized ones, but afterwards tend to converge to a biased solution.

### 5.6.2 Regularized NLMS and PEM-AFROW Algorithms for Acoustic Echo Cancellation in a Nonstationary Environment

In the next two simulations, the performance of regularized NLMS and PEM-AFROW algorithms is compared for acoustic echo cancellation in a nonstationary-

ary environment. We switch between three different room impulse responses of known length  $n_F + 1 = 1000$ , that were measured in our acoustic lab. The first RIR change occurs at  $t/T_s = 1.2 \cdot 10^5$  samples, and consists in a change of 75 cm in the microphone position, in such a way that the distance between the loudspeaker and the microphone remains constant. After the second RIR change, occurring at  $t/T_s = 2.5 \cdot 10^5$  samples, the loudspeaker and microphone positions remain unchanged, but the room is made more reverberant by decreasing the absorption coefficients of the walls and ceiling (the  $T60$  reverberation times of the first two RIRs and of the third RIR were roughly estimated as  $T60_{1,2} = 0.225$  s and  $T60_3 = 0.3$  s, respectively). In this simulation, the far-end signal is a 46 s male speech signal (equivalent to  $N = 368320$  samples), and the near-end signal is either a stationary GWN signal, or a male speech signal, both with variance  $\sigma_v^2 = 3 \cdot 10^{-5}$ , resulting in  $\text{SNR}_{\text{mic}} = 16$  dB.

The reference algorithms are now the (unregularized) standard NLMS algorithm, the proportionate NLMS algorithm with  $\rho = 5/(n_F + 1)$  and  $\delta = 0.01$  as suggested in [47], and the traditionally regularized NLMS algorithm as given in (5.26)-(5.27) with  $\alpha = \sigma_v^2$ , which is further called LMR-NLMS  $\alpha\mathbf{I}$ . In the first simulation, with a near-end GWN signal, these algorithms are compared to the LMR-NLMS  $\hat{\mathbf{R}}_{\mathbf{f},3}$  and LMR-NLMS  $\hat{\mathbf{R}}_{\mathbf{f},\text{true}}$  algorithms in which the regularization matrix is based on the 3-parameter model (with  $d = 75$ ,  $A = 0.1022$ ,  $\tau = 70$  and  $\beta = 10^{-6}$ ), and on exact knowledge of the first RIR, respectively. The regularization matrix is not altered after the two RIR changes, such that the robustness of the different regularized algorithms w.r.t. RIR changes can be evaluated. The step size  $\mu$  is individually tuned for each of the algorithms such that the excess MSE in a stationary environment would approximately be the same for all algorithms. Again, in this case, no prefiltering is applied since the near-end signal is white noise.

The convergence curves and step sizes are shown in Fig. 5.5. First of all, we observe that the improvement in convergence speed of the existing PNLMS and LMR-NLMS  $\alpha\mathbf{I}$  algorithms, as compared to the standard NLMS algorithm, is small. A significantly better convergence behaviour is obtained with the proposed LMR-NLMS  $\hat{\mathbf{R}}_{\mathbf{f},3}$  and LMR-NLMS  $\hat{\mathbf{R}}_{\mathbf{f},\text{true}}$  algorithms in the time interval  $t/T_s = [0, 1.2 \cdot 10^5]$  samples, where the regularization matrix of the proposed algorithms is based on the “correct” RIR. After the first RIR change, the LMR-NLMS  $\hat{\mathbf{R}}_{\mathbf{f},\text{true}}$  algorithm’s performance decreases dramatically, whereas the LMR-NLMS  $\hat{\mathbf{R}}_{\mathbf{f},3}$  algorithm converges equally fast as in the first time interval. This is not much of a surprise, since the regularization matrix  $\hat{\mathbf{R}}_{\mathbf{f},\text{true}}$  based on the true first RIR will be a bad model for the second RIR covariance matrix, whereas the 3-parameter model regularization matrix  $\hat{\mathbf{R}}_{\mathbf{f},3}$  of the first RIR will still be valid for the second RIR. Indeed, the microphone repositioning does not alter any of the three parameters on which  $\hat{\mathbf{R}}_{\mathbf{f},3}$  is based, since the distance between the loudspeaker and the microphone remains constant. However, after the second RIR change the parameter  $\tau$  in  $\hat{\mathbf{R}}_{\mathbf{f},3}$  will have an in-



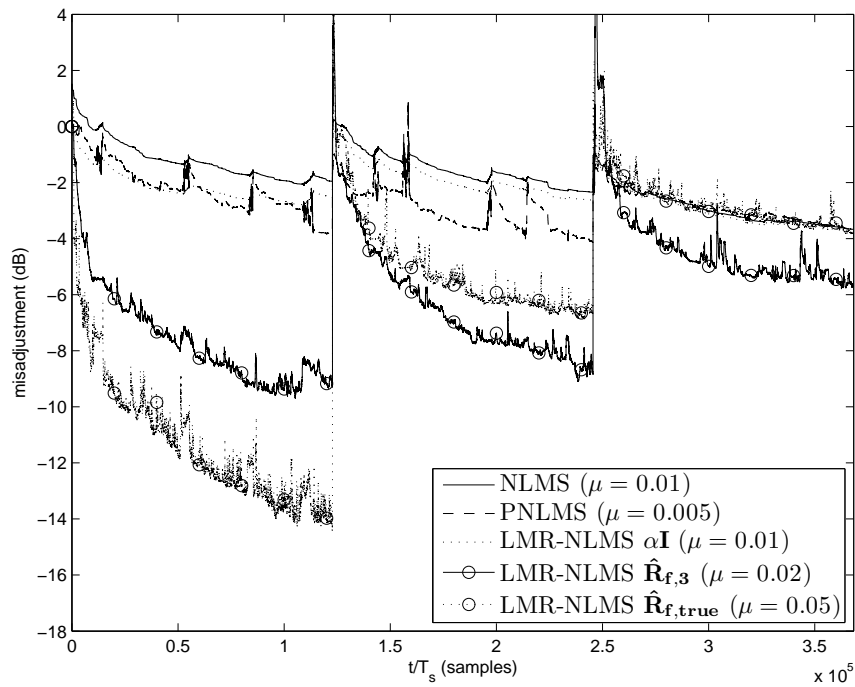


Figure 5.5: Convergence curves of regularized NLMS algorithms for an AEC application with a near-end GWN signal and two echo path changes.

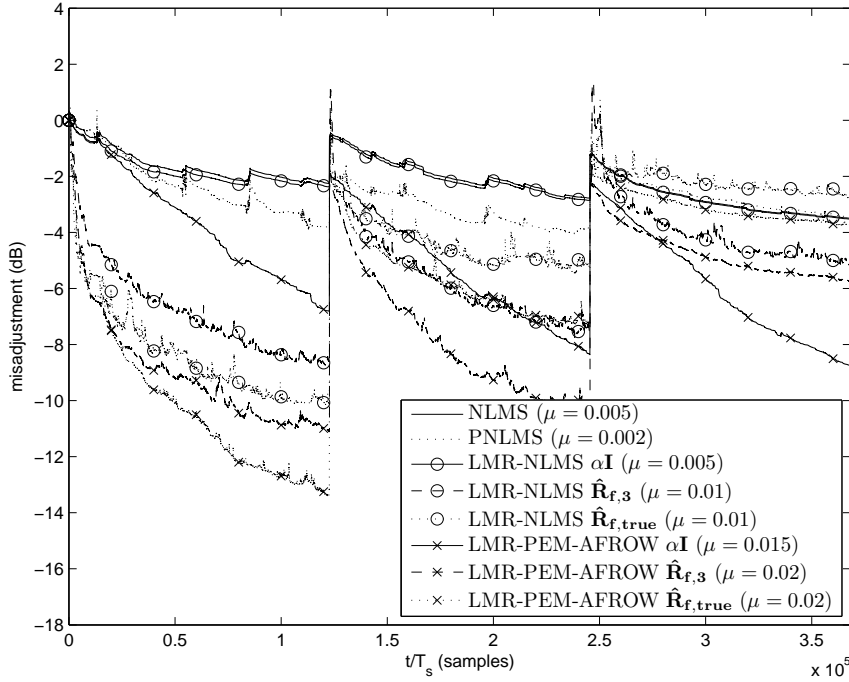


Figure 5.6: Convergence curves of regularized NLMS and PEM-AFROW algorithms for an AEC application with a near-end speech signal and two echo path changes.

accurate value, since the room reverberation has increased. This clearly affects the LMR-NLMS  $\hat{\mathbf{R}}_{f,3}$  convergence speed, but still this algorithm outperforms all other algorithms.

In the case of a near-end speech signal, the above algorithms are in addition compared to the LMR-PEM-AFROW  $\alpha\mathbf{I}$ , LMR-PEM-AFROW  $\hat{\mathbf{R}}_{f,3}$ , and LMR-PEM-AFROW  $\hat{\mathbf{R}}_{f,true}$ . The PEM-AFROW algorithm again employs an AR(12) model for the near-end signal linear prediction, but now a stochastic gradient update is used for adapting the RIR estimate. The convergence curves shown in Fig. 5.6 reveal more or less the same issues as in the white noise case. In addition, we observe that the LMR-PEM-AFROW algorithms perform significantly better than their LMR-NLMS counterparts, for the same choice of the regularization matrix.

### 5.6.3 Regularized PEM-AFROW Algorithms for Adaptive Feedback Cancellation in a Stationary Environment

In this final simulation, the proposed LMR-PEM-AFROW algorithm is applied in a closed-loop scenario for performing adaptive feedback cancellation. The source signal  $v(t)$ , equal to the same male speech signal as in the previous simulation, is added to the feedback signal before being amplified and delayed in the forward path. The resulting signal is sent to the loudspeaker, after which it is filtered in the feedback path to yield the feedback signal. As in the AEC simulations with near-end speech, we should also take into account the source signal characteristics in this scenario. We again apply the LMR-PEM-AFROW algorithm with stochastic gradient update, featuring an AR(12) linear prediction of the source signal. In this simulation, the feedback path is equal to the RIR used in the first simulation. The forward path delay is set equal to the PEM-AFROW linear prediction window length  $L = 160$  samples, as suggested in [2], and the forward path gain is set to  $K = -19$  dB, resulting in a closed-loop gain margin of 3 dB. These settings lead to  $\text{SNR}_{\text{mic}} = -11$  dB.

The LMR-PEM-AFROW algorithm is compared for different choices of the regularization matrix with the (unregularized) standard NLMS algorithm in Fig. 5.7. It is clear that the LMR-PEM-AFROW algorithm is better suited to the closed-loop estimation problem than the NLMS algorithm. We further note that initially, the LMR-PEM-AFROW  $\hat{\mathbf{R}}_{\mathbf{f},3}$  and LMR-PEM-AFROW  $\hat{\mathbf{R}}_{\mathbf{f},\text{true}}$  algorithms converge considerably faster than the LMR-PEM-AFROW  $\alpha\mathbf{I}$  algorithm, yet after some time the misadjustment of these three algorithms evolves to approximately the same level.

## 5.7 Conclusion

In this paper, we have developed a framework for understanding and optimizing the technique of regularization in adaptive filtering algorithms, with application to room acoustic signal enhancement. The traditional Tikhonov regularized and Levenberg-Marquardt regularized RLS algorithms, as well as the regularized APA and NLMS algorithms fit into this framework as special cases, but are found to be suboptimal in many scenarios. An MSE optimal approach towards regularization has been applied to RIR identification, in which prior knowledge on the unknown RIR and on the source signal may be incorporated in a more general regularization matrix, and in an appropriate prefiltering and weighting procedure. This approach results in a new family of optimally regularized adaptive filtering algorithms, which are not only linked to the existing regularized algorithms, but also to other adaptive filtering concepts such as leakage and proportionate adaptation. From simulation results, it is clear that

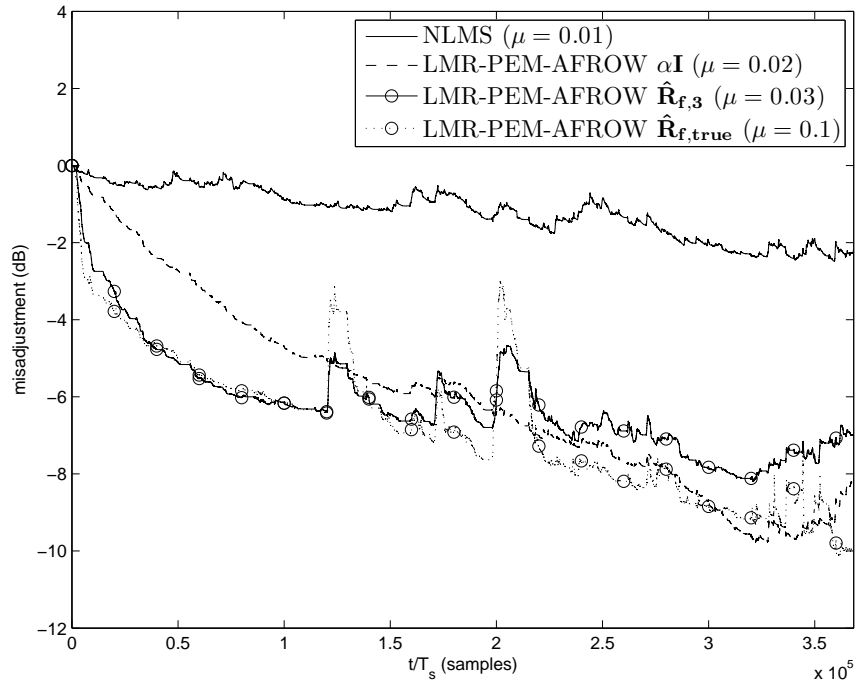


Figure 5.7: Convergence curves of regularized PEM-AFROW algorithms for an AFC application in a stationary environment.

exploiting prior knowledge may considerably increase the adaptive filter convergence speed. In addition, using a method for constructing a regularization matrix based on a 3-parameter RIR model, a significant increase in robustness with respect to RIR changes can be achieved.

## Bibliography

- [1] J. Benesty, T. Gänslér, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*. Berlin: Springer-Verlag, 2001.
- [2] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, "Acoustic feedback suppression for long acoustic paths using a nonstationary source model," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.
- [3] S. J. Elliot and P. A. Nelson, "Active noise control," *IEEE Signal Process. Mag.*, vol. 10, no. 4, pp. 12–35, Oct. 1993.
- [4] R. D. Poltmann, "Stochastic gradient algorithm for system identification using adaptive FIR-filters with too low number of coefficients," *IEEE Trans. Circuits Syst.*, vol. 35, no. 2, pp. 247–250, Feb. 1988.
- [5] G. Rombouts, T. van Waterschoot, K. Struyve, P. Verhoeve, and M. Moonen, "Identification of undermodelled room impulse responses," in *Proc. 2005 Int. Workshop Acoustic Echo Noise Control (IWAENC '05)*, Eindhoven, The Netherlands, Sep. 2005, pp. 153–156.
- [6] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
- [7] A. Tikhonov and V. Arsenin, *Solutions of Ill-Posed Problems*. New York: Wiley, 1977.
- [8] A. Neumaier, "Solving ill-conditioned and singular linear systems: A tutorial on regularization," *Siam Review*, vol. 40, pp. 636–666, 1998.
- [9] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1996.
- [10] E. Horita, K. Sumiya, H. Urakami, and S. Mitsuishi, "A leaky RLS algorithm: Its optimality and implementation," *IEEE Trans. Signal Process.*, vol. 52, no. 10, pp. 2924–2932, Oct. 2004.
- [11] K. Mayyas and T. Aboulnasr, "Leaky LMS algorithm: MSE analysis for gaussian data," *IEEE Trans. Signal Process.*, vol. 45, no. 4, pp. 927–934, Apr. 1997.

- [12] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. Cambridge, Massachusetts: MIT Press, 1986.
- [13] K. Levenberg, “A method for the solution of certain nonlinear problems in least squares,” *Quart. Appl. Math.*, vol. 2, pp. 164–168, 1944.
- [14] D. W. Marquardt, “An algorithm for least-squares estimation of non-linear parameters,” *SIAM J. Appl. Math.*, vol. 11, pp. 431–441, 1963.
- [15] S. L. Gay, “Fast projection algorithms with application to voice echo cancellation,” Ph.D. dissertation, Rutgers, The State University of New Jersey, New Brunswick, New Jersey, USA, Oct. 1994.
- [16] B. Baykal and A. G. Constantinides, “Underdetermined-order recursive least-squares adaptive filtering: The concept and algorithms,” *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 346–362, Feb. 1997.
- [17] M. Bodson, “An adaptive algorithm with information-dependent data forgetting,” in *Proc. 1995 American Control Conf. (ACC '95)*, vol. 5, Seattle, Washington, USA, Jun. 1995, pp. 3485–3489.
- [18] S. L. Gay, “Dynamically regularized fast RLS with application to echo cancellation,” in *Proc. 1996 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '96)*, vol. 2, Atlanta, Georgia, USA, May 1996, pp. 957–960.
- [19] D. R. Morgan and S. G. Kratzer, “On a class of computationally efficient, rapidly converging, generalized NLMS algorithms,” *IEEE Signal Process. Lett.*, vol. 3, no. 8, pp. 245–247, Aug. 1996.
- [20] T. Gänsler, S. L. Gay, M. M. Sondhi, and J. Benesty, “Double-talk robust fast converging algorithms for network echo cancellation,” in *Proc. 1999 IEEE Workshop Appl. Signal Process. Audio Acoust. (WASPAA '99)*, New Paltz, NY, USA, Oct. 1999, pp. 1–4.
- [21] T. Gänsler, J. Benesty, S. L. Gay, and M. M. Sondhi, “A robust proportionate affine projection algorithm for network echo cancellation,” in *Proc. 2000 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '00)*, vol. 2, Istanbul, Turkey, Jun. 2000, pp. 793–796.
- [22] T. Y. Al-Naffouri and A. H. Sayed, “An adaptive filter robust to data uncertainties,” in *Proc. Allerton Conf. Commun., Control, Computing*, vol. 2, Allerton, IL, USA, Oct. 2000, pp. 1175–1183.
- [23] Z. Chen, S. Haykin, and S. L. Gay, “Proportionate adaptation: New paradigms in adaptive filtering,” in *Advances in LMS filters*, S. Haykin and B. Widrow, Eds. New York: Wiley, 2003, ch. 8.
- [24] S. L. Gay and S. Tavathia, “The fast affine projection algorithm,” in *Proc. 1995 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '95)*, vol. 5, Detroit, Michigan, USA, May 1995, pp. 3023–3026.

- [25] V. Myllylä and G. Schmidt, "Pseudo-optimal regularization for affine projection algorithms," in *Proc. 2002 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '02)*, vol. 2, Orlando, Florida, USA, May 2002, pp. 1917–1920.
- [26] C. Kasparis, R. J. Piechocki, P. N. Fletcher, and A. R. Nix, "A bootstrap multi-user detector for CDMA based on Tikhonov regularization," in *Proc. 2003 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '03)*, vol. 4, Apr. 2003, pp. 69–72.
- [27] Z.-F. Wang and X.-W. Yu, "The minimum mean square error linear estimator and ridge regression," in *Proc. 2002 Int. Conf. Machine Learning Cybern. (ICMLC '02)*, Beijing, China, Nov. 2002, pp. 199–201.
- [28] S. Jo and S. W. Kim, "Regularized affine projection algorithm using constrained least squares of residuals," *IEEE Trans. Signal Process.*, submitted for publication, 2004.
- [29] A. H. Sayed, A. Garulli, and S. Chandrasekaran, "A fast iterative solution for worst-case parameter estimation with bounded model uncertainties," in *Proc. 1997 American Control Conf. (ACC '97)*, vol. 3, Albuquerque, New Mexico, USA, Jun. 1997, pp. 1499–1503.
- [30] D. P. Mandic, "A generalized normalized gradient descent algorithm," *IEEE Signal Process. Lett.*, vol. 11, no. 2, pp. 115–118, Feb. 2004.
- [31] Y.-S. Choi, H.-C. Shin, and W.-J. Song, "Robust regularization for normalized LMS algorithms," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 8, pp. 627–631, Aug. 2006.
- [32] E. Chau, H. Sheikhzadeh, and R. L. Brennan, "Complexity reduction and regularization of a fast affine projection algorithm for oversampled subband adaptive filters," in *Proc. 2004 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '04)*, vol. 5, Montreal, QC, Canada, May 2004, pp. 109–112.
- [33] A. Cichocki and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*. New York: Wiley, 1993.
- [34] J. M. Varah, "A practical examination of some numerical methods for linear discrete ill-posed problems," *Siam Review*, vol. 21, no. 1, pp. 100–111, Jan. 1979.
- [35] P. C. Hansen, "Truncated singular value decomposition solutions to discrete ill-posed problems with ill-determined numerical rank," *SIAM J. Sci. Stat. Comput.*, vol. 11, no. 3, pp. 503–518, May 1990.
- [36] D. MacKay, "Bayesian interpolation," *Neural Computation*, vol. 4, no. 3, pp. 415–447, May 1992.

- [37] D. Foresee and M. Hagan, "Gauss-newton approximation to bayesian learning," in *Proc. 1997 Int. Joint Conf. Neural Networks (IJCNN '97)*, Houston, Texas, USA, Jun. 1997, pp. 1930–1935.
- [38] J. P. Kaipio, V. Kolehmainen, M. Vauhkonen, and E. Somersalo, "Inverse problems with structural prior information," *Inverse Problems*, vol. 15, no. 3, pp. 713–729, Jun. 1999.
- [39] Y. Lin and D. D. Lee, "Bayesian regularization and nonnegative deconvolution for room impulse response estimation," *IEEE Trans. Signal Process.*, vol. 54, no. 3, pp. 839–847, Mar. 2006.
- [40] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943–950, Apr. 1979.
- [41] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Upper Saddle River, New Jersey: Prentice-Hall, 1993.
- [42] T. van Waterschoot and M. Moonen, "Double-talk robust acoustic echo cancellation with continuous near-end activity," in *Proc. 13th European Signal Process. Conf. (EUSIPCO '05)*, Antalya, Turkey, Sep. 2005.
- [43] T. van Waterschoot, G. Rombouts, P. Verhoeve, and M. Moonen, "Double-talk-robust prediction error identification algorithms for acoustic echo cancellation," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 846–858, Mar. 2007.
- [44] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, "Acoustic feedback suppression for long acoustic paths using a nonstationary source model," in *Proc. 13th European Signal Process. Conf. (EUSIPCO '05)*, Antalya, Turkey, Sep. 2005.
- [45] T. van Waterschoot, G. Rombouts, and M. Moonen, "Towards optimal regularization by incorporating prior knowledge in an acoustic echo canceller," in *Proc. 2005 Int. Workshop Acoustic Echo Noise Control (IWAENC '05)*, Eindhoven, The Netherlands, Sep. 2005, pp. 157–160.
- [46] A. Houacine, "Regularized fast recursive least squares algorithms," in *Proc. 1990 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '90)*, Albuquerque, New Mexico, USA, Apr. 1990, pp. 1587–1590.
- [47] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 5, pp. 508–518, Sep. 2000.
- [48] D. G. Luenberger, *Optimization by vector space methods*. New York: Wiley, 1969.



- [49] P. C. Hansen, *Rank-deficient and discrete ill-posed problems: Numerical aspects of linear inversion*. Philadelphia, PA, USA: SIAM, 1998.
- [50] W. C. Sabine, *Collected Papers on Acoustics*. Los Altos, California: Peninsula, 1992.
- [51] J. R. Hopgood, "Acoustic impulse response database," ©CEDAR Audio. [Online]. Available: [http://www-sigproc.eng.cam.ac.uk/oldhomes/jrh1008/public\\_html/Resources/ImpulseResponses/index.html](http://www-sigproc.eng.cam.ac.uk/oldhomes/jrh1008/public_html/Resources/ImpulseResponses/index.html).



## Chapter 6

# Dual Regularization for AEC/AFC

Dually regularized recursive prediction error  
identification for acoustic feedback and echo  
cancellation

Toon van Waterschoot, Geert Rombouts, and Marc Moonen

Published in *Proc. 15th European Signal Process. Conf.*  
(*EUSIPCO '07*), Poznań, Poland, Sept. 2007, pp. 1610–1614.

### Contributions of first author

- co-analysis of scaling ambiguity in RPE algorithm
- co-development of dual regularization concept
- co-development of dually regularized RPE algorithm
- co-development of method for constructing speech model prior knowledge
- co-design of evaluation experiments
- software implementation and computer simulations
- co-interpretation of simulation results
- co-formulation of conclusion
- text redaction and editing

## Abstract

Recursive prediction error (RPE) identification algorithms are attractive alternatives to the traditional least-squares-based adaptive filtering algorithms for, e.g., room impulse response identification, in such applications as acoustic feedback and echo cancellation. It has however been observed that a recently proposed RPE algorithm suffers from numerical problems due to a scaling ambiguity in the calculation of the auxiliary variables. This problem is tackled by regularizing the identification of some of the auxiliary variables, which is called “dual regularization”. This leads to a class of Dually Regularized Recursive Prediction Error (DR-RPE) identification algorithms, with different choices of regularization methods (Tikhonov or Levenberg-Marquardt) and matrices (possibly incorporating prior knowledge). Simulation results confirm that the DR-RPE algorithms do not exhibit numerical problems, and as a consequence produce more accurate estimates of the room impulse response and of the auxiliary variables.

## 6.1 Introduction

Recently, recursive prediction error (RPE) identification algorithms have been proposed as robust and efficient solutions to such problems as adaptive feedback cancellation (AFC) [1], [2] and acoustic echo cancellation (AEC) [3]. In these applications, the cancellation of interfering feedback or echo signals is based on the identification of an unknown room impulse response (RIR), see Fig. 6.1. Since both the AFC and AEC problems can be described by a linear data model,

$$y(t) = F(q, t)u(t) + v(t), \quad (6.1)$$

with the finite-order and possibly time-varying RIR defined as

$$F(q, t) = f_0(t) + f_1(t)q^{-1} + \dots + f_{n_F}(t)q^{-n_F} \quad (6.2)$$

and  $q$  denoting the time shift operator, i.e.,  $q^{-k}u(t) = u(t - k)$ , they have traditionally been solved using least-squares(LS)-based adaptive filtering algorithms such as the recursive least squares (RLS), normalized least mean squares (NLMS), and affine projection algorithm (APA). However, due to the non-whiteness of the near-end signal  $v(t)$ , which is a disturbing signal w.r.t. the RIR identification, the LS-based algorithms are suboptimal and perform poorly, especially in the stochastic gradient (NLMS) case [3]. In the AFC application, the non-whiteness of  $v(t)$  moreover produces a bias in the solution of LS-based identification algorithms, which is due to the correlation between the signals  $u(t)$  and  $v(t)$  in the closed-loop system [1].

This is where recursive prediction error (RPE) identification algorithms outperform the traditional adaptive filtering algorithms. By including a time-varying

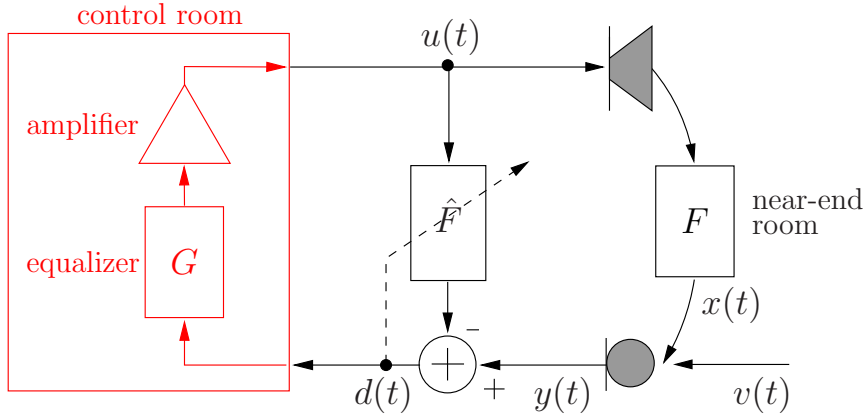


Figure 6.1: The black part of the figure depicts a typical acoustic echo cancellation (AEC) scenario. Taking also into account the red part of the figure, turns the AEC problem into an adaptive feedback cancellation (AFC) problem.

autoregressive (TVAR) model for the near-end signal in the linear data model of (6.1),

$$y(t) = F(q, t)u(t) + \frac{1}{A(q, t)}e(t), \quad (6.3)$$

with

$$A(q, t) = 1 + a_1(t)q^{-1} + \dots + a_{n_A}(t)q^{-n_A}, \quad (6.4)$$

one can obtain a transformed problem [by multiplying both sides of (6.3) with  $A(q, t)$ , and subsequently changing the order of the filters  $A(q, t)$  and  $F(q, t)$  in the cascade  $A(q, t)F(q, t)$ ]

$$A(q, t)y(t) = F(q, t)A(q, t)u(t) + e(t), \quad (6.5)$$

which has a white disturbance signal  $e(t)$  and transformed (i.e., prefiltered) input and output signals  $A(q, t)u(t)$  and  $A(q, t)y(t)$ , respectively. Due to the whiteness of the disturbance, a LS-based algorithm applied to the transformed problem in (6.5) can yield an unbiased and optimal (minimum-variance) RIR estimate [4]. It should however be pointed out that the TVAR polynomial  $A(q, t)$  is also unknown and time-varying. The concept of RPE identification lies in the joint identification of the RIR and of the TVAR near-end signal model by recursively minimizing the sum of squared prediction errors,

$$\min_{\hat{\mathbf{f}}(t), \hat{\mathbf{a}}(t)} \frac{1}{2N} \sum_{k=1}^t \frac{\lambda^{t-k}}{\hat{\sigma}_k^2} \{ \hat{A}(q, t)[y(k) - \hat{F}(q, t)u(k)] \}^2, \quad (6.6)$$

where  $\hat{A}(q, t)$  and  $\hat{F}(q, t)$  represent estimates of  $A(q, t)$  and  $F(q, t)$ , respectively, and weighting is performed using the inverse prediction error power  $\hat{\sigma}_k^{-2}$  to

account for energy variations in the disturbance  $e(t)$  in the transformed data model (6.5), and using  $\lambda^{t-k}$  to discount old data with an exponential forgetting profile. The effective window length  $N = 1/(1 - \lambda)$  is then determined by the forgetting factor  $\lambda$ .

It has been observed that the RPE identification algorithm proposed in [3] performs very well in AFC and AEC computer simulations, yet sometimes runs into numerical problems. In this paper, we will reveal the origin of this numerical shortcoming and propose a theoretically well-founded solution. In Section 6.2, we will show that the RPE algorithm exhibits a so-called scaling ambiguity, in that it may theoretically produce the correct RIR estimate, even when the TVAR coefficients and some of the other auxiliary variables are scaled with an arbitrary scaling factor (that is significantly larger than the inverse of the norm of the estimated TVAR coefficients). Since the auxiliary variables are not of direct interest to the user, this should not be a problem, unless the scaling becomes so large that numerical overflow occurs. In some simulation scenarios such numerical overflow has indeed been observed.

Hence, even if the TVAR coefficients are not of direct interest, we may benefit by improving their identification, since then numerical problems will be avoided and the resulting RIR estimate will have a higher numerical accuracy. In Section 6.3, we will indicate how the accuracy of the estimated TVAR coefficients can be increased using regularization. To distinguish between the regularization of the estimated TVAR coefficients and the regularization of the estimated RIR coefficients, we will use the term “dual regularization” to denote the former, and “primal regularization” for the latter. We will apply both the Tikhonov and Levenberg-Marquardt regularization methods to the dual regularization problem, with a regularization matrix that may incorporate prior knowledge on the true TVAR coefficients [5]. We will see how such prior knowledge can be constructed for near-end speech signals. The performance of the Dually Regularized RPE (DR-RPE) algorithms is then illustrated using results from computer simulations in Section 6.4, and finally Section 6.5 concludes the paper.

## 6.2 Scaling Ambiguity in the RPE Algorithm

For convenience, the RPE algorithm with stochastic gradient RIR weight update as proposed in [3], is reproduced in Table 6.1. The parameter vectors and data matrices are defined as:

$$\boldsymbol{\theta}(t) \triangleq \begin{bmatrix} \mathbf{f}(t) \\ \mathbf{a}(t) \end{bmatrix}_{(n_F+n_A+1) \times 1}, \quad (6.7)$$

$$\mathbf{f}(t) \triangleq [f_0(t) \quad f_1(t) \quad \dots \quad f_{n_F}(t)]_{(n_F+1) \times 1}^T, \quad (6.8)$$

$$\mathbf{a}(t) \triangleq [a_1(t) \quad a_2(t) \quad \dots \quad a_{n_A}(t)]_{n_A \times 1}^T, \quad (6.9)$$

$$\mathbf{y}(t) \triangleq [y(t-1) \quad \dots \quad y(t-n_A)]_{n_A \times 1}^T, \quad (6.10)$$

$$\mathbf{u}(t) \triangleq [u(t) \quad \dots \quad u(t-n_F)]_{(n_F+1) \times 1}^T, \quad (6.11)$$

$$\mathbf{U}(t) \triangleq \begin{bmatrix} u(t-1) & \dots & u(t-n_A) \\ \vdots & \ddots & \vdots \\ u(t-n_F-1) & \dots & u(t-n_F-n_A) \end{bmatrix}_{(n_F+1) \times n_A} \quad (6.12)$$

The TVAR coefficient vector  $\mathbf{a}(t)$  and the prediction error variance  $\sigma_t^2$  are estimated on an exponential data window with a forgetting factor  $\lambda_A$ , chosen such that the effective data window length equals approximately 20 ms (which is the average time interval on which a speech signal can be considered stationary). The stochastic gradient RIR weight update features a step size  $\mu_F$ .

The scaling ambiguity is explained as follows: if the auxiliary variables  $\hat{\mathbf{a}}(t)$ ,  $\hat{\mathbf{a}}(t-1)$ ,  $\hat{\sigma}_t^2$ ,  $\hat{\sigma}_{t-1}^2$ ,  $\boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)]$ ,  $\varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]$ ,  $\mathbf{R}_A(t)$ , and  $\mathbf{R}_A(t-1)$ , are replaced by their scaled counterparts  $K\hat{\mathbf{a}}(t)$ ,  $K\hat{\mathbf{a}}(t-1)$ ,  $K^2\hat{\sigma}_t^2$ ,  $K^2\hat{\sigma}_{t-1}^2$ ,  $K\boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)]$ ,  $K\varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]$ ,  $K^{-2}\mathbf{R}_A(t)$ , and  $K^{-2}\mathbf{R}_A(t-1)$ , respectively, then the RPE algorithm will produce the same solution  $\hat{\mathbf{f}}(t)$  as in the case without scaling, provided that  $K \gg \|\hat{\mathbf{a}}(t-1)\|_2^{-1}$ . For such large scaling factors, the unit coefficient preceding the estimated TVAR coefficients in the calculation of  $\varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]$  and  $\boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)]$  becomes negligible as compared to  $K\hat{\mathbf{a}}(t-1)$ , and as a consequence, the prediction error filter  $\hat{A}(q, t-1)$  approximately has a zero at infinity,

$$\hat{A}(q, t-1) = 1 + K\hat{a}_1(t-1)q^{-1} + \dots + K\hat{a}_{n_A}(t-1)q^{-n_A}, \quad (6.13)$$

$$\approx Kq^{-1}[\hat{a}_1(t-1) + \dots + \hat{a}_{n_A}(t-1)q^{-n_A+1}]. \quad (6.14)$$

In computer simulations using the RPE algorithm, it appears that starting at some iteration, and without any outliers occurring in the data, the aforementioned auxiliary variables undergo an exponentially increasing scaling. This is illustrated in Fig. 6.2, where the norm of the TVAR coefficients  $20 \log_{10} \|\hat{\mathbf{a}}(t)\|_2$  and the prediction error power  $10 \log_{10} \hat{\sigma}_t^2$  are drawn on a dB-scale, as a function of the number of RPE iterations. In some simulations, the numerical divergence of the auxiliary variables is hardly visible in the convergence curves of the RIR estimate until overflow occurs (see, e.g., [3, Fig. 4(c)]), yet in other simulations, it is clear that even before numerical overflow occurs, the accuracy of the RIR estimate is already severely affected by the numerical problems in the auxiliary variable estimation (see Fig. 6.4).



Table 6.1: Stochastic gradient RPE algorithm [3]

---

**Prediction error:**

$$\varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)] = [1 \quad \hat{\mathbf{a}}^T(t-1)] \left( \begin{bmatrix} y(t) \\ \mathbf{y}(t) \end{bmatrix} - \begin{bmatrix} \mathbf{u}^T(t) \\ \mathbf{U}^T(t) \end{bmatrix} \hat{\mathbf{f}}(t-1) \right)$$

**Prediction error variance:**

$$\hat{\sigma}_t^2 = \lambda_A \hat{\sigma}_{t-1}^2 + (1 - \lambda_A) \varepsilon^2[t, \hat{\boldsymbol{\theta}}(t-1)]$$

**Regression vectors:**

$$\begin{aligned} \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] &= [\mathbf{u}(t) \quad \mathbf{U}(t)] \begin{bmatrix} 1 \\ \hat{\mathbf{a}}(t-1) \end{bmatrix} \\ \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] &= \mathbf{U}^T(t) \hat{\mathbf{f}}(t-1) - \mathbf{y}(t) \end{aligned}$$

**TVAR regression vector correlation matrix update:**

$$\mathbf{R}_A(t) = \lambda_A \mathbf{R}_A(t-1) + \frac{1}{\hat{\sigma}_t^2} \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \boldsymbol{\psi}_A^T[t, \hat{\mathbf{f}}(t-1)]$$

**TVAR and RIR weight updates:**

$$\begin{aligned} \hat{\mathbf{a}}(t) &= \hat{\mathbf{a}}(t-1) + \frac{1}{\hat{\sigma}_t^2} \mathbf{R}_A^{-1}(t) \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)] \\ \hat{\mathbf{f}}(t) &= \hat{\mathbf{f}}(t-1) + \mu_F \frac{\boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] \varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]}{\boldsymbol{\psi}_F^T[t, \hat{\mathbf{a}}(t-1)] \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] + \hat{\sigma}_t^2} \end{aligned}$$


---

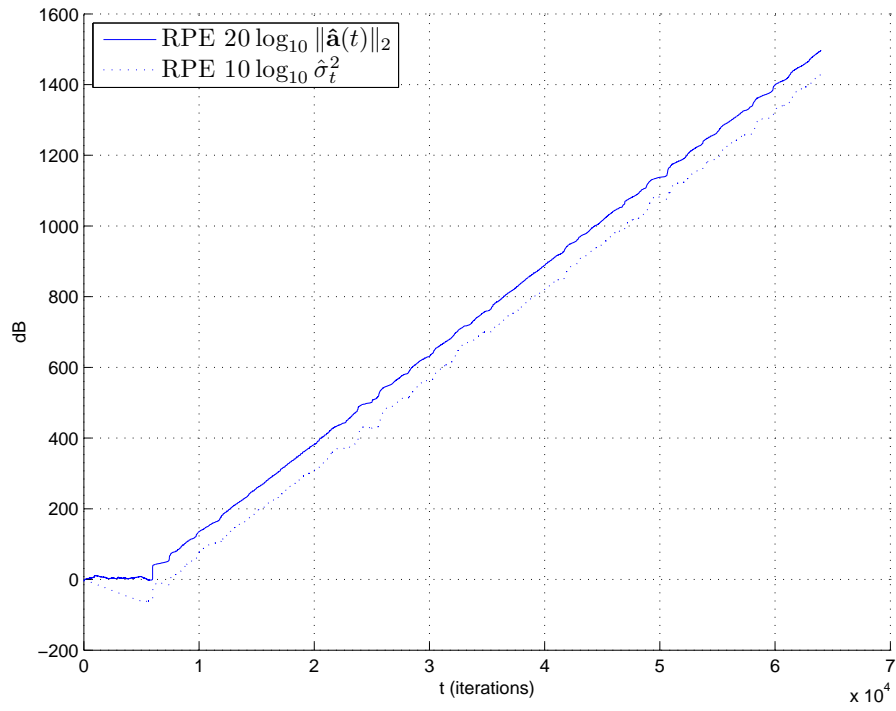


Figure 6.2: RPE scaling problem: from  $t \approx 6000$  iterations, the auxiliary variables start growing at an exponential rate.

## 6.3 Dual Regularization

A simple and intuitive solution to prevent the TVAR coefficients from diverging as in the above example, is to include a minimum-norm constraint into the prediction error criterion (6.6), i.e.,

$$\min_{\hat{\mathbf{f}}(t), \hat{\mathbf{a}}(t)} \frac{1}{2N} \sum_{k=1}^t \frac{\lambda^{t-k}}{\hat{\sigma}_k^2} \{ \hat{A}(q, t)[y(k) - \hat{F}(q, t)u(k)] \}^2 + \beta \|\hat{\mathbf{a}}(t)\|_2^2,$$

which corresponds to performing a Tikhonov regularization with regularization parameter  $\beta$ . We will denote the TVAR coefficients' regularization as "dual regularization", for pointing out the difference with the ("primal") RIR coefficients' regularization. A more thorough regularization approach would be to perform a "generalized" type of regularization, by considering the true TVAR coefficient vectors  $\mathbf{a}(k)$ ,  $k = 1, \dots, t$  as different realizations of the same stochastic variable  $\mathbf{a}$  on which some prior knowledge may be available through its mean and covariance matrix, i.e.,

$$\begin{cases} E\{\mathbf{a}\} \triangleq \mathbf{a}_0, \\ \text{cov}\{\mathbf{a}\} = E\{(\mathbf{a} - \mathbf{a}_0)(\mathbf{a} - \mathbf{a}_0)^T\} \triangleq \mathbf{R}_a. \end{cases} \quad (6.15)$$

$$(6.16)$$

An optimal approach to the dual regularization problem can then be suggested in accordance with the optimal primal regularization approach in [5], and consists in minimizing

$$\begin{aligned} \min_{\hat{\mathbf{f}}(t), \hat{\mathbf{a}}(t)} \frac{1}{2N} \left\{ \sum_{k=1}^t \frac{\lambda^{t-k}}{\hat{\sigma}_k^2} \{ \hat{A}(q, t)[y(k) - \hat{F}(q, t)u(k)] \}^2 \right. \\ \left. + [\hat{\mathbf{a}}(t) - \mathbf{a}_0]^T \mathbf{R}_a^{-1} [\hat{\mathbf{a}}(t) - \mathbf{a}_0] \right\}. \end{aligned} \quad (6.17)$$

Finally, also adding a primal regularization term results in

$$\begin{aligned} \min_{\hat{\mathbf{f}}(t), \hat{\mathbf{a}}(t)} \frac{1}{2N} \left\{ \sum_{k=1}^t \frac{\lambda^{t-k}}{\hat{\sigma}_k^2} \{ \hat{A}(q, t)[y(k) - \hat{F}(q, t)u(k)] \}^2 \right. \\ \left. + [\hat{\mathbf{f}}(t) - \mathbf{f}_0]^T \mathbf{R}_f^{-1} [\hat{\mathbf{f}}(t) - \mathbf{f}_0] + [\hat{\mathbf{a}}(t) - \mathbf{a}_0]^T \mathbf{R}_a^{-1} [\hat{\mathbf{a}}(t) - \mathbf{a}_0] \right\}, \end{aligned} \quad (6.18)$$

where the true RIR  $\mathbf{f}(t)$  is considered to be a realization of the stochastic variable  $\mathbf{f}$ , with mean and covariance matrix

$$\begin{cases} E\{\mathbf{f}\} \triangleq \mathbf{f}_0, \\ \text{cov}\{\mathbf{f}\} = E\{(\mathbf{f} - \mathbf{f}_0)(\mathbf{f} - \mathbf{f}_0)^T\} \triangleq \mathbf{R}_f. \end{cases} \quad (6.19)$$

$$(6.20)$$

Minimizing (6.18) w.r.t.  $\hat{\mathbf{f}}(t)$  and  $\hat{\mathbf{a}}(t)$  results in the so-called Dually Regularized Recursive Prediction Error (DR-RPE) identification algorithm, shown

in Table 6.2. It should be noted that the user has the choice of using either the Tikhonov regularized TVAR weight update (corresponding to  $\mathbf{a}_0 = \mathbf{0}$ ), or the Levenberg-Marquardt regularized TVAR weight update (corresponding to  $\mathbf{a}_0 = \hat{\mathbf{a}}(t-1)$ ). As for the stochastic gradient RIR weight update, the Levenberg-Marquardt regularization (corresponding to  $\mathbf{f}_0 = \hat{\mathbf{f}}(t-1)$ ) is the only relevant choice [5].

## 6.4 Simulation Results

Computer simulations were carried out to compare the performance of the unregularized RPE algorithm with the performance of the proposed DR-RPE algorithm for different choices of the regularization matrix  $\mathbf{R}_a$ , and for both Tikhonov (TR) and Levenberg-Marquardt (LMR) regularization methods. As the focus of this paper is on the dual regularization, the primal regularization matrix is set to  $\mathbf{R}_f = \mathbf{I}$ . A first choice for the dual regularization matrix is  $\mathbf{R}_a = \mathbf{I}$ , which yields a traditional (identity matrix) Tikhonov or Levenberg-Marquardt regularization. A second choice for  $\mathbf{R}_a$  allows for incorporating prior knowledge on speech signal characteristics. To this end, we have identified a TVAR model on 310929 different 20 ms speech frames read from the TIMIT database [6], and ensemble-averaged the TVAR coefficient vector outer product to obtain

$$\hat{\mathbf{R}}_{a,s} = E\{\hat{\mathbf{a}}_i \hat{\mathbf{a}}_i^T\}, \quad i = 1, \dots, 310929. \quad (6.21)$$

A 3D surface plot of the resulting regularization matrix  $\hat{\mathbf{R}}_{a,s}$  is shown in Fig. 6.3.

The algorithms were tested in an adaptive feedback cancellation (AFC) scenario at a sampling frequency of 16 kHz. Referring to Fig. 6.1, the amplifier is modeled as a broadband gain  $L$  cascaded with a saturation function, and the equalizer is a pure time delay, i.e.,  $G(q) = q^{-d}$ , needed for identifiability of  $\mathbf{f}(t)$  and  $\mathbf{a}(t)$  [1]. The closed-loop system is kept at an average gain margin of 3 dB, by setting  $L = -9$  dB and  $d = 320$  samples. The near-end signal is a 4 s male speech signal (not from the TIMIT database), while the true RIR is measured in a typical recording studio and is of known order  $n_F = 2000$ . The TVAR model order is chosen as  $n_A = 24$ , the forgetting factor for estimating the TVAR coefficients and the prediction error variance is set to  $\lambda_A = 0.9971$ , and the step size is optimized for each of the algorithms, resulting in  $\mu_F = 10^{-3}$  for the unregularized RPE algorithm and  $\mu_F = 10^{-2}$  for the DR-RPE algorithms.

The convergence of the RIR estimate is depicted in Fig. 6.4 by plotting the so-called RIR misadjustment, defined as

$$\text{RIR misadjustment (dB)} = 20 \log_{10} \frac{\|\hat{\mathbf{f}}(t) - \mathbf{f}\|_2}{\|\mathbf{f}\|_2}. \quad (6.22)$$

Table 6.2: Dually Regularized RPE (DR-RPE) algorithm

---

**Prediction error:**

$$\varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)] = [1 \quad \hat{\mathbf{a}}^T(t-1)] \left( \begin{bmatrix} y(t) \\ \mathbf{y}(t) \end{bmatrix} - \begin{bmatrix} \mathbf{u}^T(t) \\ \mathbf{U}^T(t) \end{bmatrix} \hat{\mathbf{f}}(t-1) \right)$$

**Prediction error variance:**

$$\hat{\sigma}_t^2 = \lambda_A \hat{\sigma}_{t-1}^2 + (1 - \lambda_A) \varepsilon^2[t, \hat{\boldsymbol{\theta}}(t-1)]$$

**Regression vectors:**

$$\begin{aligned} \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] &= [\mathbf{u}(t) \quad \mathbf{U}(t)] \begin{bmatrix} 1 \\ \hat{\mathbf{a}}(t-1) \end{bmatrix} \\ \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] &= \mathbf{U}^T(t) \hat{\mathbf{f}}(t-1) - \mathbf{y}(t) \end{aligned}$$

**Regularized TVAR regression vector correlation matrix update [dual]:**

$$\begin{aligned} \mathbf{R}_A(t) &= \lambda_A \mathbf{R}_A(t-1) + \frac{1}{\hat{\sigma}_t^2} \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \boldsymbol{\psi}_A^T[t, \hat{\mathbf{f}}(t-1)] \\ &\quad + (1 - \lambda_A) \mathbf{R}_a^{-1} \end{aligned}$$

**Tikhonov Regularized TVAR weight update [dual]:**

$$\begin{aligned} \hat{\mathbf{a}}(t) &= \hat{\mathbf{a}}(t-1) + \mathbf{R}_A^{-1}(t) \\ &\times \left\{ \frac{1}{\hat{\sigma}_t^2} \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)] - (1 - \lambda_A) \mathbf{R}_a^{-1} \hat{\mathbf{a}}(t-1) \right\} \end{aligned}$$

**Levenberg-Marquardt Regularized TVAR weight update [dual]:**

$$\hat{\mathbf{a}}(t) = \hat{\mathbf{a}}(t-1) + \frac{1}{\hat{\sigma}_t^2} \mathbf{R}_A^{-1}(t) \boldsymbol{\psi}_A[t, \hat{\mathbf{f}}(t-1)] \varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]$$

**Levenberg-Marquardt Regularized RIR weight update [primal]:**

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mu_F \frac{\mathbf{R}_f \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] \varepsilon[t, \hat{\boldsymbol{\theta}}(t-1)]}{\boldsymbol{\psi}_F^T[t, \hat{\mathbf{a}}(t-1)] \mathbf{R}_f \boldsymbol{\psi}_F[t, \hat{\mathbf{a}}(t-1)] + \hat{\sigma}_t^2}$$


---

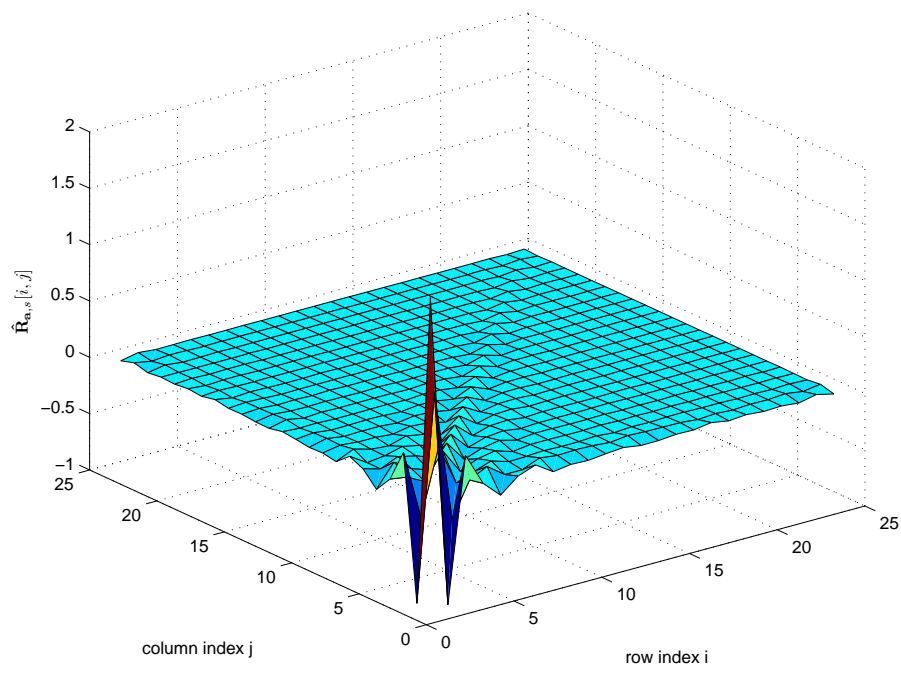


Figure 6.3: 3D surface plot of regularization matrix  $\hat{\mathbf{R}}_{\mathbf{a},\mathbf{s}}$ .

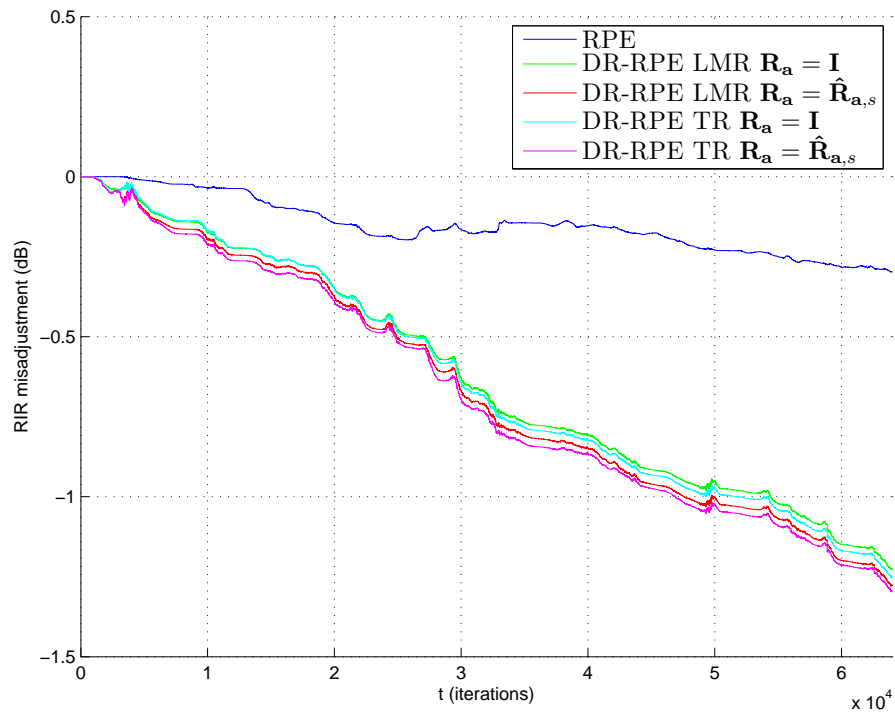


Figure 6.4: RIR misadjustment convergence curves for speech simulation.

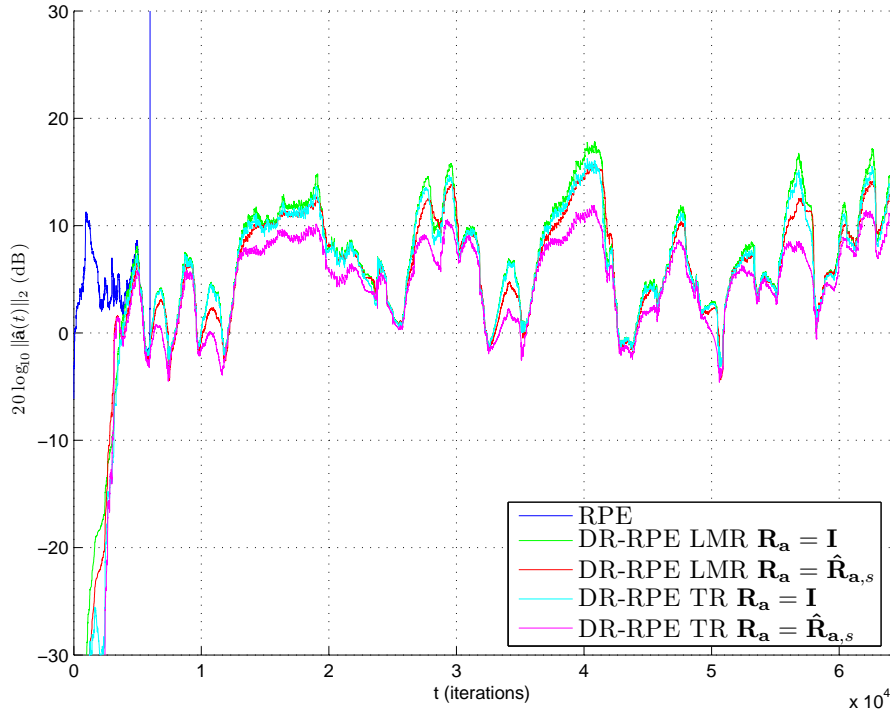


Figure 6.5: Estimated TVAR coefficient norm for speech simulation.

It can be seen that the accuracy of the RIR estimate is indeed improved by adding a dual regularization to the RPE algorithm. The impact of the different choices for the regularization matrices and methods on the RIR estimate convergence seems to be negligible. The norm  $20 \log_{10} \|\hat{\mathbf{a}}(t)\|_2$  of the estimated TVAR coefficients and the estimated prediction error power  $10 \log_{10} \hat{\sigma}_t^2$  are plotted in Figs. 6.5 and 6.6, respectively. It is clear that these auxiliary variables do not diverge in the DR-RPE algorithms, as they do in the unregularized RPE algorithm. Moreover, it can be seen from Fig. 6.5 that the norm of the estimated TVAR coefficients is somewhat smaller in case  $\mathbf{R}_a = \hat{\mathbf{R}}_{a,s}$ , especially using the Tikhonov regularization method.

In a final simulation, the accuracy of the estimated TVAR coefficients is compared for the different DR-RPE regularization matrices and methods. To this end, the near-end signal is a 60 s synthetic TVAR sequence, generated by passing a Gaussian white noise signal through a time-varying all-pole filter of order  $n_A = 24$ . The coefficients of this filter change every 20 ms and were computed by linear prediction of 20 ms true speech frames. The quality of the estimated TVAR coefficients can be compared by evaluating the time-averaged TVAR



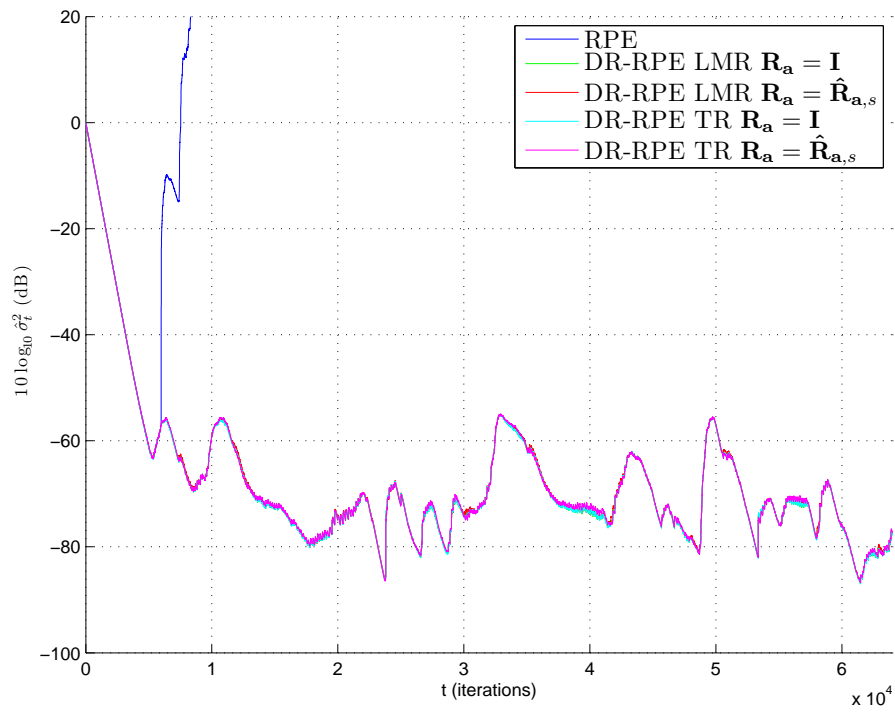


Figure 6.6: Estimated prediction error power for speech simulation.

Table 6.3: Time-averaged TVAR misadjustment for synthetic speech simulation of DR-RPE algorithms

Regularization method	$\mathbf{R}_a$	TVAR misadjustment
Levenberg-Marquardt	$\mathbf{I}$	-1.5322 dB
Levenberg-Marquardt	$\hat{\mathbf{R}}_{a,s}$	-1.9629 dB
Tikhonov	$\mathbf{I}$	-1.9814 dB
Tikhonov	$\hat{\mathbf{R}}_{a,s}$	-3.5738 dB

misadjustment, defined as

$$\text{TVAR misadjustment (dB)} = 20 \log_{10} \left( \frac{1}{M} \sum_{k=1}^M \frac{\|\hat{\mathbf{a}}(k) - \mathbf{a}(k)\|_2}{\|\mathbf{a}(k)\|_2} \right), \quad (6.23)$$

with  $M = 960 \cdot 10^3$  and  $\mathbf{a}(k)$  the corresponding all-pole filter coefficients used to generate the synthetic TVAR sequence. The time-averaged TVAR misadjustment is compared for the four different DR-RPE algorithms in Table 6.3. It can be seen that it is advantageous to use the Tikhonov regularization instead of the Levenberg-Marquardt regularization, and to use a regularization matrix incorporating prior knowledge, such as the proposed matrix  $\hat{\mathbf{R}}_{a,s}$ .

## 6.5 Conclusion

In this paper, we have highlighted a numerical problem in a recently proposed RPE identification algorithm, which seems to be due to an inherent scaling ambiguity. We have proposed to solve the problem using a so-called “dual regularization” approach, which may be combined with a primal regularization method to obtain a class of Dually Regularized Recursive Prediction Error (DR-RPE) identification algorithms. User’s choices within the DR-RPE class include the regularization method used (Tikhonov or Levenberg-Marquardt) and the type of regularization matrix used. A first observation from computer simulations is that using the dual regularization, the aforementioned numerical problem does not occur anymore. As a consequence, a more accurate RIR estimate may be obtained. A second observation is that using a regularization matrix that incorporates prior knowledge on the TVAR coefficients, results in a more accurate TVAR coefficient estimate. Moreover, the Tikhonov regularization method is preferred over the Levenberg-Marquardt method.

## Bibliography

- [1] A. Spriet, I. Proudler, M. Moonen, and J. Wouters, “Adaptive feedback cancellation in hearing aids with linear prediction of the desired signal,” *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3749–3763, Oct. 2005.
- [2] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, “Acoustic feedback suppression for long acoustic paths using a nonstationary source model,” *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.
- [3] T. van Waterschoot, G. Rombouts, P. Verhoeve, and M. Moonen, “Double-talk-robust prediction error identification algorithms for acoustic echo cancellation,” *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 846–858, Mar. 2007.
- [4] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. Cambridge, Massachusetts: MIT Press, 1986.
- [5] T. van Waterschoot, G. Rombouts, and M. Moonen, “Optimally regularized adaptive filtering algorithms for room acoustic signal enhancement,” *Signal Processing*, vol. 88, no. 3, pp. 594–611, Mar. 2008.
- [6] J. S. Garofolo *et al.*, “TIMIT acoustic-phonetic continuous speech corpus,” CD-ROM, 1993, linguistic Data Consortium, Philadelphia, PA, USA.



## Part III

# Acoustic Feedback Control



## Chapter 7

# Biquadratic Notch Filter Design

A pole-zero placement technique for designing second-order IIR parametric equalizer filters

Toon van Waterschoot and Marc Moonen

Published in *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 8, Nov. 2007, pp. 2561–2565.

©2007 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

### Contributions of first author

- literature study
- co-development of set of design equations
- co-development of notch and resonance filter design procedure
- co-design of design examples
- software implementation and computer simulations
- co-interpretation of simulation results
- co-formulation of conclusion
- text redaction and editing



## Abstract

A new procedure is presented for designing second-order parametric equalizer filters. In contrast to the traditional approach, in which the design is based on a bilinear transform of an analog filter, the presented procedure allows for designing the filter directly in the digital domain. A rather intuitive technique known as pole-zero placement, is treated here in a quantitative way. It is shown that by making some meaningful approximations, a set of relatively simple design equations can be obtained. Design examples of both notch and resonance filters are included to illustrate the performance of the proposed method, and to compare with state-of-the-art solutions.

## 7.1 Introduction

Many signal processing applications involve enhancing or attenuating only a small portion of a signal's frequency spectrum, while leaving the remainder of the spectrum unaffected. This effect is obtained by using bandpass or bandstop filters, that have a frequency response which is characterized by a gain increase or decrease around a specified center frequency  $f_c$ . In digital audio equalization, any desired frequency response may be realized by cascading such bandpass/-stop filters with different center frequencies, which are then often referred to as parametric equalizer filters or presence filters. Such a cascade may moreover include high-pass and low-pass filters, also known as shelving filters, which are in fact special (degenerate) cases of the bandpass/-stop filter, having one real pole-zero pair instead of two complex conjugate pole-zero pairs.

Apart from the center frequency, a parametric equalizer filter is also characterized by its bandwidth (which we will define later on). Filters having a small bandwidth  $B$  relative to their center frequency (i.e., having a high  $Q$ -factor, defined as  $Q = f_c/B$ ), are better known as notch and resonance (or peaking) filters. Notch filters appear in numerous applications where a (nearly) sinusoidal interference has to be cancelled from a broadband signal, e.g., suppressing a 50/60 Hz AC interference in a low-voltage measurement signal, or cancelling acoustic feedback oscillations in audio amplification systems. Resonance filters are typically used to recover sinusoids buried in noise (so-called line enhancement), e.g., in communications and sonar applications.

It is well-known that a bandpass or bandstop characteristic around a specified center frequency can be realized efficiently using a second-order infinite impulse response (IIR) filter [1]-[11], also known as a biquadratic filter, with transfer function

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \quad (7.1)$$

Traditionally, the five filter coefficients are calculated so as to satisfy a set of five design equations [7]:

$$|H(e^{j0})| = G_0 \quad (\text{gain at DC}) \quad (7.2a)$$

$$|H(e^{j\pi})| = G_\pi \quad (\text{gain at Nyquist frequency}) \quad (7.2b)$$

$$\left( \frac{\partial}{\partial \omega} |H(e^{j\omega})| \right) \Big|_{\omega=\omega_c} = 0 \quad (\text{center frequency}) \quad (7.2c)$$

$$|H(e^{j\omega_c})| = G_c \quad (\text{gain at resonance}) \quad (7.2d)$$

$$|H(e^{j(\omega_c \pm B/2)})| = G_B \quad (\text{bandwidth}) \quad (7.2e)$$

where  $\omega = 2\pi f$  represents radial frequency. These design equations incorporate the following design variables: (radial) center frequency  $\omega_c$ , (radial) bandwidth  $B$ , gain at band edges  $G_B$ , gain at resonance  $G_c$ , gain at DC  $G_0$ , and gain at Nyquist frequency  $G_\pi$ . Typically, the DC and Nyquist gain are chosen to be equal (except in [10], with the aim of digitally emulating an analog equalizer) and are set to 0 dB, i.e.,  $G_0 = G_\pi = 1$ , which facilitates the cascading of several parametric equalizer filters. Unfortunately, there is little agreement in the literature on how to appropriately define bandwidth [7]. We will adopt Moorer's bandwidth definition [2], which is found to be mathematically the most consistent one [7]. If the resonance gain relative to the gain at DC, i.e.,  $G_c/G_0$  (called the "boost" in a resonance filter and the "cut" in a notch filter), exceeds 6 dB in absolute value, then the band edges are defined as the frequencies at which the gain is 3dB below/above the peak/notch. For a boost/cut less than 6dB in absolute value, the band edges are found at the so-called midpoint gain, which is defined as the geometric mean  $\sqrt{G_c/G_0}$ , i.e.,  $1/2(G_{c,\text{dB}} - G_{0,\text{dB}})$  dB below/above the peak/notch.

Nearly all existing design procedures start from the design of an analog parametric equalizer filter, followed by a bilinear transform that maps the analog frequency axis  $[0, \infty)$  onto the digital frequency axis  $[0, f_s/2]$ , with  $f_s$  the sampling frequency [1],[3]-[7],[10]. To this end, the digital design variables  $\omega_c$  and  $B$  should be "prewarped" to analog variables, which can however not be done in an exact way for the bandwidth [5],[7]. As an alternative, the parametric equalizer filter can also be designed directly in the digital domain. A first approach was suggested in [2] and starts by designing a digital parametric equalizer filter with center frequency  $\omega_c = \pi/2$  (which is the only digital center frequency that allows for a truly symmetric frequency response, leading to  $b_1 = a_1 = 0$ ). The filter centered at  $\pi/2$  is then transformed to an arbitrarily centered filter using an appropriate bilinear transform. A second approach is more intuitive and based on a technique known as pole-zero placement. It was shown earlier how a resonance filter with a specified center frequency and bandwidth can be designed in the  $z$ -plane by placing two complex conjugate poles (inside, but close to the unit circle) on the radial lines from the origin to  $e^{\pm j\omega_c}$ , and two zeros at the origin [8, Ch. 6],[9, Ch. 6],[11, Ch. 11]. For this design procedure, several approximate relations between the bandwidth and the so-called pole ra-

dus (i.e., the distance from the origin to the pole) have been suggested [8, Ch. 6],[11, Ch. 11]. Also, it was noted in [9, Ch. 6] that by moving the zeros from the origin towards the poles, and even beyond, the resonance characteristic can be converted into a notch characteristic.

The aim of this correspondence is to present a more quantitative treatment of the pole-zero placement approach. We will derive exact relations between the pole and zero positions on the one hand, and the design variables on the other hand. These relations will however appear to be impractical for implementation, and hence we will suggest some useful approximations. The pole-zero placement design procedure is outlined in Section 7.2, where we will make a distinction between notch filter and resonance filter design. Some design examples are given in Section 7.3, and finally, Section 7.4 concludes the correspondence.

## 7.2 Design Procedure

The pole-zero placement design procedure is based on a radial representation of the biquadratic filter transfer function, in which poles and zeros are constrained to lie on the radial lines from the origin to  $e^{\pm j\omega_c}$  [9, Ch. 6],[11, Ch. 11],[12], i.e.,

$$H(z) = K \frac{(1 - r_z e^{j\omega_c} z^{-1})(1 - r_z e^{-j\omega_c} z^{-1})}{(1 - r_p e^{j\omega_c} z^{-1})(1 - r_p e^{-j\omega_c} z^{-1})}, \quad (7.3)$$

which can equivalently be written in direct form as

$$H(z) = K \frac{1 - 2r_z \cos \omega_c z^{-1} + r_z^2 z^{-2}}{1 - 2r_p \cos \omega_c z^{-1} + r_p^2 z^{-2}}. \quad (7.4)$$

The zero radius  $r_z \in [0, 1]$  is defined as the distance from the origin to each of the complex conjugate zeros, and likewise the pole radius  $r_p \in [0, 1)$  is defined as the distance from the origin to the poles. A broadband gain factor  $K$  is also included.

In contrast to the original biquadratic filter transfer function in (7.1), the radial representation contains only four distinct parameters and, as a consequence, only four design equations can be fulfilled. Therefore, the design equations (7.2a) and (7.2b) may be replaced by only one equation that specifies the filter response at an arbitrary frequency. For convenience, however, we will just omit equation (7.2b) and retain equation (7.2a). Furthermore, since the center frequency can be specified directly in the radial representation in (7.3) or (7.4), design equation (7.2c) can also be omitted. Note that equation (7.2c) would however only be exactly fulfilled for a desired center frequency at  $\omega_c = \{0, \pi/2, \pi\}$ , since the maximum/minimum in the frequency response of the constrained resonance/notch filter in (7.3) does not generally appear at the

resonance frequency, which is due to the influence of the complex conjugate pole-zero pair [11, Ch. 11]. This effect decreases as the poles approach the unit circle.

We will now derive exact and approximate relations between the remaining three filter parameters  $\{r_z, r_p, K\}$  and the design variables  $\{\omega_c, B, G_0, G_c\}$ , by evaluating the remaining design equations (7.2a), (7.2d), and (7.2e). We will generally assume that the poles are close to the unit circle, i.e.,  $0 \ll r_p < 1$ , resulting in narrowband parametric equalizer filters. To obtain a broader bandpass/bandstop characteristic, a cascade of shelving filters should be used instead. Below, the design of notch and resonance filters is treated separately, since different assumptions can be made in either case.

### 7.2.1 Notch Filters

To obtain a notch characteristic, the zeros should be placed between the poles and the unit circle, i.e.,  $0 \ll r_p < r_z \leq 1$ . The gain at resonance can be calculated by using the so-called graphical method for evaluating a frequency response [8, Ch. 4],[11, Ch. 9], see Fig. 7.1,

$$|H(e^{j\omega_c})| = K \frac{|Z_1 R| |Z_2 R|}{|P_1 R| |P_2 R|}. \quad (7.5)$$

Note that the notation  $|\cdot|$  is used in two different ways:  $|XY|$  denotes the distance between two points  $X$  and  $Y$  in the complex plane, whereas  $|H(e^{j\omega})|$  stands for the frequency magnitude response of the biquadratic filter.

Since the poles are close to the unit circle, and hence close to the zeros (in practice even much closer than in Fig. 7.1), we can assume the distances from the complex conjugate pole and zero to  $e^{j\omega_c}$  to be nearly equal, i.e.  $|Z_2 R| \approx |P_2 R|$ , as compared to the distances  $|Z_1 R|$  and  $|P_1 R|$ . With this approximation, (7.5) can be rewritten as

$$|H(e^{j\omega_c})| = K \frac{(1 - r_z)}{(1 - r_p)}. \quad (7.6)$$

The gain at DC can be evaluated by using the same approach, i.e.,

$$|H(e^{j0})| = K \frac{|Z_1 A| |Z_2 A|}{|P_1 A| |P_2 A|}. \quad (7.7)$$

Because the distance between the pole at  $P_1$  and the zero at  $Z_1$  is much smaller than the distance of  $P_1$  and  $Z_1$  to the point  $e^{j0}$ , we may state that  $|Z_1 A| \approx |P_1 A|$  (this may appear to be a bad approximation for  $\omega_c \rightarrow 0$ , however in that case a highpass characteristic will be obtained, unless  $r_p, r_z \rightarrow 1$ , which still

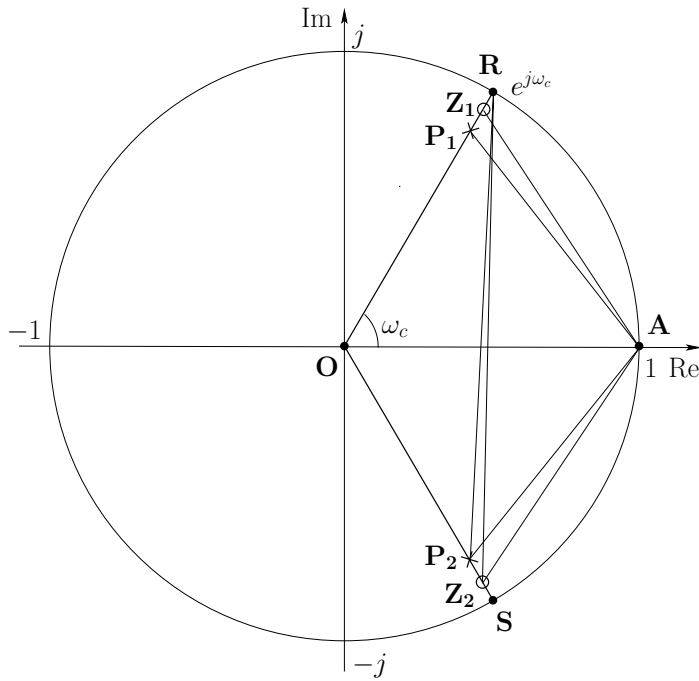


Figure 7.1: Pole-zero plot for notch filter (x = poles, o = zeros).

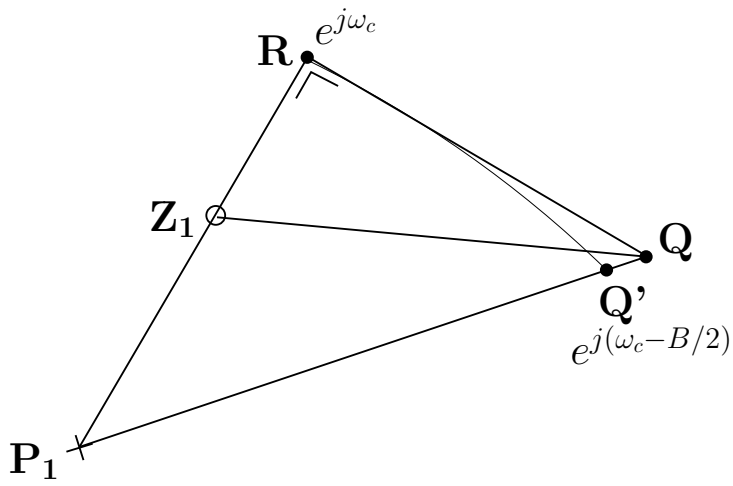


Figure 7.2: Bandwidth calculation for notch filter.

leads to  $|Z_1A| \approx |P_1A|$ ). The same reasoning results in  $|Z_2A| \approx |P_2A|$  for the complex conjugate pole-zero pair, such that (7.7) can be rewritten as

$$|H(e^{j0})| = K. \quad (7.8)$$

Dividing (7.6) and (7.8), and applying design equations (7.2a) and (7.2d), leads to

$$\frac{G_c}{G_0} = \frac{(1 - r_z)}{(1 - r_p)}. \quad (7.9)$$

We will now identify the lower band edge  $e^{j(\omega_c - B/2)}$  and calculate its distance from the poles and zeros. Since we are dealing with narrowband filters, the distance between  $e^{j\omega_c}$  and  $e^{j(\omega_c - B/2)}$  will be relatively small, hence the influence of the complex conjugate pole-zero pair on the change in frequency response from  $e^{j\omega_c}$  to  $e^{j(\omega_c - B/2)}$  will be negligible. It then suffices to consider the region depicted in Fig. 7.2 for calculating the lower band edge, and

$$|H(e^{j(\omega_c - B/2)})| \approx K \frac{|Z_1Q'|}{|P_1Q'|} \approx K \frac{|Z_1Q|}{|P_1Q|}, \quad (7.10)$$

where the second approximation is justified by the fact that all points shown in Fig. 7.2 are close to the unit circle, so that the unit circle may be locally approximated by its tangent in  $e^{j\omega_c}$ , and hence  $Q \approx Q'$ . Furthermore, the arc length between  $R$  and  $Q'$  is nearly equal to  $|RQ|$ . Since the frequency response is approximately symmetric around the center frequency if the poles are close to the unit circle [11, Ch. 11], the bandwidth can be calculated as  $B \approx 2|RQ|$ .

Recall our definition of bandwidth,

$$\frac{|H(e^{j\omega_c})|}{|H(e^{j(\omega_c - B/2)})|} = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } \frac{G_c}{G_0} \leq \frac{1}{2}, \\ \sqrt{\frac{G_c}{G_0}} & \text{if } \frac{G_c}{G_0} \geq \frac{1}{2}. \end{cases} \quad (7.11)$$

Moreover, combining (7.6) and (7.10) yields

$$\frac{|H(e^{j\omega_c})|}{|H(e^{j(\omega_c - B/2)})|} = \frac{(1 - r_z)|P_1Q|}{(1 - r_p)|Z_1Q|}. \quad (7.12)$$

With the Pythagorean theorem applied to the triangles  $P_1RQ$  and  $Z_1RQ$ , we obtain two more equations,

$$|P_1Q|^2 = |P_1R|^2 + |RQ|^2 = (1 - r_p)^2 + |RQ|^2, \quad (7.13)$$

$$|Z_1Q|^2 = |Z_1R|^2 + |RQ|^2 = (1 - r_z)^2 + |RQ|^2, \quad (7.14)$$

which can be solved for  $|RQ|$ , together with (7.11) and (7.12), resulting in

$$B \approx 2|RQ| = \begin{cases} 2\sqrt{\frac{(1-r_p)^2(1-r_z)^2}{(1-r_p)^2 - 2(1-r_z)^2}} & \text{if } \frac{G_c}{G_0} \leq \frac{1}{2}, \\ 2\sqrt{\frac{(\frac{G_c}{G_0}-1)(1-r_p)^2(1-r_z)^2}{(1-r_z)^2 - \frac{G_c}{G_0}(1-r_p)^2}} & \text{if } \frac{G_c}{G_0} \geq \frac{1}{2}. \end{cases} \quad (7.15)$$

Dividing both the numerator and denominator in the above square root expressions by either  $(1 - r_p)^2$  or  $(1 - r_z)^2$ , and subsequently using the result in (7.9), yields the following expressions for the zero and pole radius:

$$r_z = \begin{cases} 1 - \frac{B}{2} \sqrt{1 - 2\frac{G_c^2}{G_0^2}} & \text{if } \frac{G_c}{G_0} \leq \frac{1}{2}, \\ 1 - \frac{B}{2} \sqrt{\frac{G_c}{G_0}} & \text{if } \frac{G_c}{G_0} \geq \frac{1}{2}, \end{cases} \quad (7.16)$$

$$r_p = \begin{cases} 1 - \frac{B}{2} \sqrt{\frac{G_0^2}{G_c^2} - 2} & \text{if } \frac{G_c}{G_0} \leq \frac{1}{2}, \\ 1 - \frac{B}{2} \sqrt{\frac{G_0}{G_c}} & \text{if } \frac{G_c}{G_0} \geq \frac{1}{2}. \end{cases} \quad (7.17)$$

These expressions, together with  $K = G_0$ , constitute the notch filter design procedure. Note that, while the expressions in (7.16) and (7.17) may theoretically yield negative results, this should never occur for the narrowband filters we are dealing with (i.e., for sufficiently small  $B$ ).

### 7.2.2 Resonance Filters

A similar procedure can be used to design resonance filters, but then the zeros should be placed between the origin and the poles, i.e.,  $0 \leq r_z < r_p < 1$ , see Fig. 7.3. Note that in this case, no assumption can be made about the position of the zeros, i.e., depending on the specified gain at resonance, the zeros may be close to the origin, or close to the poles (near the unit circle), or anywhere in between. We will however assume that the distance between  $Z_1$  and  $P_1$  is significantly larger than the distance from  $P_1$  to the unit circle, and the same for  $Z_2$  and  $P_2$ , such that the zeros can be neglected when calculating the bandwidth. As in the notch filter case, we will also assume that the influence of the complex conjugate pole-zero pair on the change in frequency response from  $e^{j\omega_c}$  to  $e^{j(\omega_c - B/2)}$  is negligible. Hence the bandwidth, which in this case is defined by

$$\frac{|H(e^{j\omega_c})|}{|H(e^{j(\omega_c - B/2)})|} = \begin{cases} \sqrt{2} & \text{if } \frac{G_c}{G_0} \geq 2, \\ \sqrt{\frac{G_c}{G_0}} & \text{if } \frac{G_c}{G_0} \leq 2, \end{cases} \quad (7.18)$$

only depends on the pole position, see Fig. 7.4, i.e.,

$$\frac{|H(e^{j\omega_c})|}{|H(e^{j(\omega_c - B/2)})|} \approx \frac{|P_1 Q|}{(1 - r_p)}. \quad (7.19)$$

This expression, together with the Pythagorean theorem in (7.13), results in

$$B \approx 2|RQ| = \begin{cases} 2(1 - r_p) & \text{if } \frac{G_c}{G_0} \geq 2, \\ 2(1 - r_p) \sqrt{\frac{G_c}{G_0} - 1} & \text{if } \frac{G_c}{G_0} \leq 2, \end{cases} \quad (7.20)$$

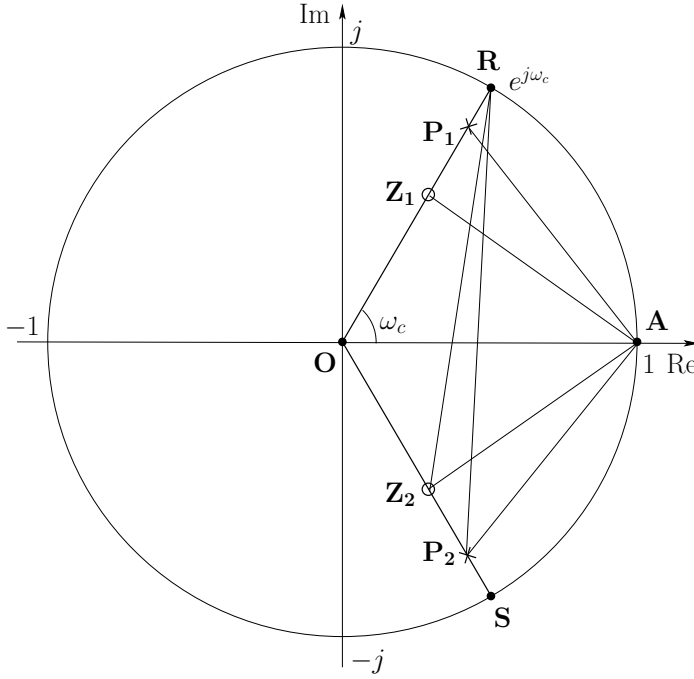


Figure 7.3: Pole-zero plot for resonance filter.

so that the pole radius can be calculated as

$$r_p = \begin{cases} 1 - \frac{B}{2} & \text{if } \frac{G_c}{G_0} \geq 2, \\ 1 - \frac{B}{2\sqrt{\frac{G_c}{G_0} - 1}} & \text{if } \frac{G_c}{G_0} \leq 2. \end{cases} \quad (7.21)$$

The upper part of (7.21) was also derived in [8, Ch. 6].

Determining the zero radius for the resonance filter is somewhat more difficult, as no assumption can be made about the position of the zeros. Graphically evaluating the gain at resonance and at DC yields

$$|H(e^{j\omega_c})| = K \frac{(1 - r_z)|Z_2 R|}{(1 - r_p)|P_2 R|}, \quad (7.22)$$

$$|H(e^{j0})| = K \frac{|Z_1 A||Z_2 A|}{|P_1 A||P_2 A|} = K \frac{|Z_1 A|^2}{|P_1 A|^2}. \quad (7.23)$$

Eliminating  $K$  from (7.22) and (7.23), and taking into account design equations (7.2a) and (7.2d), leads to

$$\frac{(1 - r_z)|Z_2 R||P_1 A|^2}{(1 - r_p)|P_2 R||Z_1 A|^2} = \frac{G_c}{G_0}, \quad (7.24)$$



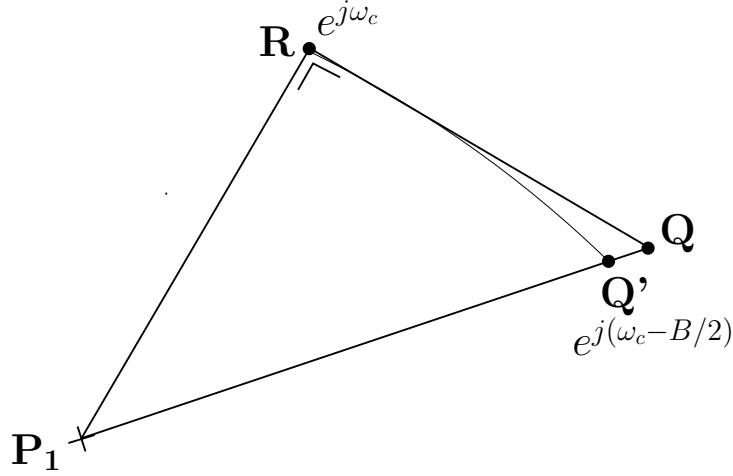


Figure 7.4: Bandwidth calculation for resonance filter.

where we can make the following approximations. Since the poles are close to the unit circle,  $|P_2R| \approx |SR| = 2 \sin \omega_c$ , and  $|P_1A| \approx |RA| = 2(1 - \cos \omega_c)$ . Furthermore, the fraction

$$\frac{|Z_1A|^2}{|Z_2R|} = \frac{r_z^2 - 2r_z \cos \omega_c + 1}{\sqrt{r_z^2 - 2r_z \cos 2\omega_c + 1}}, \quad (7.25)$$

may be approximated by a first-order polynomial in  $r_z$ . As  $r_z$  varies between 0 and 1, the fraction in (7.25) takes on a value between 1 and  $(1 - \cos \omega_c) / \sin \omega_c$ , which leads to the following approximation:

$$\frac{|Z_1A|^2}{|Z_2R|} \approx (1 - r_z) + \left( \frac{1 - \cos \omega_c}{\sin \omega_c} \right) r_z. \quad (7.26)$$

Substituting the above approximations in (7.24), results in a first-order equation in  $r_z$ ,

$$\frac{(1 - r_z)}{(1 - r_p)} = \frac{G_c}{G_0} \left( r_z + (1 - r_z) \frac{\sin \omega_c}{1 - \cos \omega_c} \right), \quad (7.27)$$

which can readily be solved by using (7.21), i.e.,

$$r_z = \begin{cases} \frac{1 - \Gamma}{1 - \Gamma + \frac{B}{2} \frac{G_c}{G_0}} & \text{if } \frac{G_c}{G_0} \geq 2, \\ \frac{1 - \Gamma}{1 - \Gamma + \frac{B}{2\sqrt{G_c/G_0 - 1}} \frac{G_c}{G_0}} & \text{if } \frac{G_c}{G_0} \leq 2, \end{cases} \quad (7.28)$$

with

$$\Gamma = \begin{cases} \frac{B}{2} \frac{G_c}{G_0} \frac{\sin \omega_c}{1 - \cos \omega_c} & \text{if } \frac{G_c}{G_0} \geq 2, \\ \frac{B}{2\sqrt{G_c/G_0 - 1}} \frac{G_c}{G_0} \frac{\sin \omega_c}{1 - \cos \omega_c} & \text{if } \frac{G_c}{G_0} \leq 2. \end{cases} \quad (7.29)$$

It should be noted that, once the bandwidth is specified, the boost  $G_c/G_0$  cannot take on an arbitrarily large value. The maximum boost, given  $B$ , is obtained when the zeros are placed at the origin, and can be calculated from (7.29) with  $\Gamma = 1$ .

Finally, the broadband gain factor  $K$  can be calculated by evaluating (7.4) at  $e^{j0}$ , i.e.,

$$K = G_0 \frac{1 - 2r_p \cos \omega_c + r_p^2}{1 - 2r_z \cos \omega_c + r_z^2}. \quad (7.30)$$

### 7.3 Design Examples

We have applied the above design procedure to four design examples. In the first example, the bandwidth of a resonance filter with  $\omega_c = 2$  rad ( $0.64\pi$  rad),  $G_0 = 1$  (0 dB), and  $G_c = 4$  (12 dB) takes on different values  $B = \{0.01, 0.1, 0.2\}$  rad ( $\{0.0032, 0.032, 0.064\}\pi$  rad), see Fig. 7.5. The second example illustrates the effect of varying the boost of a resonance filter as  $G_c/G_0 = \{2, 3, 4\}$  ( $\{6, 9.5, 12\}$  dB), with  $\omega_c = 1$  rad ( $0.32\pi$  rad),  $B = 0.05$  rad ( $0.016\pi$  rad), and  $G_0 = 1$  (0 dB), see Fig. 7.6. In the third example, in Fig. 7.7, the center frequency of a notch filter with  $B = 0.1$  rad ( $0.032\pi$  rad),  $G_0 = 1$  (0 dB), and  $G_c = 0.3$  ( $-10.5$  dB) is set to  $\omega_c = \{1, 1.5, 2\}$  rad ( $\{0.32, 0.48, 0.64\}\pi$  rad). The fourth example is included to illustrate the capability to fade from a resonance filter to a notch filter using the proposed design procedure. In Fig. 7.8, the boost/cut of a parametric equalizer filter with  $\omega_c = 1.5$  rad ( $0.48\pi$  rad),  $B = 0.1$  rad ( $0.032\pi$  rad), and  $G_0 = 1$  (0 dB) is varied as  $G_c/G_0 = \{1.1220, 1.0593, 1.0233, 1, 0.9772, 0.9441, 0.8913\}$  ( $\{1, 0.5, 0.2, 0, -0.2, -0.5, -1\}$  dB). Note that a flat magnitude response ( $G_c/G_0 = 1$ ) is obtained by setting  $r_z = r_p$  in (7.3), resulting in pole-zero cancellation, and  $K = G_0$ .

Finally, we compare the new procedure to three state-of-the-art techniques for designing parametric equalizer filters. A cascade of a notch filter with  $\omega_c = 1$  rad ( $0.32\pi$  rad),  $B = 0.05$  rad ( $0.016\pi$  rad),  $G_0 = 1$  (0 dB), and  $G_c = 0.7$  ( $-3$  dB), and a resonance filter with  $\omega_c = 2$  rad ( $0.64\pi$  rad),  $B = 0.1$  rad ( $0.032\pi$  rad),  $G_0 = 1$  (0 dB), and  $G_c = 4$  (12 dB), was designed using the proposed pole-zero placement technique, as well as using the techniques described by Regalia and Mitra [4], Bristow-Johnson [7], and Moorer [2]. From Fig. 7.9 it can be seen that the four methods perform equally well on the notch filter, whereas some discrepancies in the 3 dB-bandwidth appear for the resonance filter. More specifically, in Bristow-Johnson's method, the resonance filter bandwidth is somewhat underestimated.

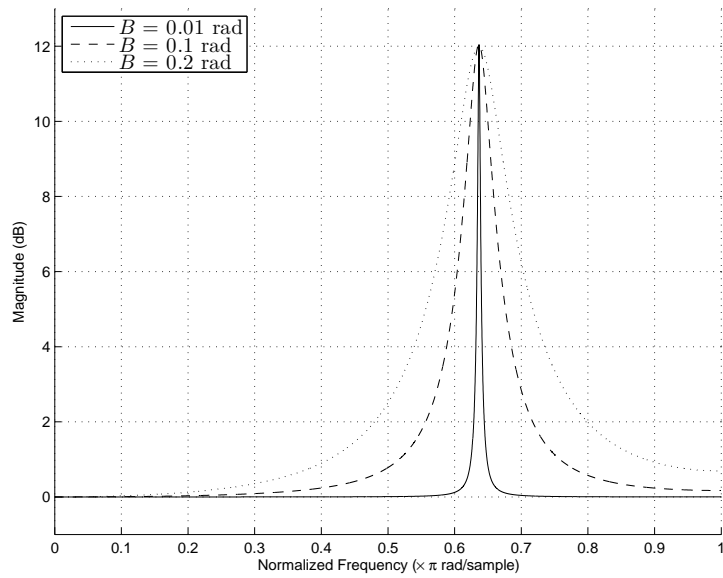


Figure 7.5: Design example 1: resonance filter with varying bandwidth.

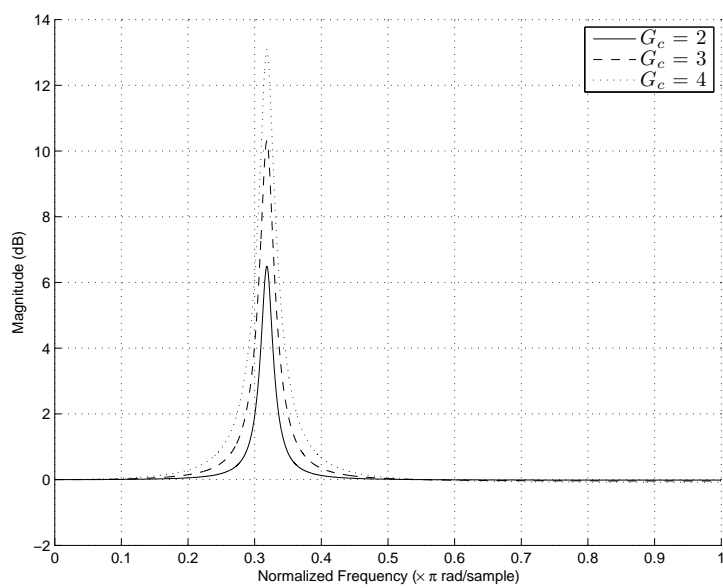


Figure 7.6: Design example 2: resonance filter with varying boost.

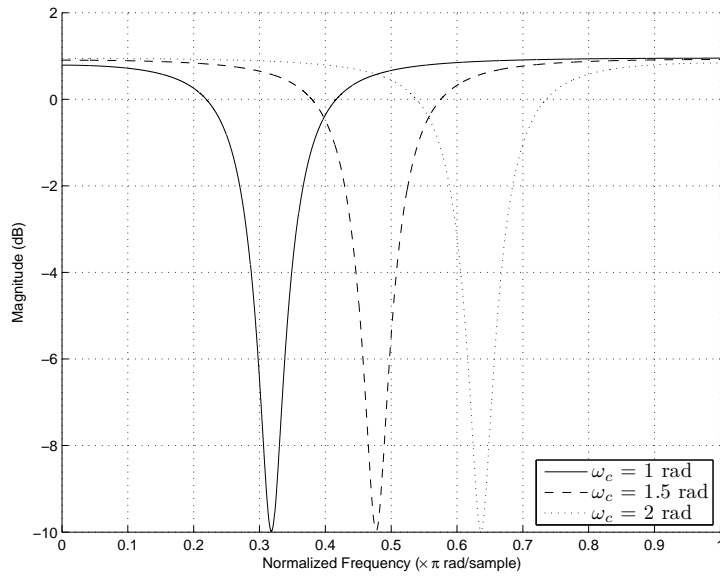


Figure 7.7: Design example 3: notch filter with varying center frequency.

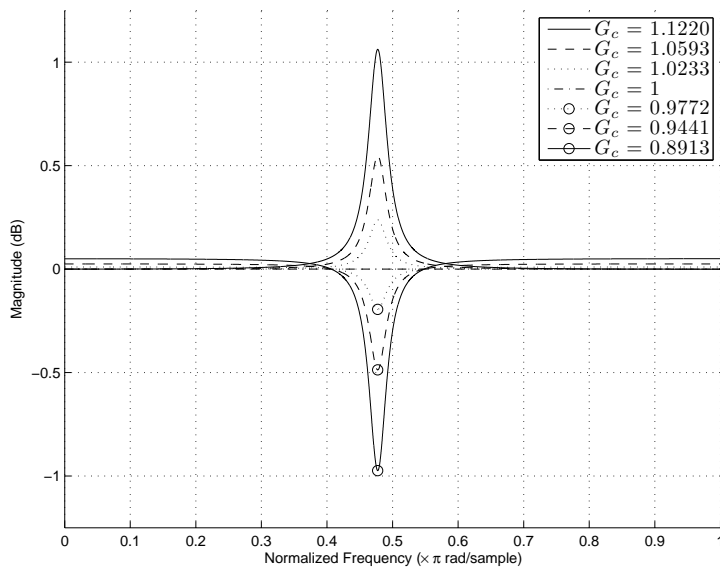


Figure 7.8: Design example 4: parametric equalizer filter fading from boost to cut.

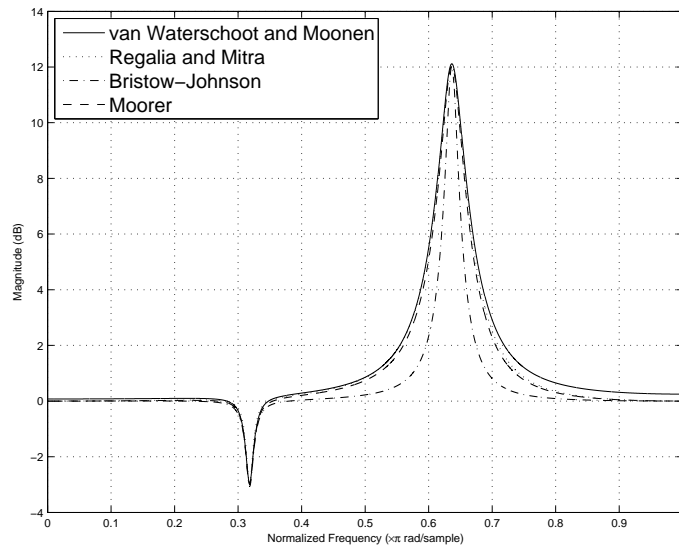


Figure 7.9: Comparison with three state-of-the-art design procedures.

## 7.4 Conclusion

In this correspondence, we have proposed a new pole-zero placement technique for designing second-order parametric equalizer filters. Depending on whether a notch or resonance filter characteristic is desired, certain approximations on the pole and zero positions can be made which simplify the filter design equations. Expressions were given for calculating the filter's pole radius, zero radius, and broadband gain factor, which, together with the specified center frequency, completely determine the filter transfer function. The effect of varying the design variables, i.e., center frequency, bandwidth, and boost/cut, was illustrated in four design examples. The proposed design procedure was also compared to three state-of-the-art algorithms and was found to be equally accurate, with the additional advantage of providing an intuitive design approach executed entirely in the digital domain.

## Bibliography

- [1] K. Hirano, S. Nishimura, and S. K. Mitra, "Design of digital notch filters," *IEEE Trans. Commun.*, vol. COM-22, no. 7, pp. 964–970, Jul. 1974.
- [2] J. A. Moorer, "The manifold joys of conformal mapping: Applications of

- digital filtering in the studio,” *J. Audio Eng. Soc.*, vol. 31, no. 11, pp. 826–841, Nov. 1983.
- [3] S. A. White, “Design of a digital biquadratic peaking or notch filter for digital audio equalization,” *J. Audio Eng. Soc.*, vol. 34, no. 6, pp. 479–483, Jun. 1986.
- [4] P. A. Regalia and S. K. Mitra, “Tunable digital frequency response equalization filters,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 1, pp. 118–120, Jan. 1987.
- [5] D. J. Shpak, “Analytical design of biquadratic filter sections for parametric filters,” *J. Audio Eng. Soc.*, vol. 40, no. 11, pp. 876–885, Nov. 1992.
- [6] D. C. Massie, “An engineering study of the four-multiply normalized ladder filter,” *J. Audio Eng. Soc.*, vol. 41, no. 7/8, pp. 564–582, July/August 1993.
- [7] R. Bristow-Johnson, “The equivalence of various methods of computing biquad coefficients for audio parametric equalizers,” in *Preprints AES 97th Convention*, San Francisco, CA, USA, Nov. 1994, AES Preprint 3906.
- [8] P. A. Lynn and W. Fuerst, *Introductory Digital Signal Processing with Computer Applications*. New York: Wiley, 1994.
- [9] S. J. Orfanidis, *Introduction to Signal Processing*. Upper Saddle River, New Jersey: Prentice-Hall, 1996.
- [10] ———, “Digital parametric equalizer design with prescribed Nyquist-frequency gain,” *J. Audio Eng. Soc.*, vol. 45, no. 6, pp. 444–455, Jun. 1997.
- [11] J. O. Smith, *Introduction to Digital Filters*. Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, August 2006 Edition. [Online]. Available: <http://ccrma.stanford.edu/~jos/filters05/>
- [12] A. Nehorai, “A minimal parameter adaptive notch filter with constrained poles and zeros,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 4, pp. 983–996, Aug. 1985.

## Chapter 8

# Efficient Parametric Frequency Estimation

Constrained pole-zero linear prediction: an  
efficient and near-optimal method for multi-tone  
frequency estimation

Toon van Waterschoot and Marc Moonen

Published in *Proc. 16th European Signal Process. Conf.*  
(*EUSIPCO '08*), Lausanne, Switzerland, Aug. 2008.

### Contributions of first author

- literature study
- co-development of CPZLP concept
- co-development of decoupled optimization approach
- co-development of line search numerical optimization algorithm
- co-analysis of gradient and Hessian calculation
- co-analysis of computational complexity
- co-design of evaluation experiments
- software implementation and computer simulations
- co-interpretation of simulation results
- co-formulation of conclusion
- text redaction and editing



## Abstract

Constrained pole-zero linear prediction (CPZLP) is proposed as a new method for parametric frequency estimation of multiple real sinusoids buried in noise. The method is based on a signal model that consists of a cascade of second-order constrained pole-zero models, thereby exploiting the linear prediction property of sinusoidal signals. The signal model is parametrized directly with the unknown frequencies, which are then estimated using a numerical optimization approach. By independently optimizing each second-order stage in the cascade model, a computationally efficient algorithm is obtained with a complexity that is linear in both the data record length and the number of sinusoids. The linear complexity allows for using relatively long data records, leading to high accuracy even at low signal-to-noise ratios (SNR). Simulation results confirm that the CPZLP algorithm nearly achieves the Cramér-Rao lower bound for SNR as low as 5 dB.

## 8.1 Introduction

The problem of estimating the frequencies of a sum of sinusoidal signals (multi-tone signals) buried in additive noise has received a lot of attention during the past decades. Solutions to this problem have been applied in many different areas, such as audio and speech processing, radar signal processing, telecommunications, etc. The existing methods are usually categorized as being either nonparametric or parametric. Nonparametric frequency estimation is directly based on Fourier transform theory, hence the signal is processed in a frame-based manner. The main drawback of nonparametric methods is their limited frequency resolution for finite frame length. Parametric methods, on the other hand, can achieve a higher resolution but require the postulation of a generating signal model. We refer to [1] for a recent overview of parametric frequency estimation methods.

A particular class of parametric methods exploits the linear prediction (LP) property of sinusoidal signals. It is well known that a sum of  $P$  sinusoids can be described exactly using an all-pole model of order  $2P$ , with mirror symmetric LP coefficients [1]. However, it has been shown that the all-pole model is not exact when noise is added, and in this case a pole-zero model of order  $2P$  should be used [2]. Still, by constraining the poles and zeros to lie on common radial lines in the  $z$ -plane, the number of unknown parameters in the pole-zero model can be limited to  $P$  and the LP parameters can be uniquely related to the unknown frequencies [3]. The constrained pole-zero model has been widely applied in adaptive notch filtering (ANF), see, e.g., [3]-[5]. The ANF algorithms are however very sensitive to the choice of the initial conditions and the exponential forgetting factor, and in nonstationary scenarios memory

resetting of the ANF is regularly required to enable sufficiently fast tracking.

In this paper, we describe a new parametric frequency estimation method that is based on the constrained pole-zero model proposed in [3], realized using a cascade of second-order sections with a direct frequency parametrization [4],[5]. The proposed method is referred to as constrained pole-zero linear prediction (CPZLP) and, in contrast to the ANF approach, the signal is processed in a frame-based manner. In the CPZLP method, the minimization of a least-squares (LS) objective for multi-tone frequency estimation is decoupled into a set of single-tone subproblems that can be solved consecutively by exploiting the cascade structure of the signal model. Each subproblem can be viewed as a single-variable unconstrained nonlinear optimization problem, and is solved iteratively using a numerical line search method [6, Ch. 3]. Because of the decoupling, the CPZLP method achieves a computational complexity that depends linearly on the frame length and on the number of second-order sections, even when Hessian information is used in the optimization method. As a consequence, relatively long frame lengths can be used to increase the noise robustness.

The paper is organized as follows. In Section 8.2, we introduce the constrained pole-zero signal model and derive the CPZLP algorithm by considering the decoupled optimization of the LS objective. We describe a line search method with three possible ways of calculating the search direction (steepest descent, Gauss-Newton, and quasi-Newton), and provide details on the gradient and Hessian calculation. Section 8.3 deals with the computational complexity of the CPZLP algorithm, and Section 8.4 contains Monte Carlo simulation results that illustrate the CPZLP performance in terms of frequency variance as compared to the Cramér-Rao lower bound (CRLB). Finally, Section 8.5 concludes the paper.

## 8.2 Constrained Pole-Zero Linear Prediction

### 8.2.1 Signal Model

The observed signal  $y(t)$  is assumed to consist of a sum of real sinusoids and additive noise,

$$y(t) = \sum_{n=1}^P A_n \cos(\omega_n t + \phi_n) + r(t), \quad t = 1, \dots, N \quad (8.1)$$

with  $A_n$  the amplitude,  $\omega_n \in [0, \pi]$  the radial frequency, and  $\phi_n \in [0, 2\pi)$  the phase of the  $n$ th sinusoid. While most parametric frequency estimators rely on the hypothesis that the noise  $r(t)$  is white [1], we do not make explicit assumptions about the noise. The CPZLP algorithm has been tested both with white

noise, see Section 8.4, and with colored noise, see [7]. Throughout this paper, it is assumed that the number of sinusoids  $P$  in the observed signal is known a priori, which is a common assumption in parametric frequency estimation [1], [3]-[5]. We should note that the CPZLP approach can be elegantly extended to achieve joint order and frequency estimation [7]. The frequencies  $\omega_n$  are assumed to be stationary over the observed data frame  $t \in [1, N]$ . Frequency tracking in a nonstationary environment is possible if CPZLP is preceded by an adaptive segmentation.

It is well known that a sum of  $P$  sinusoids can be described exactly as an autoregressive process of order  $2P$ . A sum of  $P$  sinusoids in additive white noise, similarly, can be modeled as an autoregressive moving average process of order  $2P$ , having zeros that coincide with the poles [2]. This observation has led to the constrained pole-zero model for signals that consist of sinusoidal or narrowband components in noise [3], which is given as

$$y(t) = \frac{\sum_{i=0}^{2P} \rho^i a_i z^{-i}}{\sum_{i=0}^{2P} a_i z^{-i}} e(t). \quad (8.2)$$

This model has been widely used for deriving ANF algorithms. The LP coefficients  $a_i$  in (8.2) are mirror symmetric because the poles are constrained to lie on the unit circle, i.e.,  $a_i = a_{2P-i}$  and also  $a_0 = a_{2P} = 1$ . Moreover, the zeros are constrained to lie on the same radial lines as the poles, at a constant distance  $\rho$  from the origin ( $0 \leq \rho < 1$ ). Note that  $\rho$  is defined as the pole radius or pole contraction factor since in the prediction error filter, corresponding to the inverse signal model,  $\rho$  appears in the denominator. Throughout this paper,  $\rho$  is assumed to be a fixed parameter, the choice of which is however of great importance to the frequency estimation performance. The residual signal  $e(t)$  is usually assumed to be an uncorrelated sequence, such as a white noise sequence or a Dirac impulse. With the aim of achieving direct frequency estimation, the model in (8.2) is sometimes rewritten using a second-order sections cascade structure [4], [5],

$$y(t) = \left( \prod_{n=1}^P \frac{1 - 2\rho \cos \theta_n z^{-1} + \rho^2 z^{-2}}{1 - 2 \cos \theta_n z^{-1} + z^{-2}} \right) e(t) \quad (8.3)$$

with  $\theta_n \in [0, \pi]$  denoting the angles of the pole-zero pairs in the upper half of the  $z$ -plane.

### 8.2.2 Decoupled Optimization

The goal of the proposed frequency estimation method is to have the angles  $\theta_n$  in the constrained pole-zero signal model (8.3) converge to the frequencies  $\omega_n$

of the observed signal in (8.1). To this end, a LS objective is defined as follows:

$$V(\boldsymbol{\theta}) = \frac{1}{N} \sum_{t=1}^N e^2(t, \boldsymbol{\theta}) \quad (8.4)$$

with, from (8.3),

$$e(t, \boldsymbol{\theta}) = \left( \prod_{n=1}^P \frac{1 - 2 \cos \theta_n z^{-1} + z^{-2}}{1 - 2\rho \cos \theta_n z^{-1} + \rho^2 z^{-2}} \right) y(t) \quad (8.5)$$

and

$$\boldsymbol{\theta} = [\theta_1 \quad \dots \quad \theta_P]^T. \quad (8.6)$$

Instead of directly minimizing  $V(\boldsymbol{\theta})$  w.r.t. the parameter vector  $\boldsymbol{\theta}$ , we divide the minimization problem into  $P$  subproblems. Let the intermediate residual signal  $e_n(t, \boldsymbol{\theta}_n)$  be defined as the output of the  $n$ th section of the prediction error filter cascade,

$$e_n(t, \boldsymbol{\theta}_n) = \left( \prod_{l=1}^n \frac{1 - 2 \cos \theta_l z^{-1} + z^{-2}}{1 - 2\rho \cos \theta_l z^{-1} + \rho^2 z^{-2}} \right) y(t) \quad (8.7)$$

with

$$\boldsymbol{\theta}_n = [\theta_1 \quad \dots \quad \theta_n]^T \quad (8.8)$$

and  $e_P(t, \boldsymbol{\theta}_P) = e(t, \boldsymbol{\theta})$ . Then the  $n$ th subproblem is defined as follows:

$$\min_{\boldsymbol{\theta}_n} V_n(\boldsymbol{\theta}_n) = \min_{\boldsymbol{\theta}_n} \frac{1}{N} \sum_{t=1}^N e_n^2(t, \boldsymbol{\theta}_n). \quad (8.9)$$

Note that the minimization in (8.9) is performed w.r.t.  $\boldsymbol{\theta}_n$  only, while the objective  $V_n(\boldsymbol{\theta}_n)$  depends on the entire vector  $\boldsymbol{\theta}_n$ . However, if the subproblems are solved consecutively, starting at  $n = 1$ , then in the  $n$ th subproblem, estimates for  $\theta_1, \dots, \theta_{n-1}$  are available and only  $\theta_n$  needs to be estimated. As a consequence, the subproblems are entirely decoupled and can be treated individually.

The solution to the  $n$ th subproblem is obtained iteratively using a line search optimization method [6, Ch. 3], i.e.,

$$\hat{\boldsymbol{\theta}}_n^{(k+1)} = \hat{\boldsymbol{\theta}}_n^{(k)} + \mu_k p^{(k)} \quad (8.10)$$

with  $k \in \mathbb{N}$  the iteration index. The step length  $\mu_k$  is determined using backtracking with Armijo's sufficient decrease condition [6, Ch. 3]. The search direction  $p^{(k)}$  can be obtained with one of the following methods:

1. Steepest descent (SD):

$$p^{(k)} = -\frac{\partial}{\partial \theta_n} V_n(\hat{\theta}_n^{(k)}) \quad (8.11)$$

in which

$$\hat{\theta}_n^{(k)} = [\hat{\theta}_1^{(\kappa_1)} \quad \dots \quad \hat{\theta}_{n-1}^{(\kappa_{n-1})} \quad \hat{\theta}_n^{(k)}]^T \quad (8.12)$$

with  $\kappa_i$ ,  $i = 1, \dots, n-1$  the index of the final iteration in the  $i$ th subproblem, and  $k$  the current iteration index in the  $n$ th subproblem.

2. Gauss-Newton (GN):

$$p^{(k)} = -\frac{\left(\frac{\partial}{\partial \theta_n} \mathbf{e}_n(\hat{\theta}_n^{(k)})\right)^T \mathbf{e}_n(\hat{\theta}_n^{(k)})}{\left(\frac{\partial}{\partial \theta_n} \mathbf{e}_n(\hat{\theta}_n^{(k)})\right)^T \left(\frac{\partial}{\partial \theta_n} \mathbf{e}_n(\hat{\theta}_n^{(k)})\right)} \quad (8.13)$$

with

$$\mathbf{e}_n(\theta_n) = [e_n(1, \theta_n) \quad \dots \quad e_n(N, \theta_n)]^T. \quad (8.14)$$

3. Quasi-Newton with damped BFGS updating [6, Ch. 18] (BFGS) :

$$p^{(k)} = -B_k^{-1} \frac{\partial}{\partial \theta_n} V_n(\hat{\theta}_n^{(k)}) \quad (8.15)$$

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{u_k u_k^T}{u_k^T s_k} \quad (8.16)$$

with

$$s_k = \hat{\theta}_n^{(k+1)} - \hat{\theta}_n^{(k)} \quad (8.17)$$

$$v_k = \frac{\partial}{\partial \theta_n} V_n(\hat{\theta}_n^{(k+1)}) - \frac{\partial}{\partial \theta_n} V_n(\hat{\theta}_n^{(k)}) \quad (8.18)$$

$$u_k = \lambda v_k + (1 - \lambda) B_k s_k \quad (8.19)$$

and

$$\lambda = \begin{cases} 1 & \text{if } s_k^T v_k \geq \gamma s_k^T B_k s_k \\ (1 - \gamma) \frac{s_k^T B_k s_k}{s_k^T B_k s_k - s_k^T v_k} & \text{if } s_k^T v_k < \gamma s_k^T B_k s_k \end{cases} \quad (8.20)$$

$$\lambda = \begin{cases} 1 & \text{if } s_k^T v_k \geq \gamma s_k^T B_k s_k \\ (1 - \gamma) \frac{s_k^T B_k s_k}{s_k^T B_k s_k - s_k^T v_k} & \text{if } s_k^T v_k < \gamma s_k^T B_k s_k \end{cases} \quad (8.21)$$

The parameter  $\gamma \in (0, 1)$  is usually chosen as  $\gamma = 0.2$  [8]. Since each CPZLP subproblem is a scalar optimization problem, the Hessian approximation  $B_k$ , the displacement vector  $s_k$ , the change of gradients vector  $v_k$  and its damped counterpart  $u_k$  are all scalars, and the general BFGS calculations in (8.16), (8.19)-(8.21) can be greatly simplified:

$$B_{k+1} = \max\left(\frac{v_k}{s_k}, \gamma B_k\right). \quad (8.22)$$

The iterative algorithm for solving the  $n$ th subproblem is terminated at iteration  $\kappa_n$ , either when

$$\left| p^{(\kappa_n)} \frac{\partial}{\partial \theta_n} V_n(\hat{\boldsymbol{\theta}}_n^{(\kappa_n)}) \right| \leq \tau \quad (8.23)$$

with  $\tau$  a specified tolerance, or when the maximum number of iterations is reached, i.e.,  $\kappa_n = k_{\max}$ .

### 8.2.3 Gradient and Hessian Calculation

The above methods for calculating the search direction  $p^{(k)}$  in the line search algorithm in (8.10) require gradient and Hessian information, which can be calculated using either  $\frac{\partial}{\partial \theta_n} V_n(\boldsymbol{\theta}_n)$  (in the SD and BFGS methods) or  $\mathbf{e}_n(\boldsymbol{\theta}_n)$  and  $\frac{\partial}{\partial \theta_n} \mathbf{e}_n(\boldsymbol{\theta}_n)$  (in the GN method). These quantities can be calculated in an efficient manner as follows.

From the definition of  $e_n(t, \boldsymbol{\theta}_n)$  in (8.7) it follows that

$$e_n(t, \boldsymbol{\theta}_n) = \frac{1 - 2 \cos \theta_n z^{-1} + z^{-2}}{1 - 2\rho \cos \theta_n z^{-1} + \rho^2 z^{-2}} e_{n-1}(t, \boldsymbol{\theta}_{n-1}) \quad (8.24)$$

with  $e_0(t) = y(t)$ . This filtering operation can be executed for  $t = 1, \dots, N$  provided that the initial filter states are known. For simplicity, we set  $e_n(t, \boldsymbol{\theta}_n) = e_{n-1}(t, \boldsymbol{\theta}_{n-1}) = 0$  for  $t \leq 0$ . The intermediate residual signal vector  $\mathbf{e}_n(\boldsymbol{\theta}_n)$  can be constructed from  $e_n(t, \boldsymbol{\theta}_n)$ ,  $t = 1, \dots, N$  as in (8.14).

By differentiating both sides of (8.24) w.r.t.  $\theta_n$ , we obtain

$$\frac{\partial}{\partial \theta_n} e_n(t, \boldsymbol{\theta}_n) = \frac{2(1 - \rho) \sin \theta_n z^{-1} (1 - \rho z^{-2})}{(1 - 2\rho \cos \theta_n z^{-1} + \rho^2 z^{-2})^2} e_{n-1}(t, \boldsymbol{\theta}_{n-1}) \quad (8.25)$$

which can again be calculated for  $t = 1, \dots, N$  by setting  $\frac{\partial}{\partial \theta_n} e_n(t, \boldsymbol{\theta}_n) = e_{n-1}(t, \boldsymbol{\theta}_{n-1}) = 0$  for  $t \leq 0$ . The derivative vector  $\frac{\partial}{\partial \theta_n} \mathbf{e}_n(\boldsymbol{\theta}_n)$  is constructed using  $\frac{\partial}{\partial \theta_n} e_n(t, \boldsymbol{\theta}_n)$ ,  $t = 1, \dots, N$ .

Finally, the gradient  $\frac{\partial}{\partial \theta_n} V_n(\boldsymbol{\theta}_n)$  can be calculated by differentiating (8.9), i.e.,

$$\frac{\partial}{\partial \theta_n} V_n(\boldsymbol{\theta}_n) = \frac{2}{N} \sum_{t=1}^N \left( \frac{\partial}{\partial \theta_n} e_n(t, \boldsymbol{\theta}_n) \right) e_n(t, \boldsymbol{\theta}_n) \quad (8.26)$$

$$= \frac{2}{N} \left( \frac{\partial}{\partial \theta_n} \mathbf{e}_n(\boldsymbol{\theta}_n) \right)^T \mathbf{e}_n(\boldsymbol{\theta}_n). \quad (8.27)$$

The above quantities should be evaluated at  $\boldsymbol{\theta}_n = \hat{\boldsymbol{\theta}}_n^{(k)}$  for calculating the search direction in (8.11), (8.13), and (8.15). It follows from (8.12) that this

evaluation can be achieved by replacing  $\theta_n$  with its current estimate  $\hat{\theta}_n^{(k)}$  in the filter transfer functions in (8.24) and (8.25), and by evaluating  $e_{n-1}(t, \theta_{n-1})$  at  $\theta_{n-1} = \hat{\theta}_{n-1}^{(\kappa_{n-1})}$  (which has already been done in the final iteration of the  $(n-1)$ th subproblem).

### 8.3 Computational Complexity

The operations that are performed in each iteration  $k$  of the  $n$ th subproblem in the CPZLP algorithm are summarized in Table 8.1, with reference to the relevant equations and with the number of multiplications as a measure for computational complexity. The number of backtracking steps needed until Armijo's sufficient decrease condition is satisfied in iteration  $k$  of subproblem  $n$ , is denoted by  $\beta_{n,k}$ .

The computational complexity of the entire CPZLP algorithm can then be calculated as follows. As an example, we derive the total number of multiplications  $M_{\text{BFGS}}$  when the BFGS method is applied. From Table 8.1, we have

$$M_{\text{BFGS}} = \sum_{n=1}^P \left( \sum_{k=1}^{\kappa_n} ((13 + 3\beta_{n,k})N + (17 + 5\beta_{n,k})) \right) \quad (8.28)$$

$$= (13N + 17) \sum_{n=1}^P \kappa_n + (3N + 5) \sum_{n=1}^P \left( \sum_{k=1}^{\kappa_n} \beta_{n,k} \right) \quad (8.29)$$

$$= (13N + 17)\bar{\kappa}P + (3N + 5) \sum_{n=1}^P \kappa_n \bar{\beta}_n \quad (8.30)$$

with  $\bar{\kappa}$  the average number of iterations per subproblem and  $\bar{\beta}_n$  the average number of backtracking steps per iteration in subproblem  $n$ , i.e.,

$$\bar{\kappa} = \frac{1}{P} \sum_{n=1}^P \kappa_n, \quad \bar{\beta}_n = \frac{1}{\kappa_n} \sum_{k=1}^{\kappa_n} \beta_{n,k}. \quad (8.31)$$

Assuming that the average number of backtracking steps per iteration is the same for all subproblems, i.e.,  $\bar{\beta}_1 = \dots = \bar{\beta}_P = \bar{\beta}$ , the computational complexity for the three different methods can be written as

$$\begin{array}{l} M_{\text{SD}} = \bar{\kappa}P[(13 + 3\bar{\beta})N + (14 + 5\bar{\beta})] \\ M_{\text{GN}} = \bar{\kappa}P[(14 + 3\bar{\beta})N + (15 + 5\bar{\beta})] \\ M_{\text{BFGS}} = \bar{\kappa}P[(13 + 3\bar{\beta})N + (17 + 5\bar{\beta})] \end{array} \quad (8.32)$$

From the above expressions, it is clear that the computational complexity is linear w.r.t. both the frame length  $N$  and the number of sinusoids  $P$ . As a

Table 8.1: CPZLP complexity comparison: number of multiplications in iteration  $k$  of subproblem  $n$ 

calculation of	Eqs.	SD	GN	BFGS
gradient	(8.24)-(8.26)	$10N + 6$	$10N + 6$	$10N + 6$
Hessian	(8.13), (8.17), (8.18), (8.22)	0	$N$	2
search direction	(8.11), (8.13), (8.15)	1	2	2
termination criterion	(8.23)	1	1	1
step length	[6, p. 37]	$(1 + \beta_{n,k})(3N + 5)$	$(1 + \beta_{n,k})(3N + 5)$	$(1 + \beta_{n,k})(3N + 5)$
parameter estimate	(8.10)	1	1	1
TOTAL		$(13 + 3\beta_{n,k})N + (14 + 5\beta_{n,k})$	$(14 + 3\beta_{n,k})N + (15 + 5\beta_{n,k})$	$(13 + 3\beta_{n,k})N + (17 + 5\beta_{n,k})$



consequence of decoupling the problem into scalar subproblems, the GN and BFGS methods are not significantly more expensive than the SD algorithm, although they do take into account Hessian information in the optimization algorithm. The actual complexity depends on the average number of iterations and backtracking steps per iteration in the  $P$  subproblems. This leads to the peculiar observation that the fastest converging method will also have the lowest complexity, which is in contrast with the traditional trade-off between convergence speed and complexity.

## 8.4 Simulation Results

Monte Carlo simulations were carried out to validate the performance of the CPZLP algorithm. The observed signal is a sum of  $P = 3$  sinusoids, with amplitudes  $[A_1, A_2, A_3] = [1, 0.5, 1.5]$ , radial frequencies  $[\omega_1, \omega_2, \omega_3] = [0.25, 0.4, 0.7]\pi$ , and phases  $[\phi_1, \phi_2, \phi_3] = [0, 0.8, 1.5]\pi$ . The pole radius is fixed to  $\rho = 0.95$ , which appears to be an optimal value for most sinusoidal frequency estimation problems [7]. The optimization algorithm parameters are set as recommended in [6]: the initial step length is  $\mu_k^{(0)} = 1$ , the contraction factor determining the step length  $\mu_k^{(m)} = \eta^m \mu_k^{(0)}$  in the  $m$ th backtracking step is  $\eta = 0.9$ , the scaling factor determining Armijo's sufficient decrease condition is  $c = 10^{-4}$ , Powell's parameter in the damped BFGS update is  $\gamma = 0.2$  [8], the termination criterion tolerance is  $\tau = 10^{-6}$ , and the maximum number of iterations per subproblem is  $k_{\max} = 30$ . The initial estimate  $\hat{\theta}_n^{(0)} = \pi/3$  is chosen equal for all three subproblems, to illustrate the sensitivity of the algorithm w.r.t. the choice of initial conditions. An additional rescue procedure is implemented, which restarts the iterative procedure for subproblem  $n$  with a different initial estimate if  $\kappa_n = k_{\max}$ . When after five rescue restarts, subproblem  $n$  still remains unsolved, we set  $\hat{\theta}_n = \pi/2$  and continue with subproblem  $n + 1$ .

The CPZLP algorithm is evaluated w.r.t. frequency bias and frequency variance, defined as ( $n = 1, 2, 3$ )

$$\text{bias}(\hat{\theta}_n^{(\kappa_n)}) = E\{\hat{\theta}_n^{(\kappa_n)}\} - \omega_n \quad (8.33)$$

$$\text{var}(\hat{\theta}_n^{(\kappa_n)}) = E\{(\hat{\theta}_n^{(\kappa_n)} - \omega_n)^2\}. \quad (8.34)$$

The expectation operator  $E\{\cdot\}$  in (8.33)-(8.34) is approximated by averaging over 100 simulation runs, with different realizations of the Gaussian white noise signal  $r(t)$ . The CPZLP algorithm was found to produce approximately unbiased frequency estimates for  $N \geq 256$  and SNR  $\geq 0$  dB (SD),  $N \geq 512$  and SNR  $\geq 25$  dB (GN), and  $N \geq 512$  and SNR  $\geq 10$  dB (BFGS).

The frequency variance is displayed in Figs. 8.1(a)-(c) as a function of different frame lengths  $N \in [64, 8192]$ , with SNR = 15 dB. The CRLB for estimating

$\omega_n$ ,  $n = 1, 2, 3$ , from the true signal model in (8.1) is also shown in Fig. 8.1, and was calculated under the assumption that the sinusoidal frequencies are not near 0 and  $\pi$  [9, Ch. 3] and sufficiently separated from each other [7]. In this case, the Fisher information matrix is diagonal and the CRLBs for the different frequencies are independent and equal to [7], [9, Ch. 3]

$$\text{CRLB}(\omega_n) = \frac{6}{N(N+1)(2N+1)\text{SNR}_n} \quad (8.35)$$

with  $\text{SNR}_n = A_n^2/(2\sigma_r^2)$  and  $\sigma_r^2$  the noise variance. It can be seen that with the GN method, only  $\text{var}(\hat{\theta}_1^{(\kappa_1)})$  comes close to the CRLB, which is probably due to the proximity of  $\hat{\theta}_1^{(0)}$  to  $\omega_1$  and the relatively good  $\text{SNR}_1$ . The BFGS method performs much better, with all three variance curves staying near the CRLB for  $N \geq 512$ . Figs. 8.1(d)-(f) show the frequency variance versus  $\text{SNR} \in [-10, 40]$  dB, with  $N = 2048$ . With the GN and BFGS methods there is a clear threshold effect, i.e., the variance suddenly drops for  $\text{SNR} \geq 25$  dB (GN) and  $\text{SNR} \geq 15$  dB (BFGS). In favorable estimation conditions the threshold effect can occur at  $\text{SNR}$  as low as 5 dB (which is illustrated by the  $\text{var}(\hat{\theta}_1^{(\kappa_1)})$  curve in Figs. 8.1(e)-(f)).

To have an idea of the actual computational complexity, the required number of iterations  $\kappa_n$  and the average number of backtracking steps per iteration  $\bar{\beta}_n$  are plotted for  $n = 1, 2, 3$  as a function of  $N$  and  $\text{SNR}$  in Fig. 8.2. It is clear that the SD method is not suited for the frequency estimation problem under consideration. The GN method requires more iterations than the BFGS method, but due to the fact that GN consistently produces estimates that meet Armijo's sufficient decrease condition without backtracking ( $\bar{\beta}_n \equiv 0$ ), it is computationally cheaper than BFGS.

## 8.5 Conclusion

We have presented a new parametric frequency estimation method for multiple real sinusoids corrupted by noise. The so-called CPZLP algorithm provides frame-based frequency estimation by optimizing the parameters of a cascade of second-order constrained pole-zero filter sections in a decoupled and consecutive fashion. Each of the unknown frequencies is estimated using a line search optimization algorithm, which has been implemented with three popular line search methods (SD, GN, and BFGS). The computational complexity of the CPZLP algorithm is linear w.r.t. both the number of sinusoids and the frame length, such that long data frames can be used and hence noise robustness is increased. Monte Carlo simulation results show that the BFGS method is particularly promising, since it provides unbiased and near-optimal frequency estimates for frame lengths larger than 512 samples and  $\text{SNR}$  as low as 5 dB

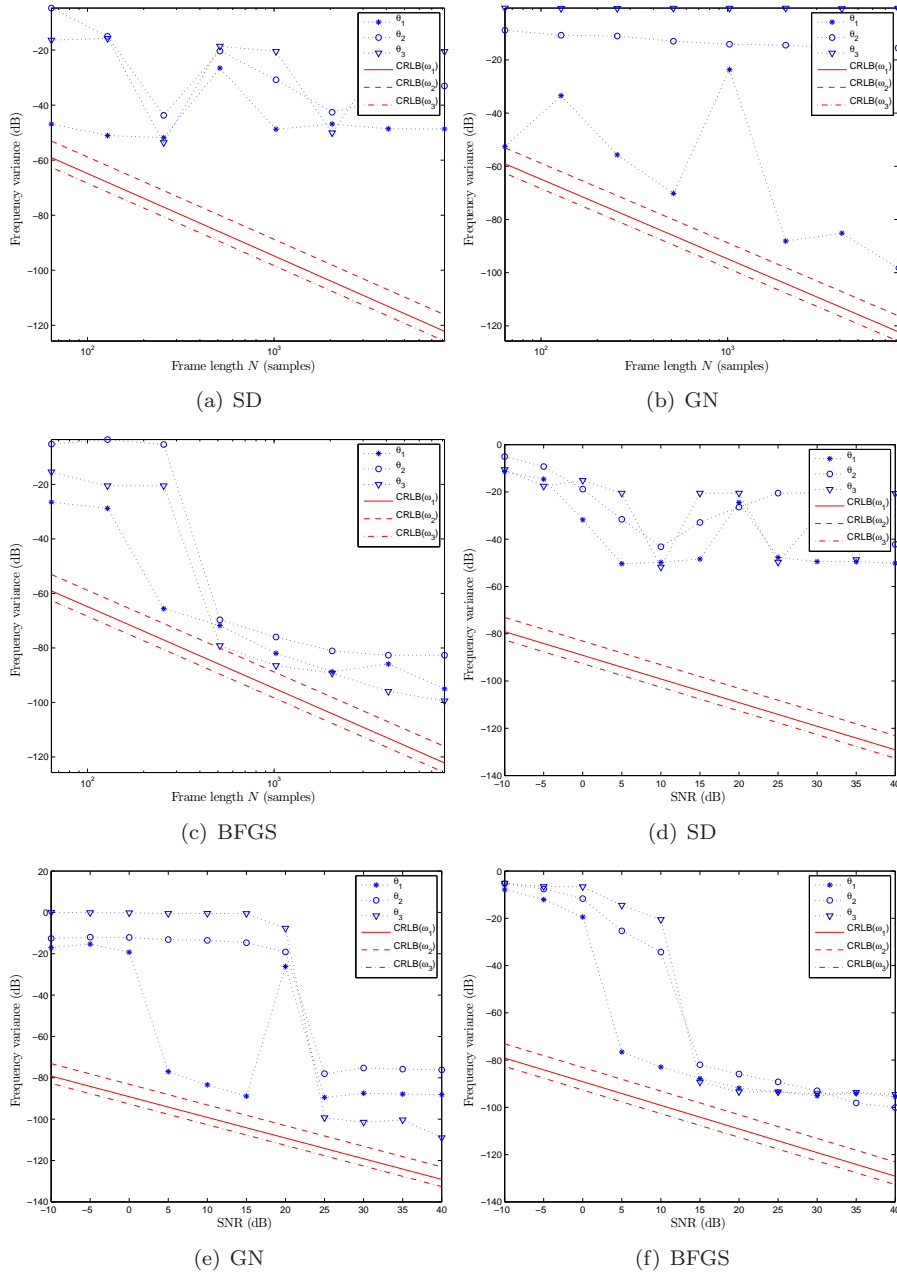


Figure 8.1: CPZLP frequency variance and CRLB, (a)-(c) versus frame length  $N$ , (d)-(f) versus SNR.

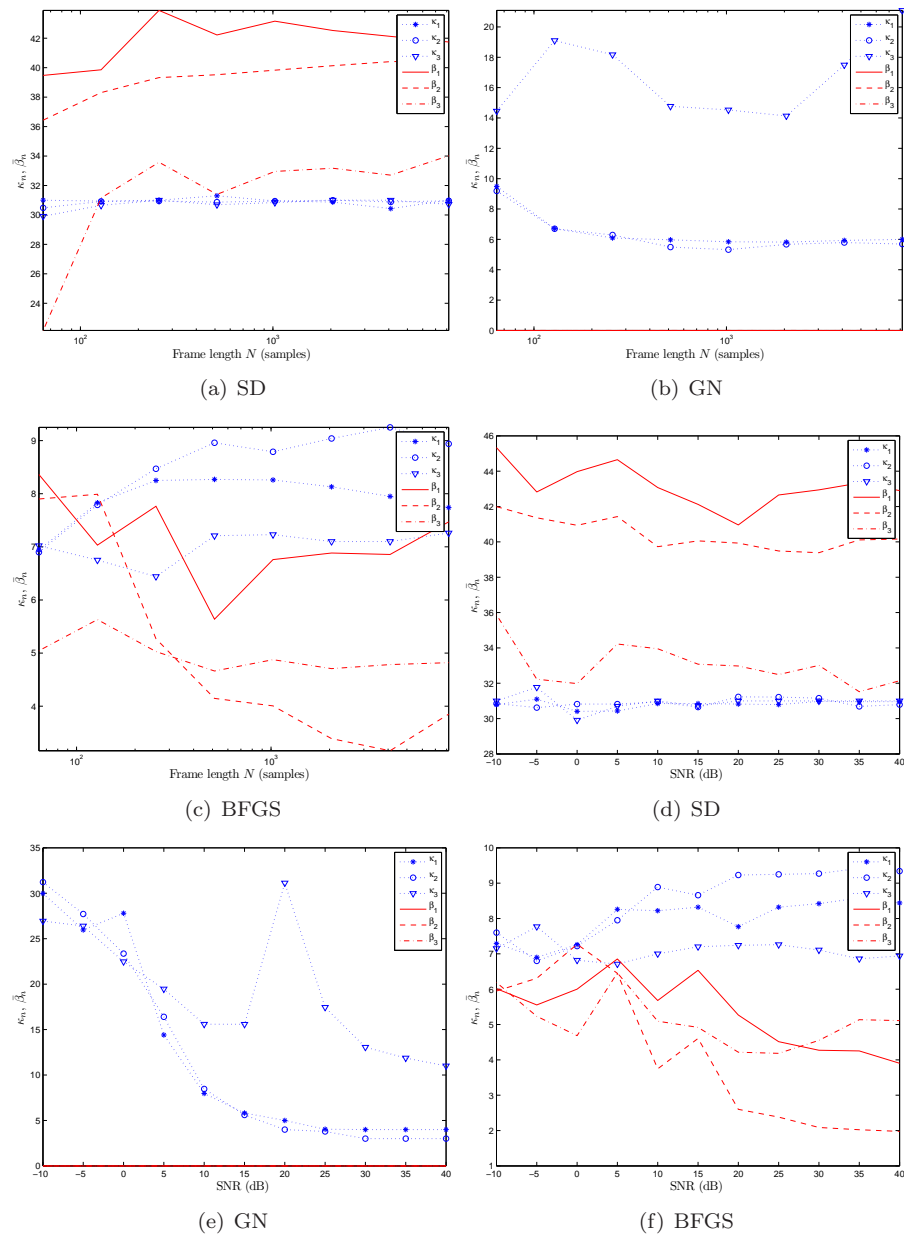


Figure 8.2: CPZLP number of iterations  $\kappa_n$  and average number of backtracking steps per iteration  $\bar{\beta}_n$ , (a)-(c) versus  $N$ , (d)-(f) versus SNR.

in favorable estimation conditions and 15 dB in worse conditions. Since the required number of iterations and backtracking steps has a profound effect on the actual complexity, the faster converging GN and BFGS methods are computationally much more interesting than the SD method. Further work [7] includes an extension of the CPZLP algorithm to multi-pitch estimation and an approach to joint order and frequency estimation.

## Bibliography

- [1] H. C. So, K. W. Chan, Y. T. Chan, and K. C. Ho, "Linear prediction approach for efficient frequency estimation of multiple real sinusoids: algorithms and analyses," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2290–2305, Jul. 2005.
- [2] S. M. Kay, "The effects of noise on the autoregressive spectral estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 5, pp. 478–485, Oct. 1979.
- [3] A. Nehorai, "A minimal parameter adaptive notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 4, pp. 983–996, Aug. 1985.
- [4] M. V. Dragošević and S. S. Stanković, "An adaptive notch filter with improved tracking properties," *IEEE Trans. Signal Process.*, vol. 43, no. 9, pp. 2068–2078, Sep. 1995.
- [5] G. Li, "A stable and efficient adaptive notch filter for direct frequency estimation," *IEEE Trans. Signal Process.*, vol. 45, no. 8, pp. 2001–2009, Aug. 1997.
- [6] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York: Springer, 2006.
- [7] T. van Waterschoot, M. Diehl, and M. Moonen, "Constrained pole-zero linear prediction: Optimization of cascaded biquadratic notch filters for multi-tone and multi-pitch estimation," to be submitted for publication, Mar. 2009, ESAT-SISTA Technical Report TR 07-115, Katholieke Universiteit Leuven, Belgium, Feb. 2008.
- [8] M. J. D. Powell, "A fast algorithm for nonlinearly constrained optimization calculation," in *Proc. 7th Biennial Conf. Numerical Analysis*, Dundee, UK, G. A. Watson, Ed. Berlin: Springer-Verlag, 1977, pp. 144–157.
- [9] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Upper Saddle River, New Jersey: Prentice-Hall, 1993.



## Chapter 9

# AFC for Speech Applications

Acoustic feedback cancellation for long acoustic  
paths using a nonstationary source model

Geert Rombouts, Toon van Waterschoot, Kris Struyve,  
and Marc Moonen

Published in *IEEE Trans. Signal Process.*, vol. 54, no. 9, Sept.  
2006, pp. 3426–3434.

©2006 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

### Contributions of second author

- co-analysis of acoustic feedback problem
- co-development of prediction error identification concept for AFC
- co-development of PEM-AFROW algorithm
- co-development of LTP approach
- co-analysis of computational complexity
- co-design of evaluation experiments
- co-interpretation of simulation results
- co-formulation of conclusion
- text co-editing and co-revision



## Abstract

While several proactive acoustic feedback (Larsen-effect) cancellation schemes have been presented for speech applications with short acoustic feedback paths as encountered in hearing aids, these schemes fail with the long impulse responses inherent to for instance public address systems. We derive a new prediction-error-method (PEM)-based scheme (referred to as PEM-AFROW) which identifies both the acoustic feedback path and the nonstationary speech source model. A cascade of a short- and a long-term predictor removes the coloring and periodicity in voiced speech segments, which account for the unwanted correlation between the loudspeaker signal and the speech source signal. The predictors calculate row operations which are applied to prewhiten the speech source signal, resulting in a least squares system which is solved recursively by means of NLMS or RLS algorithms. Simulations show that this approach is indeed superior to earlier approaches whenever long acoustic channels are dealt with.

## 9.1 Introduction

Acoustic feedback, also referred to as the *Larsen-effect* (howling) occurs in microphone-amplifier-loudspeaker-room systems (Fig. 9.1) when the microphone is too close to the loudspeaker and/or when the amplification is too large. In Fig. 9.1,  $K$  is the amplifier gain,  $y(t)$  is the microphone signal,  $u(t) = Ky(t)$  is the loudspeaker signal,  $\mathbf{f}(t)$  is the feedback path impulse response, and  $v(t)$  is the (speech) source signal. Acoustic feedback occurs when the loop gain is larger than one at a frequency where the loop phase is a multiple of  $2\pi$ .

A conventional solution consists of inserting notch filters into the signal path, thus decreasing the loop gain at those frequencies for which the problem arises. Examples of this approach are available in commercial products (Sabine [1], DBX, Behringer, Shure [2]). The main disadvantage is that the system is reactive, i.e., the howling phenomenon first needs to be detected before the notch filter can be computed and placed in the loop. In most systems, detection takes more than 0.5 s, hence a “proactive” approach is more desirable. Another disadvantage of the notch-filter-based systems is that they do not suppress the “reverb-like” sound which occurs in a system which is marginally stable.

In this paper, we will focus on single-channel acoustic feedback cancellation (AFC) schemes as depicted in Fig. 9.2 (although multi-loudspeaker (beamforming) techniques exist [3],[4]). This type of AFC schemes is inspired by acoustic echo cancellation (AEC) schemes. The filter coefficient vector  $\mathbf{f}(t)$  of the acoustic path from the loudspeaker to the microphone

$$F(q, t) = \mathbf{f}(t)^T \mathbf{q} \quad (9.1)$$

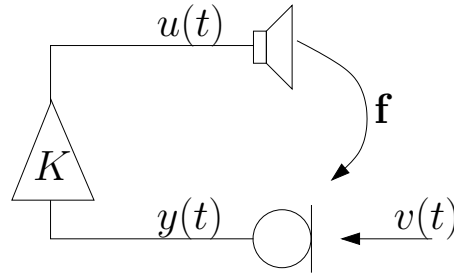


Figure 9.1: A microphone-amplifier-loudspeaker-room system.

$$= [f_0(t) \quad f_1(t) \quad \dots \quad f_{n_F}(t)] \begin{bmatrix} q^0 \\ q^{-1} \\ \vdots \\ q^{-n_F} \end{bmatrix} \quad (9.2)$$

is estimated as  $\hat{\mathbf{f}}(t)$ . Here  $q^{-1}$  is the delay operator. The  $n_F + 1$  coefficients of  $\hat{\mathbf{f}}(t)$  are copied at regular time instants to the cancellation filter  $\hat{\mathbf{f}}_0(t)$ . The loudspeaker signal  $u(t)$  is filtered by the room impulse response  $\mathbf{f}(t)$  and also by the cancellation filter  $\hat{\mathbf{f}}_0(t)$ . The difference between the cancellation filter output and the microphone signal is the feedback-compensated signal  $d(t)$  which should then be equal to the speech source signal  $v(t)$  (for a correct model  $\hat{\mathbf{f}}(t)$ ). In Fig. 9.2,  $e(t)$  is the excitation sequence of the source signal, and

$$H(q, t) = \frac{1}{1 + a_1(t)q^{-1} + \dots + a_{n_A}(t)q^{-n_A}} \quad (9.3)$$

is a time-varying autoregressive (AR) speech model of order  $n_A$ . The coefficient vector of the denominator is  $\mathbf{a}(t)$ . Finally, the  $q^{-d}$  block in Fig. 9.2 is a forward delay, which is often unavoidable in digital implementations (buffers for A/D and D/A converters, ...), and which will be exploited further on.

An AEC-like approach has been used for AFC in, e.g., [5],[6]. In AEC, an important assumption is that the “near-end speech signal”  $v(t)$  is uncorrelated with the loudspeaker signal  $u(t)$ , and hence Wiener filtering theory can be applied in order to identify the room impulse response with only the loudspeaker signal as an input to the system. The main complication in AFC compared to AEC is that this assumption is not fulfilled anymore. It is known that a correlation between the speech source signal  $v(t)$  and the loudspeaker signal  $u(t)$  results in a bias in the identified room impulse response when direct identification is applied (i.e., when the room impulse response is identified as if the system were operating in open loop) [7],[8].

This bias can be removed using the prediction error method (PEM), which

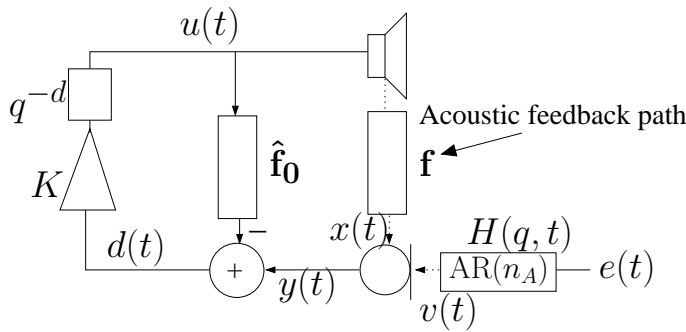


Figure 9.2: Acoustic feedback cancellation scheme.

incorporates a speech source signal model into the identification procedure [9]. The speech source signal in this system identification setup is to be considered as a “noise” signal, and its parameters have to be estimated along with the room impulse response. This speech model can then be used to prewhiten the speech source signal component in both the loudspeaker signal and the microphone signal. The feedback problem has been studied mainly in the hearing aids context where the feedback path impulse response is less than 5 ms. In this paper, we focus on public address (PA) systems, where the feedback path typically has a much longer impulse response, e.g., up to 500 ms, and hence an alternative approach will be needed.

Speech, although highly nonstationary over longer time periods, is often considered to be stationary during short frames of ca. 20 ms (e.g., 160 samples at 8 kHz). Within these frames, it can be whitened by a cascade of a short-term predictor (STP) and a long-term predictor (LTP). The STP models the filtering operation performed by the vocal tract, and the LTP models the periodic behavior of the glottal excitation sequence during voiced speech. The STP identifies an AR process of order  $n_A$  (e.g.,  $n_A = 10$  at 8 kHz). The LTP is an AR process with basically one single non-zero tap at a lag which can vary between 20 and 160 samples at 8 kHz. The room impulse response length for PA applications on the other hand (unlike for hearing aid applications) can have several thousand of non-zero taps, and is mostly stationary over longer periods of time. So it is advantageous to use data windows of several seconds to estimate the room impulse response, over which the speech signal then will be nonstationary. This contrast between the long stationarity period of the long room impulse response and the short stationarity period of the STP speech model, and the corresponding number of data points which are available to identify each of them, is fundamental to the problem of AFC for PA systems.

In [9], the authors derive a two-channel algorithm (2ch-AF), which uses the

same data window length for the identification of the speech model as for the room impulse response identification. This is justified for short acoustic paths such as in hearing aids. For PA systems, this approach is suboptimal as is shown in [10], but it still performs better than so-called direct identification, since it still provides some degree of prewhitening (with an “averaged” speech model).

Another method is described in [11], and its theoretical basis is derived in [9]. We will refer to this algorithm as PEM-AF. It effectively distinguishes between the estimation window length for the speech model and for the room impulse response model, but it implicitly uses an assumption about the stationarity of the speech model, which is not valid for long acoustic impulse responses as in PA applications. Existing methods also use an incomplete speech model (STP only), which provides some whitening, but does not remove the periodicity due to the glottal excitation, resulting in the loudspeaker signal still being correlated to the source signal during voiced speech.

In this paper, we introduce a new technique which estimates the speech model over short time windows (over which it is stationary), and the room impulse model over longer time windows (which is necessary because the number of parameters is much larger). The speech model is then not required to be stationary during the complete room impulse response. Our scheme will also include an LTP which models the periodicity in  $e(t)$ . The difference with the existing schemes like [9] is, aside from this LTP, also the “row transformation” approach. The row transformations allow the whitening to be performed correctly (they can be interpreted as a decorrelation of the speech source component in the right hand side of a least squares system), and we will show that this scheme indeed outperforms existing methods. These row transformations are required to perform the prewhitening step correctly, hence the algorithm is named “prediction-error-method-based adaptive filtering with row operations” (PEM-AFROW).

This paper is organized as follows. In Section 9.2, we first review the prediction error method, 2ch-AF and PEM-AF. In Sections 9.3 and 9.4, we introduce our new procedure. It uses alternating updates of the speech model and the adaptive filter which models the room. In Section 9.7 we show simulation results, and Section 9.8 contains the conclusion of the paper.

## 9.2 Prediction Error Method

In [9] the so-called prediction error method (known from system identification theory [12]) is applied to the AFC problem. Under the assumption of stationarity, it is proved that if in a scheme as shown in Fig. 9.3 the forward delay  $d$  is larger than the order  $n_A$  of the AR process which generates the

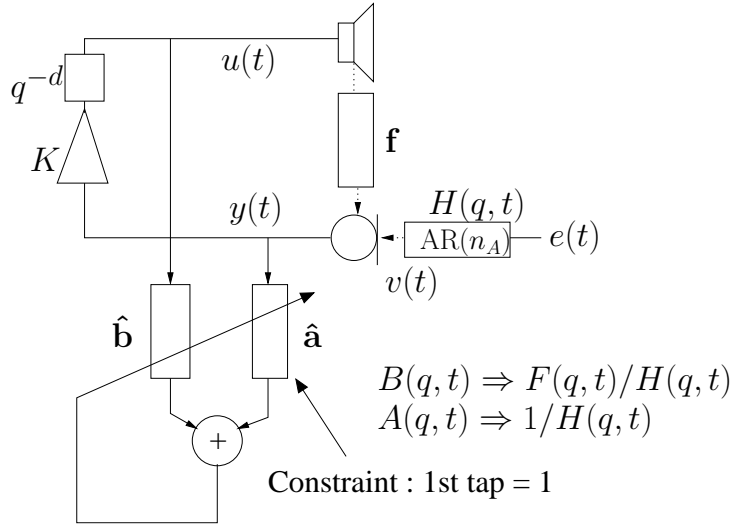


Figure 9.3: Identification part of the 2ch-AF scheme [9].

speech signal, the two-channel MMSE-based adaptive filter  $\{\hat{A}(q, t), \hat{B}(q, t)\}$  with filter coefficient vectors  $\{\hat{\mathbf{a}}(t), \hat{\mathbf{b}}(t)\}$  will converge to the unique solution  $\{H^{-1}(q, t), F(q, t)H^{-1}(q, t)\}$ .

This allows to introduce a cancellation filter  $\hat{F}_0(q, t) = A^{-1}(q, t)B(q, t)$  into the scheme (see Fig. 9.2). If this is implemented as an IIR filter, stability monitoring is needed, but it is a trivial modification to perform a deconvolution first, which results in an FIR cancellation filter.

The filters  $\hat{\mathbf{b}}(t)$  and  $\hat{\mathbf{a}}(t)$  are updated concurrently in the two parallel branches, and so the scheme is referred to as two-channel adaptive feedback canceller (2ch-AF). The estimation data window is the same for both branches, and hence the scheme provides good results in hearing aid applications, where the speech signal is stationary over the data window length required to estimate the acoustic path.

PEM-AF was proposed in [9],[11]. This technique uses two independently updated adaptive filters  $\hat{\mathbf{f}}(t)$  and  $\hat{\mathbf{a}}(t)$ , as shown in Fig. 9.4. The estimation of  $\hat{\mathbf{f}}(t)$  and  $\hat{\mathbf{a}}(t)$  can be performed over different data window lengths (different RLS forgetting factors). In PEM-AF,  $\hat{\mathbf{a}}(t)$  decorrelates the *scalar* output residuals of the adaptive filter  $\hat{\mathbf{f}}(t)$ . In a stationary setup, where the order of both filters is arbitrary, this is equivalent to a prewhitening operation for the input of the filter  $\hat{\mathbf{f}}(t)$ . Also this scheme performs well in hearing aid applications, where the speech model does not change over the data window needed to estimate

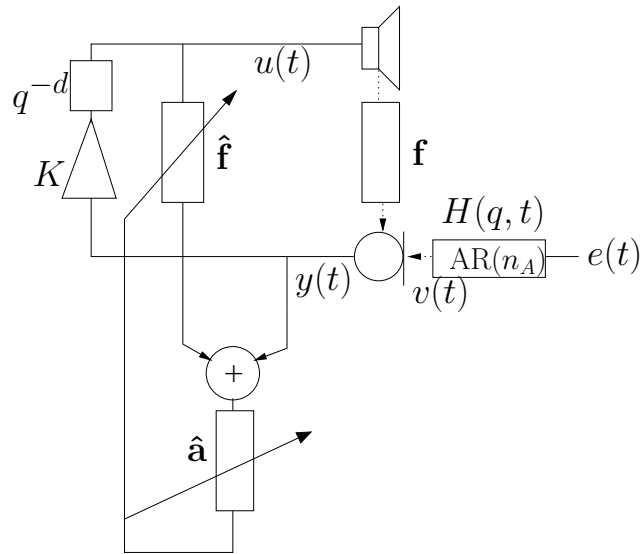


Figure 9.4: PEM-AF scheme [9],[11]. Only the identification part is shown, the estimate  $\hat{\mathbf{f}}$  can be used in a scheme like Fig. 9.2 to perform acoustic feedback cancellation.

the acoustic impulse response.

As explained in Section 9.1, this assumption does not hold in PA applications. In the next section we will derive a new method with much better performance for long acoustic paths. It will become clear that PEM-AF can be viewed as an approximation of this method. The performance gain of the new method in the next section will be shown by simulations.

### 9.3 Proposed Algorithm: PEM-AFROW

We model the speech source signal as a time-varying AR (TVAR) signal of order  $n_A$  (see Figs. 9.1 and 9.2):

$$v(t) = a_1(t)v(t-1) + \dots + a_{n_A}(t)v(t-n_A) + b(t)e(t). \quad (9.4)$$

It is instructive to first assume that  $e(t)$ , the excitation sequence, is a white noise sequence, and the factor which accounts for the energy variations in the excitation sequence,  $b(t) = 1$ . Further on, we will only assume  $b(t)$  to be constant during the length of a frame and in Section 9.4, we will use a more

general model for  $e(t)$ . We start from the minimization problem

$$\min_{\hat{\mathbf{f}}(t)} \left\| \mathbf{U}\hat{\mathbf{f}}(t) - \mathbf{y} \right\|, \quad (9.5)$$

with

$$\mathbf{U} = \begin{bmatrix} u(t) & u(t-1) & \dots & u(t-n_F) \\ u(t-1) & u(t-2) & \dots & u(t-n_F+1) \\ \vdots & \vdots & \ddots & \vdots \\ u(0) & 0 & \dots & 0 \end{bmatrix}, \quad (9.6)$$

$$\mathbf{y} = \begin{bmatrix} y(t) \\ y(t-1) \\ \vdots \\ y(0) \end{bmatrix}. \quad (9.7)$$

This minimization results in a biased estimate  $\hat{\mathbf{f}}(t)$  for  $\mathbf{f}(t)$  since  $y(t)$  contains the “disturbance”  $v(t)$  which is correlated with  $u(t)$ , i.e.,

$$\mathbf{y} = \mathbf{U}\mathbf{f}(t) + \mathbf{v} \quad (9.8)$$

with

$$\mathbf{v} = \begin{bmatrix} v(t) \\ v(t-1) \\ \vdots \\ v(0) \end{bmatrix}. \quad (9.9)$$

If we have the finite impulse response filter

$$A(q, t) = H^{-1}(q, t) \quad (9.10)$$

available at each time instant, with coefficients  $\mathbf{a}(t)$ , we can apply a prewhitening by forming the matrix

$$\mathbf{A} = \begin{bmatrix} 1 & a_1(t) & a_2(t) & \dots & a_{n_A}(t) & 0 & \dots & 0 \\ 0 & 1 & a_1(t-1) & \dots & a_{n_A-1}(t-1) & a_{n_A}(t-1) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & \dots & 1 \end{bmatrix} \quad (9.11)$$

and modifying the minimization problem (9.5) to

$$\min_{\hat{\mathbf{f}}(t)} \left\| \mathbf{A}\mathbf{U}\hat{\mathbf{f}}(t) - \mathbf{A}\mathbf{y} \right\|. \quad (9.12)$$

This will effectively whiten the component  $v(t)$  in  $y(t)$ , while performing only row operations on the input matrix  $\mathbf{U}$ , i.e., from (9.8)

$$\mathbf{A}\mathbf{y} = \mathbf{A}\mathbf{U}\mathbf{f}(t) + \mathbf{e}, \quad (9.13)$$

and thus lead to an unbiased estimate  $\hat{\mathbf{f}}(t)$ . If we have an estimate  $\hat{A}(q, t)$  of  $H^{-1}(q, t)$  available at each time instant, with coefficients  $\hat{\mathbf{a}}(t) \in \mathbb{R}^{n_A+1}$ , we can apply a prewhitening by forming the matrix

$$\hat{\mathbf{A}} = \begin{bmatrix} 1 & \hat{a}_1(t) & \hat{a}_2(t) & \dots & \hat{a}_{n_A}(t) & 0 & \dots & 0 \\ 0 & 1 & \hat{a}_1(t-1) & \dots & \hat{a}_{n_A-1}(t-1) & \hat{a}_{n_A}(t-1) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & \dots & 1 \end{bmatrix} \quad (9.14)$$

and modifying the minimization problem (9.5) to

$$\min_{\hat{\mathbf{f}}(t)} \left\| \hat{\mathbf{A}} \mathbf{U} \hat{\mathbf{f}}(t) - \hat{\mathbf{A}} \mathbf{y} \right\|. \quad (9.15)$$

We now introduce the assumption that  $\mathbf{a}(t)$  and  $b(t)$  are constant during frames of 20 ms, which is a common assumption for speech signals. This means that we rewrite

$$\hat{\mathbf{A}} = \begin{bmatrix} 1 & \hat{a}_i^{(1)} & \dots & \hat{a}_i^{(n_A)} & 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 1 & \hat{a}_i^{(1)} & \hat{a}_i^{(2)} & \dots & \hat{a}_i^{(n_A)} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 & \hat{a}_{i-1}^{(1)} & \dots & \hat{a}_{i-1}^{(n_A-1)} & \hat{a}_{i-1}^{(n_A)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & \dots & 1 \end{bmatrix} \begin{matrix} \uparrow \\ M \\ \downarrow \\ \uparrow \\ M \\ \downarrow \\ \vdots \end{matrix} \quad (9.16)$$

and

$$b_i = b(iM) = \dots = b((i+1)M - 1), \quad (9.17)$$

with, e.g.,  $M = 160$  for a sampling rate of 8 kHz, and  $i = \lceil t/M \rceil$ , the first integer larger than or equal to  $t/M$ . This means that  $i$  is the frame index. We define

$$\hat{\mathbf{a}}_i = \left[ 1 \quad \hat{a}_i^{(1)} \quad \dots \quad \hat{a}_i^{(n_A)} \right]^T. \quad (9.18)$$

We now decouple the nonlinear equations in order to calculate the estimates  $\hat{\mathbf{a}}_i$  and the room response  $\hat{\mathbf{f}}(t)$  in an alternating fashion. In the first step, an estimate  $\hat{\mathbf{f}}(iM - 1)$  from the previous frame is used to filter a frame  $iM, \dots, (i+1)M - 1$  of data (20 ms), see Fig. 9.5. The filter output is subtracted from the corresponding microphone samples, resulting in

$$d(t) = y(t) - \mathbf{u}^T(t) \hat{\mathbf{f}}(iM - 1), \quad t = iM, \dots, (i+1)M - 1, \quad (9.19)$$

with

$$\mathbf{u}(t) = \begin{bmatrix} u(t) \\ \vdots \\ u(t - n_F) \end{bmatrix}. \quad (9.20)$$



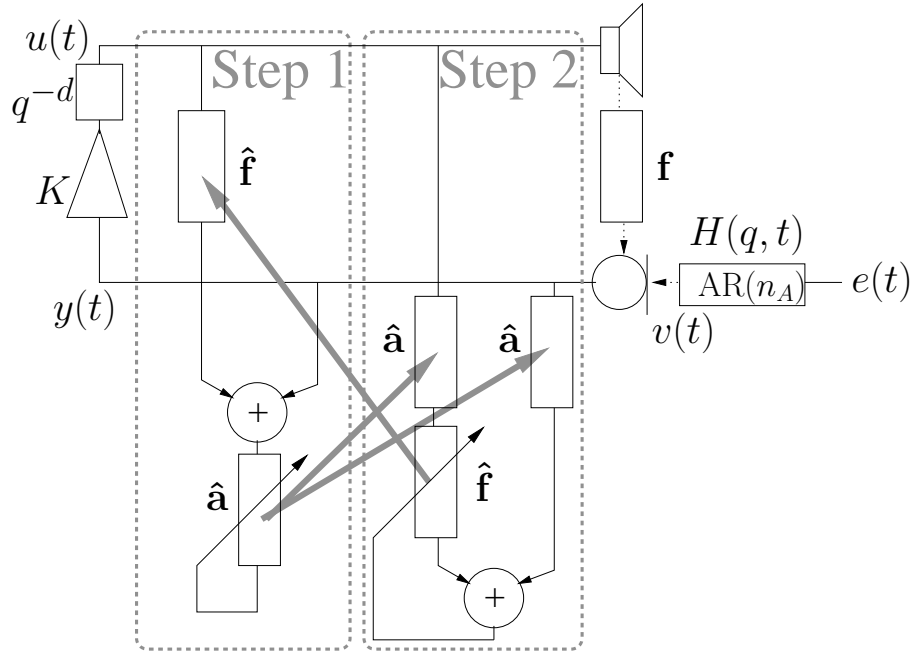


Figure 9.5: PEM-AFROW. In the first step,  $\hat{\mathbf{a}}$  is estimated in the left hand side. It is then copied to the right hand side, where the estimation of  $\hat{\mathbf{f}}$  is performed on the same data frame (step 2). Finally,  $\hat{\mathbf{f}}$  is copied to the left hand side and used in the next frame.

Note that if the estimate  $\hat{\mathbf{f}}(iM - 1)$  were exact, then  $d(t) = v(t)$ . Then linear prediction is performed on this  $d(t)$ ,  $t = iM, \dots, (i+1)M - 1$  (Levinson-Durbin algorithm) to find the linear prediction error filter  $\hat{\mathbf{a}}_i$  for frame  $i$ .

In the second step, (9.15) is solved for  $\hat{\mathbf{f}}(t)$ ,  $t = iM, \dots, (i+1)M - 1$ , which is possible since  $\hat{\mathbf{a}}_i$  is known from step 1. Note that the window length should be set appropriately. This improves the estimate  $\hat{\mathbf{f}}(t)$  for  $\mathbf{f}(t)$ , since  $\mathbf{f}(t)$  is assumed constant or “slowly changing” compared to the frame length. Note that the stimulus energy  $b_i$  is assumed constant over a frame, and this means that the variance on the estimate  $\hat{\mathbf{f}}(t)$  may differ from frame to frame. The two steps can be iterated on the frame. Since none of these two steps will increase

$$E \{ \|\boldsymbol{\varepsilon}\| \} = E \left\{ \left\| \hat{\mathbf{A}}\mathbf{U}\hat{\mathbf{f}}(t) - \hat{\mathbf{A}}\mathbf{y} \right\| \right\}, \quad (9.21)$$

with  $E\{\cdot\}$  the expected value operator, the algorithm will converge to a (possibly local) minimum of (9.15).

In order to reduce the complexity, we will perform only one iteration per frame.

The minimization problem (9.15), with a fixed value of  $\hat{\mathbf{A}}$ , can be solved for  $\hat{\mathbf{f}}(t)$  by means of any adaptive filtering algorithm. We have implemented this both using a QR-decomposition-based RLS (QRD-RLS) algorithm and an NLMS algorithm. The input vector is in both cases

$$\check{\mathbf{u}}^T(t) = \hat{\mathbf{a}}_i^T \begin{bmatrix} \mathbf{u}^T(t) \\ \vdots \\ \mathbf{u}^T(t - n_A) \end{bmatrix}, \quad i = \lceil t/M \rceil, \quad (9.22)$$

and the desired signal input (right hand side sample) is

$$\check{y}(t) = \hat{\mathbf{a}}_i^T \begin{bmatrix} y(t) \\ \vdots \\ y(t - n_A) \end{bmatrix}, \quad i = \lceil t/M \rceil. \quad (9.23)$$

Since  $\mathbf{u}(t)$  is a shifted version of  $\mathbf{u}(t - 1)$  with one sample prepended, and  $\hat{\mathbf{a}}_i$  remains constant during a frame of  $M$  samples,  $\check{\mathbf{u}}(t)$  will be a shifted version of  $\check{\mathbf{u}}(t - 1)$  with one sample prepended. So inside a frame, only one vector multiplication has to be performed to calculate  $\check{\mathbf{u}}(t)$ . On the other hand, at the start of each frame, a matrix multiplication should be performed to calculate all elements of  $\check{\mathbf{u}}(iM)$  as follows:

$$\check{\mathbf{u}}^T(iM) = \hat{\mathbf{a}}_i^T \begin{bmatrix} \mathbf{u}^T(iM + 1) \\ \vdots \\ \mathbf{u}^T(iM - n_A + 1) \end{bmatrix}. \quad (9.24)$$

For real-time implementation, the scheme involves a delay of one frame for the update of  $\hat{\mathbf{f}}(t)$ , since  $\hat{\mathbf{a}}_i$  can only be calculated at time  $t = (i+1)M - 1$ . Note that this is not a problem since we have assumed that the room impulse response is constant over more than one frame. The delay is effectively implemented as a delay line for the input samples  $u(t)$  before they are fed to Eq. (9.22).

Once the room impulse response has been identified, the next step is to insert the cancellation filter into the feedback loop scheme by setting  $\hat{\mathbf{f}}_0(t) = \hat{\mathbf{f}}(t)$ , e.g., at regular time intervals (see Fig. 9.2). It is important to notice that this obviously influences the adaptation. The input data used for the identification procedure then depend on the current model estimate, which is reminiscent of a nonlinear optimization problem. This dependency is effectively ignored in our implementation (it is also ignored in adaptive control theory [13, Ch. 7]). Experiments indicate that updating the cancellation filter regularly is beneficial to the identification process. This can be explained because a time-variant forward path (from microphone to loudspeaker) decreases the correlation between the loudspeaker signal and the speech source signal.

At this point, the difference between PEM-AF and PEM-AFROW becomes obvious: in PEM-AFROW the stationarity of the speech model is explicitly assumed in the minimization problem by stating that  $\hat{\mathbf{a}}_i$  remains constant during a frame (see Eq. (9.16)). At the start of each frame, the full input vector  $\check{\mathbf{u}}^T(t)$  is recalculated. In PEM-AF, this assumption of stationarity is not made for the optimization problem itself (the optimization is decoupled in two completely independent adaptive filters), and the full input vector is never recomputed after a change of  $\hat{\mathbf{a}}(t)$  in PEM-AF. This can only be justified for short acoustic impulse responses, and (as will be shown later on) PEM-AF performs much worse than PEM-AFROW for long acoustic impulse responses.

## 9.4 Long-Term Prediction

For the TVAR signals studied in the previous section (where  $e(t)$  was a white noise sequence), the prewhitening step removes all of the correlation between the loudspeaker signal and the source signal. However, the excitation sequence  $e(t)$  for voiced speech is periodical (glottal excitation). Hence the input signal  $u(t)$  to the adaptive filter is –due to this periodicity– still correlated with the source signal, even after prewhitening. This in practice leads to a severely biased estimate of the room impulse response (the bias being localized at the excitation harmonics).

A possible solution could be to increase the length (span) of the prediction filter  $\mathbf{a}(t)$  so that it becomes longer than the period of the excitation sequence. It would then combine the periodicity together with the AR coefficients for the vocal tract model. The problem here is that the pitch period can be as long as 160 samples, and hence not enough data is then available in a 160 sample window to estimate the  $(160 + n_A - 1)$  prediction filter coefficients.

A standard approach in speech coding [14] is to cascade an STP of order  $n_A$  (e.g., 12), which models the vocal tract characteristics in the feedback-compensated signal  $d(t)$  (defined in (9.19)),

$$w(t) = \hat{\mathbf{a}}_i^T \begin{bmatrix} d(t) \\ \vdots \\ d(t - n_A) \end{bmatrix}, \quad i = \lceil t/M \rceil, \quad (9.25)$$

with an LTP with only one tap and a lag equal to the pitch period to model the periodicity in  $d(t)$ ,

$$\varepsilon(t) = w(t) - \alpha_j w(t - K_j), \quad j = \lceil t/M_{\text{LTP}} \rceil. \quad (9.26)$$

The LTP can be estimated in windows of 20 ms (which is the frame length  $M$  of the STP), with a 10 ms overlap. This means that the LTP model is estimated

each 10 ms, corresponding to  $M_{LTP}$  samples (at 8 kHz,  $M_{LTP} = 80$ ). In order to estimate the LTP, we minimize

$$\min E \left\{ \|\varepsilon(t)\|^2 \right\} = \min_{K_j, \alpha_j} E \left\{ \|w(t) - \alpha_j w(t - K_j)\|^2 \right\}. \quad (9.27)$$

The solution follows from

$$E \{ w(t - K_j) w(t - K_j) \} \alpha_j = E \{ w(t) w(t - K_j) \}. \quad (9.28)$$

In order to estimate these values, we define

$$\mathbf{w}(t) = \begin{bmatrix} w(t) \\ w(t-1) \\ \vdots \\ w(t-M+1) \end{bmatrix}. \quad (9.29)$$

We can then estimate

$$\alpha_j = [\mathbf{w}^T(t - K_j) \mathbf{w}(t - K_j)]^{-1} \mathbf{w}^T(t) \mathbf{w}(t - K_j). \quad (9.30)$$

The variance of the LTP residual is

$$\sigma_j^2 = \mathbf{w}^T(t) \mathbf{w}(t) - \frac{[\mathbf{w}^T(t) \mathbf{w}(t - K_j)]^2}{\mathbf{w}^T(t - K_j) \mathbf{w}(t - K_j)}. \quad (9.31)$$

This variance  $\sigma_j^2$  is evaluated for different values of  $K_j = K_{\min}, \dots, K_{\max}$  (the lag), and the parameters  $\{K_j, \alpha_j\}$  which result in the minimum value of  $\sigma_j^2$  are chosen as the predictor for LTP frame  $j$ .

In the PEM-AFROW method of Section 9.3 and Fig. 9.5, the speech model will now be a cascade of an STP and LTP. The details of the resulting algorithm will be given in Section 9.5.

It is important to note that by applying the LTP, the actual order of the speech source model is the lag of the LTP model plus the order of the STP model, and as stated in [9], to guarantee identifiability, the forward delay must be larger than the order of this model. In practice this delay is approximately 20 ms, but then it does not matter too much where this forward delay is implemented: often a latency  $d$  is introduced by buffering after and before the A/D and D/A converters, or even –due to the relatively low velocity of sound waves– this delay appears automatically from the distance between the loudspeaker and the microphone.

In Section 9.3 it was mentioned that at frame borders, the whole input vector has to be recalculated by means of a matrix multiplication. It must be noted that when the LTP is added to the algorithm, this matrix multiplication has to be performed not only at frame borders of the STP, but also at frame borders of the LTP. The next section will show this in detail.

Note that a multi-tap LTP [14] may be used instead of a single-tap LTP. In the algorithmic description of the next section, we have made abstraction of this in order to keep the pseudo-code readable. In the algorithm description, we also assumed no overlap between the LTP frames. This means that the LTP parameters  $K_j$  and  $\alpha_j$  are estimated only once per STP frame (i.e.,  $i = j$ ).

## 9.5 Algorithm Description

In order to apply row updates, the input vector of the adaptive filter  $\tilde{\mathbf{u}}(t)$  (which consists of the residual signal of the cascade of the STP and LTP) and the vector from which this is computed, namely  $\check{\mathbf{u}}(t)$ , have to be recalculated whenever the STP changes (which happens at each frame border). When the vector  $\check{\mathbf{u}}(t)$  has been recalculated  $K_j$  samples ago ( $K_j$  is the lag of the LTP),  $\tilde{\mathbf{u}}(t)$  has to be recomputed completely (by applying the LTP). At all other time instants,  $\check{\mathbf{u}}(t)$  and  $\tilde{\mathbf{u}}(t)$  are shifted versions of the previous  $\check{\mathbf{u}}(t-1)$  and  $\tilde{\mathbf{u}}(t-1)$ , with only one new element in front. If the LTP data windows overlap, the vector  $\tilde{\mathbf{u}}(t)$  has to be recomputed whenever the LTP changes.

An adaptation threshold  $T_a$  was included in order to account for A/D-D/A or amplifier noise. Without this threshold, the algorithm's coefficients could drift during silence. A complete pseudo-code description is given in Table 9.1.

## 9.6 Complexity

The complexity is evaluated when the algorithm is operated with an NLMS adaptive filter. In these complexity expressions a multiplication and an addition are counted as two separate floating point operations. A “search range”  $K_{\min}$  to  $K_{\max}$  has to be specified for the lag of the LTP (at 8 kHz, typically  $K_{\min} = 20$ ,  $K_{\max} = 160$ ). The complexity depends on these parameters through

$$\Delta_K = K_{\max} - K_{\min}. \quad (9.32)$$

For the complexity calculation we assume a single-tap LTP, and we also assume that the frames do not overlap. Since at each frame border the full NLMS input vector is recalculated, the complexity per sample is

$$8(L_F + n_A) + 4\Delta_K + 5 + \frac{(2n_A + 4)L_F + 4n_A^2 - 5n_A + 15}{M} \quad (9.33)$$

floating point operations, with  $L_F = n_F + 1$ . The algorithm was implemented in C++ on a Pentium III, 1GHz PC without any specific optimization effort, and runs in real time with  $K_{\min} = 40$ ,  $K_{\max} = 320$ ,  $L_F = 2000$ ,  $n_A = 24$ ,  $M = 320$  at 16 kHz sampling rate, with an LTP frame overlap of 160 samples.

Table 9.1: PEM-AFROW. The function `Adapt(coefficients,inputvector, desired signal sample)` is an adaptive filtering algorithm, e.g., RLS or NLMS. The function `LongTermPredictor(short-term prediction residual vector)` performs the operations which are explained in section 9.4.

---

```

for each sample  $u(t), y(t), t = 0, 1, \dots$ 
   $\mathbf{u}(t) = [u(t) \ \bar{\mathbf{u}}^T(t-1)]^T$  //^ means all components but last
   $\mathbf{y}(t) = [y(t) \ \bar{\mathbf{y}}^T(t-1)]^T$  // $\mathbf{u}(t), \mathbf{y}(t)$  have length  $n_A$ 
   $d(t) = y(t) - \hat{\mathbf{f}}_0^T(t)[u(t) \ u(t-1) \ \dots \ u(t-n_F)]^T$  // $d(t)$  is the compensator output
   $\mathbf{d}(t) = [d(t) \ \bar{\mathbf{d}}^T(t-1)]^T$  // $\mathbf{d}(t)$  has length  $M$ 
   $\ddot{u}(t-M) = \hat{\mathbf{a}}_{i-1}^T \mathbf{u}(t-M)$ 
   $\ddot{\mathbf{u}}(t-M) = [\ddot{u}(t-M) \ \bar{\mathbf{u}}^T(t-M-1)]^T$  } //Short-term whitening
   $\ddot{y}(t-M) = \hat{\mathbf{a}}_{i-1}^T \mathbf{y}(t-M)$ 
   $\ddot{u}(t-M) = \ddot{u}(t-M) - \alpha_{i-1} \ddot{u}(t-M-K_{i-1})$ 
   $\ddot{y}(t-M) = \ddot{y}(t-M) - \alpha_{i-1} \ddot{y}(t-M-K_{i-1})$  } //Long-term whitening
   $\ddot{\mathbf{u}}(t-M) = [\ddot{u}(t-M) \ \bar{\mathbf{u}}^T(t-M-1)]^T$ 
  if  $\ddot{\mathbf{u}}^T(t-M)\ddot{\mathbf{u}}(t-M) > T_a$  //Energy threshold  $T_a$ : adapt or not
     $\hat{\mathbf{f}}(t+1) = \text{Adapt}(\hat{\mathbf{f}}(t), \ddot{\mathbf{u}}(t-M), \ddot{y}(t-M))$  //With RLS or NLMS
  end if
  if  $t = iM - 1, i \in \mathbb{Z}$ 
     $\{\hat{\mathbf{a}}_i, \mathbf{w}(t)\} = \text{LevinsonDurbin}(\mathbf{d}(t))$ 
     $\{K_i, \alpha_i\} = \text{LongTermPredictor}(\mathbf{w}(t))$ 
    for  $k = 0, \dots, n_F - 1$  //Recalc. whole NLMS input vector
       $\ddot{u}(t-M-k) = \hat{\mathbf{a}}_i^T \mathbf{u}(t-M-k)$ 
       $\ddot{u}(t-M-k) = \ddot{u}(t-M-k) - \alpha_i \ddot{u}(t-M-k-K_i)$ 
    end for
     $\ddot{\mathbf{u}}(t-M) = [\ddot{u}(t-M) \ \dots \ \ddot{u}(t-M-n_F)]^T$ 
     $\ddot{\mathbf{u}}(t-M) = [\ddot{\mathbf{u}}(t-M) \ \dots \ \ddot{\mathbf{u}}(t-M-n_F)]^T$ 
  end if
  if  $t - K_i = iM - 1, i \in \mathbb{Z}$  //Recalc. LTP on full input vector
    for  $k = 0, \dots, n_F - 1$ 
       $\ddot{u}(t-M+1-k) = \ddot{u}(t-M+1-k) - \alpha_i \ddot{u}(t-M+1-k-K_i)$ 
    end for
  end if
  if DecideToUpdateCompensator( $t$ )
     $\hat{\mathbf{f}}_0(t) = \hat{\mathbf{f}}(t)$ 
  else
     $\hat{\mathbf{f}}_0(t) = \hat{\mathbf{f}}_0(t-1)$ 
  end if
end for

```

---

## 9.7 Simulation Results

In Fig. 9.6, we first compare the PEM-AFROW approach to the decoupled PEM-AF approach for short paths, as they occur in, e.g., hearing aids. In the figure the average of the error norm

$$\left\| \mathbf{f}(t) - \hat{\mathbf{f}}(t) \right\| \quad (9.34)$$

for 7 different speech signal inputs is plotted as a function of time. In order to provide an idea of the variance, also the convergence curves of the individual realizations are shown. Note that only the identification performance (in closed loop) is shown, which means that the cancellation filter  $\hat{\mathbf{f}}_0(t)$  (see Fig. 9.2) is *not* inserted into the scheme during adaptation. If the filter would be inserted, the performance will generally be better, since inserting the filter corresponds in fact to inserting an uncorrelated signal into the loop. The acoustic path has 50 taps. We use NLMS for both approaches, since in a practical implementation this would be the adaptive algorithm of choice (due to complexity constraints). The NLMS step size is 0.04 for PEM-AFROW, while PEM-AF, which uses a modified NLMS algorithm and hence a different definition of the step size, was tuned to give equal initial convergence speed (step size 0.03). This allows us to make a fair comparison of the resulting bias/variance of the solution. Note that the performance of all algorithms is dependent of the energy ratio (“signal to noise ratio”) of the loudspeaker component arriving on the microphone versus the source signal arriving on the microphone (the source signal should thus be interpreted as “noise”). The simulations shown here were performed with a value of -5 dB for this ratio, which was the edge of stability for this setup, and hence certainly relevant. Experiments show a similar performance difference between the algorithms for other ratios. The STP frame length is 160 samples, the LTP frame overlap is 80 samples, the minimum and maximum lag for the LTP are 20 and 160 respectively. The sampling frequency is 8 kHz. The STP model order in PEM-AFROW and PEM-AF is 10. The forward delay is 200 taps in both PEM-AFROW and PEM-AF (note that the PEM-AF version of [11] does not explicitly incorporate a forward delay, but the theoretical analysis of [9] shows that this is required for correct performance, hence we added it to the system). PEM-AF and PEM-AFROW are seen to yield comparable performance, which is to be expected since the assumption of stationarity of the speech model during the data window for acoustic path identification is fulfilled in this experiment. The difference with PEM-AFROW including LTP is insignificant (and hence this curve is not plotted). Direct identification is seen to give poor results (i.e., when the room impulse response is identified as if the system were operating in open loop).

For longer paths, as in PA applications, PEM-AF performance decreases with path length, and for, e.g.,  $L_F = 1000$ , its behavior is certainly better than the direct identification behavior, as shown in Fig. 9.7, but a large bias remains.

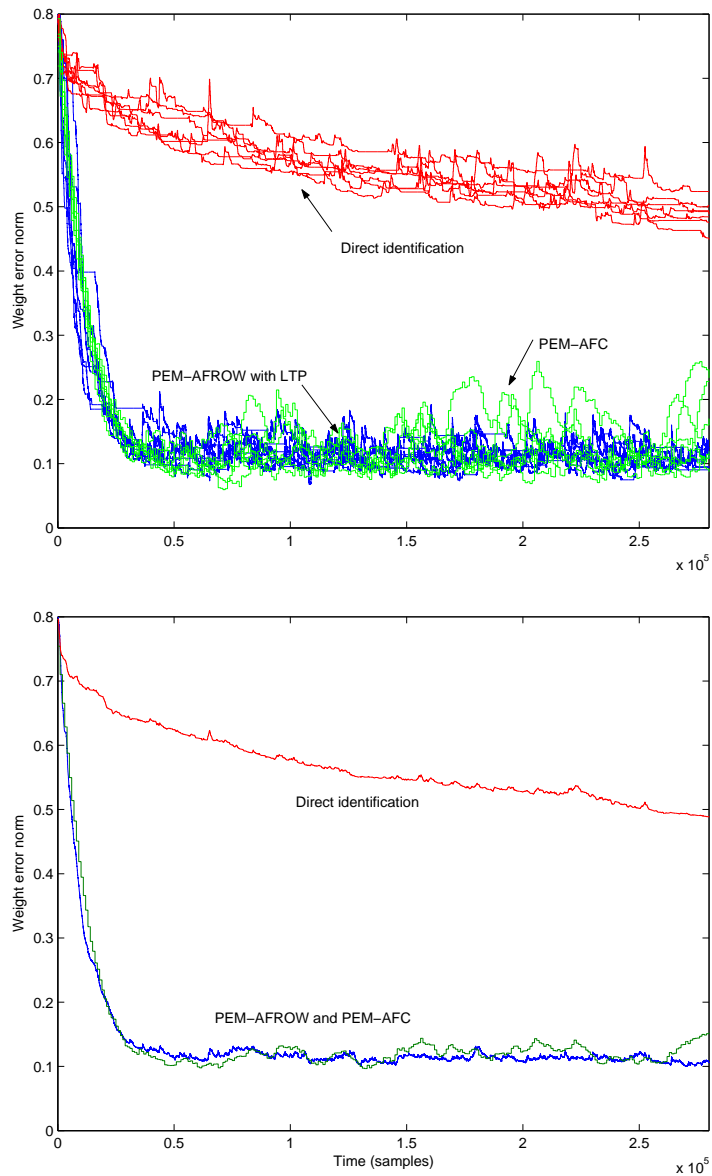


Figure 9.6: Closed-loop identification for short acoustic paths (50 taps). Upper figure: weight error norm versus time for 7 realizations with different speech signals for PEM-AFROW, PEM-AF and direct identification. Lower figure: the averages of the convergence curves of the realizations. PEM-AFROW performs equally well as PEM-AF for short paths (as expected). PEM-AFROW with LTP does not perform better than without LTP.



Also here the shown convergence curve of PEM-AF was tuned to have equal initial convergence to PEM-AFROW. PEM-AFROW on the other hand, does perform well also for long paths. The poor performance of PEM-AF is to be attributed to the stationary speech model assumption, which is not fulfilled for long paths.

## 9.8 Conclusion

We have introduced a new algorithm, referred to as PEM-AFROW, which allows for acoustic feedback cancellation in setups with long acoustic paths. It uses a speech source model with short- and long-term prediction. Not only the howling phenomenon is suppressed but also the reverberation-like sounds, which indeed become audible in the region of marginal stability. The main difference with existing schemes is twofold. First, our algorithm incorporates a long-term prediction filter which removes periodicity in the short-term speech signal residual, and second, we do not assume stationarity of the speech signal over the length of the data window on which the acoustic path is identified.

PEM-AFROW hence performs very well for long acoustic paths, contrarily to existing algorithms which were developed for short path applications. For short acoustic paths the performance of PEM-AFROW is equal to that of the existing methods. Thanks to the low complexity, the algorithm can easily be implemented in real time.

## Bibliography

- [1] M. P. Lewis, T. J. Tucker, and D. M. Oster, "Method and apparatus for adaptive audio resonant frequency filtering," U.S. Patent 5,245,665, Sep., 1993.
- [2] R. Porayath and D. J. Mapes-Riordan, "Acoustic feedback elimination using adaptive notch filter algorithm," U.S. Patent 5,999,631, Dec., 1999.
- [3] C. P. Janse and H. J. W. Belt, "Sound reinforcement system having an echo suppressor and loudspeaker beamformer," WIPO Patent Application WO/2003/010996, Feb., 2003.
- [4] G. W. Elko and M. M. Goodwin, "Beam dithering: Acoustic feedback control using a modulated-directivity loudspeaker array," in *Proc. 1993 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '93)*, vol. 1, Minneapolis, MN, USA, Apr. 1993, pp. 173–176.

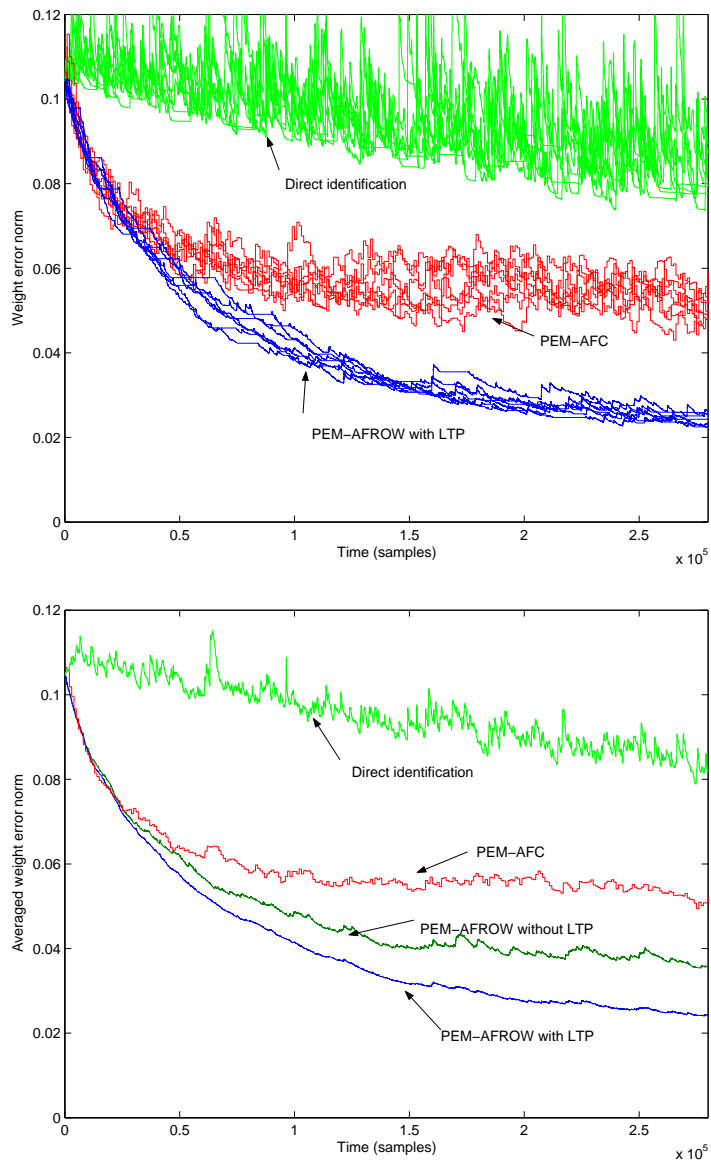


Figure 9.7: Identification for long acoustic paths (1000 taps). Upper figure: weight error norm versus time for 7 realizations with different speech signals for PEM-AFROW, PEM-AF and direct identification. PEM-AFROW without LTP is not shown to preserve the clarity of the figure. PEM-AFROW clearly outperforms the others. Lower figure: the averages of the weight error norms for the realizations.

- [5] S. Kamerling, K. Janse, and F. van der Meulen, "A new way of acoustic feedback suppression," in *Preprints AES 104th Convention*, Amsterdam, The Netherlands, May 1998, AES Preprint 4735.
- [6] C. P. Janse and P. A. A. Timmermans, "Signal amplifier system with improved echo cancellation," U.S. Patent 5,748,751, May, 1998.
- [7] M. G. Siqueira and A. Alwan, "Steady-state analysis of continuous adaptation in acoustic feedback reduction systems for hearing-aids," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 4, pp. 443–453, Jul. 2000.
- [8] J. Hellgren and U. Forssell, "Bias of feedback cancellation algorithms in hearing aids based on direct closed loop identification," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 7, pp. 906–913, Nov. 2001.
- [9] A. Spriet, I. Proudler, M. Moonen, and J. Wouters, "Adaptive feedback cancellation in hearing aids with linear prediction of the desired signal," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3749–3763, Oct. 2005.
- [10] T. van Waterschoot, G. Rombouts, and M. Moonen, "On the performance of decorrelation by prefiltering for adaptive feedback cancellation in public address systems," in *Proc. 4th IEEE Benelux Signal Process. Symp. (SPS '04)*, Hilvarenbeek, The Netherlands, Apr. 2004, pp. 167–170. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/04-24.html>
- [11] R. Leber and A. Schaub, "Circuit and method for the adaptive suppression of an acoustic feedback," U.S. Patent 6,611,600, Aug., 2003.
- [12] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
- [13] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. Cambridge, Massachusetts: MIT Press, 1986.
- [14] R. P. Ramachandran and P. Kabal, "Pitch prediction filters in speech coding," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-37, no. 4, pp. 467–478, Apr. 1989.



## Chapter 10

# Robust and Efficient AFC Implementation

Robust and efficient implementation of the  
PEM-AFROW algorithm for acoustic feedback  
cancellation

Geert Rombouts, Toon van Waterschoot, and Marc Moonen

Published in *J. Audio Eng. Soc.*, vol. 55, no. 11, Nov. 2007,  
pp. 955–966.

### Contributions of second author

- co-analysis of acoustic feedback problem
- co-development of robust implementation techniques
- co-development of subband and frequency domain implementation
- co-design of evaluation experiments
- co-interpretation of simulation results
- co-formulation of conclusion
- text co-editing and co-revision

## Abstract

The PEM-AFROW algorithm has been developed to model a room impulse response under closed-loop conditions such as in microphone-amplifier-loudspeaker systems, and is particularly suited for a speech signal input. If a controller is inserted in the loop which uses this model to remove the loudspeaker signal component from the microphone signal, an efficient feedback cancellation can be performed. In this paper, we address crucial implementation-oriented issues which have not been studied so far. In particular, we propose adaptation control mechanisms for feedback cancellation based on PEM-AFROW, and we provide efficient subband and frequency domain implementations for this algorithm.

## 10.1 Introduction

In closed-loop electro-acoustic systems such as public address (PA) systems or hearing aids, system instability due to acoustic feedback (the so-called Larsen-effect [1], or *howling*) is a well-known phenomenon. It occurs when the system loop gain exceeds unity on frequencies  $\omega_i$  where the loop phase equals  $2n\pi$  radians, with  $n$  integer. An *acoustic feedback canceller* is a controller for this system, which attempts to maintain stability, resulting in an “added gain”. This performance measure is the difference between the maximum gain for a stable controlled system and the maximum gain for a stable uncontrolled system.

The most important disadvantage of traditional acoustic feedback solutions [2],[3], mainly based on the insertion of notch filters centered on  $\omega_i$ , is that they are *reactive* (the instability first has to occur and be detected before the notch filter can be designed, and typically this detection takes up to 0.5 s). Notch filters in the signal path also result in signal distortion. They cannot remove the reverberation-like sound which occurs in marginally stable systems. On the other hand, a major advantage is that the notch filtering technique can be made quite robust against fast changes in the acoustic environment: typically, a small displacement of a microphone will only modestly affect the center frequencies on which instability occurs. This means that by using wider notch filters, robustness can be achieved.

In [4],[5], an alternative acoustic feedback cancelling method is derived which is based on the so-called PEM-AFROW algorithm (which is a Prediction Error Method Adaptive Filter using matrix-ROW operations as a preprocessing step). This method belongs to a class of adaptive feedback cancellation algorithms which are based on adaptive filtering and are similar to standard acoustic echo cancelling [6]-[10]. Fig. 10.1 provides a diagram, where  $K$  is the gain,  $q^{-1}$  is

the delay operator, hence  $q^{-d}$  is a delay of  $d$  samples. The speech source signal  $v(t)$  is assumed to be a time-variant autoregressive (TVAR) process, and is modeled as the output of an IIR filter

$$H(q, t) = \frac{1}{1 + a_1(t)q^{-1} + \dots + a_{n_A}(t)q^{-n_A}} \quad (10.1)$$

driven by white noise  $e(t)$ , hence  $v(t) = H(q, t)e(t)$ . Because of the closed signal loop, standard adaptive filters such as used in acoustic echo cancellation are found to provide biased solutions [10]. However, the PEM-AFROW algorithm allows for an unbiased joint identification of a linear model  $\hat{\mathbf{f}}(t)$  of the room impulse response (RIR)  $\mathbf{f}(t)$ , and of the TVAR coefficients  $\hat{\mathbf{a}}(t)$  of the denominator of  $H(q, t)$ , under closed-loop conditions (see also Fig. 10.1). Here  $\hat{\mathbf{f}}(t)$  is a vector which contains estimates of the coefficients of the finite impulse response (FIR) filter  $\mathbf{f}(t)$ , modeling the RIR (acoustic feedback path)

$$\hat{\mathbf{f}}(t) = \begin{bmatrix} \hat{f}_0(t) \\ \hat{f}_1(t) \\ \vdots \\ \hat{f}_{n_F}(t) \end{bmatrix}, \quad (10.2)$$

$\hat{\mathbf{a}}(t)$  is a vector

$$\hat{\mathbf{a}}(t) = \begin{bmatrix} 1 \\ \hat{a}_1(t) \\ \vdots \\ \hat{a}_{n_A}(t) \end{bmatrix}, \quad (10.3)$$

and  $q^{-d}$  is a delay of  $d$  taps. In [11], it is shown that for  $d > n_A$ , both  $\mathbf{f}(t)$  and  $\mathbf{a}(t)$  can be uniquely estimated. For a real speech input signal, apart from a short-term predictor with coefficients  $\hat{\mathbf{a}}(t)$ , additionally a long-term predictor is used, but for simplicity we will ignore this in this paper (details are in [5]). The estimated RIR  $\hat{\mathbf{f}}(t)$  can then be used in a so-called ‘‘controller’’  $\hat{\mathbf{f}}_0(t)$ , where the loudspeaker signal (far-end signal)  $u(t)$ , filtered by the model of the RIR (the controller), is subtracted from the microphone signal  $y(t)$ .

The resulting signal  $d(t)$  ideally has no remaining component stemming from the loudspeaker, but only contains the so-called ‘‘near-end signal’’  $v(t)$ . The echo canceller terminology ‘‘near-end’’ and ‘‘far-end’’ will also be used in this paper to denote the corresponding signals (far-end for the loudspeaker signal, and near-end for the speech signal), although the description ‘‘far-end’’ does not strictly fit into the acoustic feedback cancellation context.

The PEM-AFROW algorithm [4],[5] operates in two steps, Fig. 10.2. In a first step, the estimate  $\hat{\mathbf{f}}(t)$  is assumed to be correct, and it is used to find a linear prediction error filter  $\hat{\mathbf{a}}(t)$  that whitens the speech source signal. In a second step, this linear prediction error filter is in turn assumed to be correct, and it



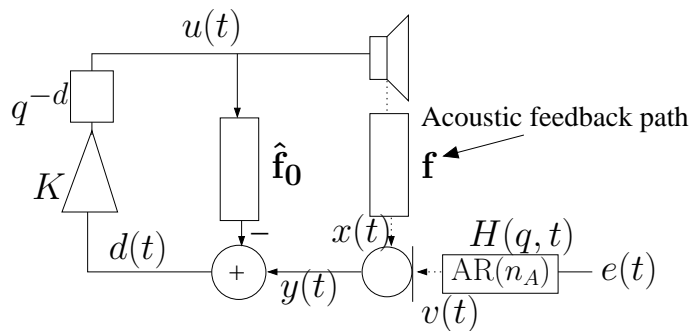


Figure 10.1: Acoustic feedback cancellation.

is used to perform a whitening of the speech source component in the input and desired signal for an adaptive filter which then provides an update of the estimate for  $\hat{\mathbf{f}}(t)$ . The linear prediction filter  $\hat{\mathbf{a}}(t)$  is –after convergence– equal to the inverse of the speech source model  $H(q, t)$ , and hence the adaptive filter  $\hat{\mathbf{f}}(t)$  then indeed operates on whitened versions of the signals  $u(t)$  and  $y(t)$  as its input and desired signal, respectively. It is easily shown that from the point of view of the adaptive filter  $\hat{\mathbf{f}}(t)$ , this scheme with a TVAR input and whitening filters as described above, corresponds to a scheme without whitening filters, but driven by a white noise signal. For a white noise input, decorrelation between  $u(t)$  and  $v(t)$  is achieved when  $d \geq 1$ . Hence due to the decorrelation provided by the delay, the adaptive filter  $\hat{\mathbf{f}}(t)$  will converge to an unbiased solution.

The advantages of using such a scheme are obvious. It is *proactive* (no instability has to occur before feedback path identification can take place), and the RIR is continuously tracked. Secondly, no (notch) filtering occurs in the signal path, which means that in case perfect identification is achieved, no signal distortion occurs. Finally, this scheme also removes the “reverberating” sound which occurs near the edge of stability.

In this paper, we address crucial implementation-oriented issues which have not been studied so far. We will focus on some robustness issues (Section 10.2) which exist in the context of PEM-AFROW-based acoustic feedback cancellation. In particular, we will study the topics adaptation control (Section 10.2.1), the use of prior knowledge (Section 10.2.2), the use of a foreground/background filter pair (Section 10.2.3), proactive notch filtering (Section 10.2.4), using nonlinearities (Section 10.2.5), and finally the inclusion of a highpass filter (Section 10.2.6). Section 10.3 deals with subband and frequency domain implementations of PEM-AFROW. Section 10.4 provides simulation results, and Section 10.5 gives the overall conclusions.

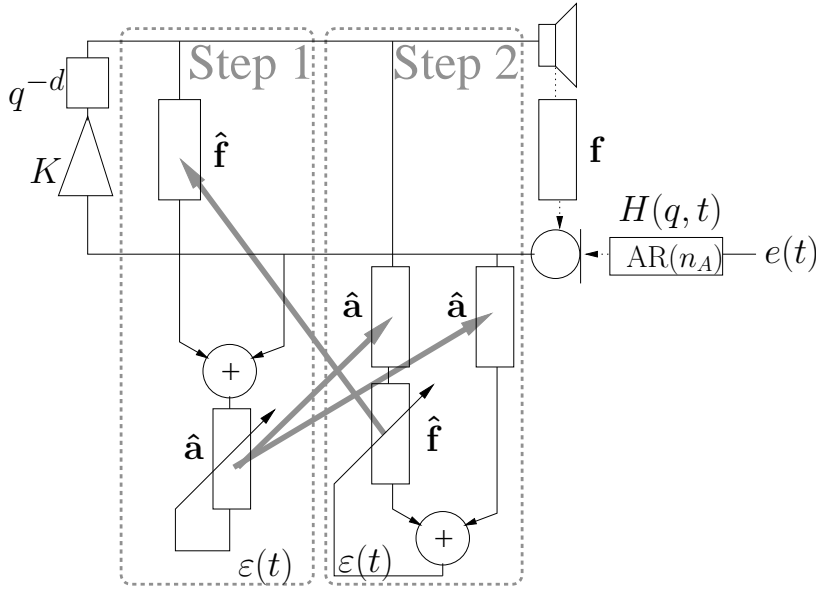


Figure 10.2: PEM-AFROW identification algorithm.

## 10.2 Robustness

The controller in a PEM-AFROW-based acoustic feedback cancellation scheme stabilizes an unstable system (loudspeaker-room-microphone-amplifier). The controller  $\hat{\mathbf{f}}_0(t)$  should be matched to the acoustic feedback path  $\mathbf{f}(t)$  at all times to avoid instabilities. Fast changes in the environment can often not be tracked fast enough by the adaptive filter and hence several robustness issues arise. These will be addressed in the following sections.

### 10.2.1 Adaptation Control

#### Problem Statement

In the context of standard acoustic echo cancellation, a near-end signal forms a disturbance (measurement noise) for the identification process performed by the adaptive filter. If the near-end signal  $v(t)$  is *correlated* with the far-end signal  $u(t)$ , this will result in a *bias* in the estimate of the filter coefficients  $\hat{\mathbf{f}}(t)$ . If the signals are *uncorrelated*, still *variance* on  $\hat{\mathbf{f}}(t)$  will result due to the measurement noise. When NLMS-based algorithms are used (as is often the case because of their low complexity), an *excess mismatch* on  $\hat{\mathbf{f}}(t)$  is introduced in the presence of near-end signals. Hence a *double-talk detector* is used to switch off adaptation during periods where near-end speech is present in order

to avoid variance on the estimate of the RIR, and a *far-end activity detector* is used to switch off adaptation when the far-end signal energy is below the noise level.

In the acoustic feedback cancellation context, see Fig. 10.1, this far-end activity detector is also required (adaptation may only occur when a signal is present), but a double-talk detector is obviously irrelevant. The loudspeaker signal is correlated with the near-end signal due to the presence of the forward path  $Kq^{-d}$ , hence the feedback cancellation system continuously operates in double-talk mode. In order to be able to perform continuous adaptation when the far-end (and hence also near-end) signals are present, the PEM-AFROW algorithm decorrelates  $u(t)$  and  $v(t)$  by the whitening procedure combined with a delay in the forward path. This effectively avoids a bias in the estimate. Due to the continuous adaptation during signal presence, convergence speed in a real-world scenario, is fast enough to turn the audio volume level from  $-\infty$  dB to 5 to 10 dB added gain in 5 seconds without howling occurring. Note that in this case 0 dB is defined as the level at which in the system without a feedback controller howling would start to occur. The PEM-AFROW algorithm avoids the bias, but still a large variance may result. The amount of variance depends on the ratio FNR between the two components in the microphone signal  $y(t)$ : the far-end component  $x(t)$  and the near-end component  $v(t)$ . We define

$$\text{FNR} = \frac{E\{x^2(t)\}}{E\{v^2(t)\}}, \quad (10.4)$$

where  $E\{\cdot\}$  is the expectation operator. The near-end component can be seen as measurement noise for the identification process, while the far-end component in the microphone signal is indeed the loudspeaker signal filtered by the RIR  $\mathbf{f}(t)$ , and so represents the signal part which is useful for the identification process. Hence for a large value of the FNR, the variance will be low. For high gains in the forward path of an acoustic feedback application, the identification works better, which is of course useful since the Larsen-effect occurs at high gains. On the other hand, in several scenarios the FNR may be low (see below), and this will lead to a large variance on the estimate of the RIR, which may then result in instability of the system (Larsen-effect).

### Onset Detection

Instabilities primarily occur at speech onsets, or in general at sudden level increases of the near-end signal. At a speech onset (a sudden level increase of  $v(t)$ ), the FNR is temporarily very small because the corresponding level increase in  $x(t)$  is delayed by the forward path  $q^{-d}$  and the delay due to the RIR  $\mathbf{f}(t)$  itself, see Fig. 10.3. Hence if no control algorithm is incorporated in the system, onsets of speech may cause the estimate of the RIR to drift away from the correct solution. This will often lead to instability of the whole system.

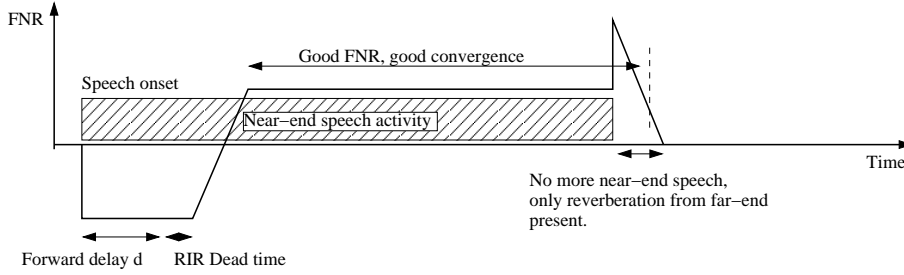


Figure 10.3: Timeline of the far-end near-end ratio, starting from a near-end speech onset.

The control algorithm we propose is based on the observation that a level increase in  $y(t)$  is always followed by a time interval with low FNR, until the level in  $x(t)$  increases correspondingly. If an increase in  $y(t)$  (e.g., a speech onset) is detected, adaptation is switched off for a predefined time interval  $\Delta$ . It is important that the onset is detected instantaneously, and hence time averaging to obtain an energy estimate of  $y(t)$  is impossible, because the lag introduced by the averaging would already allow the adaptive filter coefficients to drift substantially.

In order to achieve instantaneous detection, a Gaussian distribution of the (whitened) near-end signal  $\varepsilon(t)$  is assumed, and the short-term variance  $\sigma_\varepsilon^2(t)$  of this distribution is estimated over an exponential window

$$\sigma_\varepsilon^2(t) = \lambda_\varepsilon \sigma_\varepsilon^2(t-1) + (1 - \lambda_\varepsilon) \varepsilon^2(t), \quad (10.5)$$

with  $0 \ll \lambda_\varepsilon < 1$  a forgetting factor. A speech onset is detected at time  $t_0$  when an (instantaneous) value of  $\varepsilon(t)$  occurs which is larger than  $c\sigma_\varepsilon(t)$ , with  $c$  a constant.

If an onset is detected, adaptation is disabled from time  $t_0$  to time  $t_0 + \Delta$ . A conservative setting for  $\Delta$  would be the sum of the forward delay and the number of taps in the RIR. Sometimes most energy of the RIR is concentrated in the beginning of the RIR, and  $\Delta$  can be taken smaller: it is important that after the delay  $\Delta$ , the FNR assumes a sufficiently large value.

### 10.2.2 Prior Knowledge of the Feedback Path

The near-end speech source signal takes on the role of “measurement noise” for the identification process of the RIR by an adaptive filter. The PEM-AFROW algorithm contains an adaptive filter, which identifies the RIR, and a prewhitening stage, which whitens the measurement noise. The whitened measurement noise will result in a variance on the estimate of the room impulse response.

In order to provide robustness, the cost function can be modified to incorporate prior knowledge. This will lead to regularized versions [12],[13] of the adaptive filtering algorithm, in which the variance on the estimate of the RIR is reduced at the expense of a (hopefully) minor bias. We now introduce other regularized versions of the NLMS algorithm starting from a Mean Square Error (MSE) cost function:

$$V[\hat{\mathbf{f}}(t), t] = E\{\varepsilon^2(t)\} \quad (10.6)$$

$$= E\{[\tilde{y}(t) - \tilde{\mathbf{u}}^T(t)\hat{\mathbf{f}}(t)]^2\}, \quad (10.7)$$

with  $\tilde{y}(t) = \hat{\mathbf{a}}^T(t)\mathbf{y}(t)$  the whitened microphone signal where

$$\mathbf{y}(t) = \begin{bmatrix} y(t) \\ y(t-1) \\ \vdots \\ y(t-n_A) \end{bmatrix} \quad (10.8)$$

and

$$\tilde{\mathbf{u}}(t) = \begin{bmatrix} \tilde{u}(t) \\ \tilde{u}(t-1) \\ \vdots \\ \tilde{u}(t-n_F) \end{bmatrix} \quad (10.9)$$

with  $n_F + 1$  the adaptive filter length and  $\tilde{u}(t) = \hat{\mathbf{a}}^T(t)\mathbf{u}(t)$  the whitened loudspeaker signal where

$$\mathbf{u}(t) = \begin{bmatrix} u(t) \\ u(t-1) \\ \vdots \\ u(t-n_A) \end{bmatrix} \quad (10.10)$$

From this minimization problem, the well-known LMS or stochastic gradient algorithm is derived:

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) - \bar{\mu} \left. \frac{\partial V[\hat{\mathbf{f}}, t]}{\partial \hat{\mathbf{f}}} \right|_{\hat{\mathbf{f}}=\hat{\mathbf{f}}(t-1)} \quad (10.11)$$

$$= \hat{\mathbf{f}}(t-1) + \bar{\mu} E\{\tilde{\mathbf{u}}^T(t)\varepsilon(t)\} \quad (10.12)$$

$$\approx \hat{\mathbf{f}}(t-1) + \bar{\mu}\tilde{\mathbf{u}}(t)\varepsilon(t), \quad (10.13)$$

where  $\bar{\mu}$  is the step size parameter. With a normalized step size,

$$\bar{\mu}(t) = \frac{\mu}{\tilde{\mathbf{u}}^T(t)\tilde{\mathbf{u}}(t)}, \quad (10.14)$$

the update formula for Normalized LMS (NLMS) is obtained as

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mu \frac{1}{\tilde{\mathbf{u}}^T(t)\tilde{\mathbf{u}}(t)} \tilde{\mathbf{u}}(t)\varepsilon(t). \quad (10.15)$$

If a good estimate  $\bar{\mathbf{f}}$  for  $\hat{\mathbf{f}}(t)$  has been obtained off-line (e.g., by a measurement of the impulse response at system setup, or by an off-line procedure based on, e.g., RLS, which does not exhibit excess mismatch as NLMS does, but which is computationally more expensive), then an alternative cost function can be defined:

$$V[\hat{\mathbf{f}}(t), t] = E \left\{ \varepsilon^2(t) + \beta \|\hat{\mathbf{f}}(t) - \bar{\mathbf{f}}\|^2 \right\}. \quad (10.16)$$

Here  $\beta$  is a parameter which can be used to weight the importance of both terms. Along the lines of the derivation of the LMS and NLMS algorithms, regularized LMS and NLMS algorithms can be obtained. The regularized LMS update is

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \bar{\mu} (\bar{\mathbf{u}}(t)\varepsilon(t) - \beta[\hat{\mathbf{f}}(t-1) - \bar{\mathbf{f}}]) \quad (10.17)$$

$$= (1 - \beta\bar{\mu})\hat{\mathbf{f}}(t-1) + \bar{\mu}[\bar{\mathbf{u}}(t)\varepsilon(t) + \beta\bar{\mathbf{f}}], \quad (10.18)$$

from which the regularized NLMS update follows by choosing  $\bar{\mu}$  to minimize the a posteriori error, and then introducing an additional step size parameter,

$$\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mu \frac{\varepsilon(t)}{\bar{\mathbf{u}}^T(t)\bar{\mathbf{u}}(t)} \bar{\mathbf{u}}(t) \quad (10.19)$$

$$\bar{\mathbf{u}}(t) = \bar{\mathbf{u}}(t)\varepsilon(t) - \beta[\hat{\mathbf{f}}(t-1) - \bar{\mathbf{f}}]. \quad (10.20)$$

Experiments show that using the update formula (10.19) instead of (10.15) in the RIR identification part of the PEM-AFROW algorithm is especially useful at system startup. So at startup, a relatively large  $\beta$  can be used, which can then be decreased gradually. A similar approach was used in [14].

### 10.2.3 Foreground/Background Filter

In order to minimize the variance on the estimate and (for NLMS algorithms) the excess mismatch due to the presence of the (whitened) speech source signal (measurement noise), a small step size should be chosen. While this provides robustness against measurement noise, it results in slow convergence. Under variations in the RIR however, fast tracking is required, in order to avoid instability.

In the acoustic echo cancellation context, often a twin filter structure is used [15],[16] to overcome problems in double-talk situations. A similar idea can now be applied in acoustic feedback cancellation. Two adaptive filters estimate the RIR from the whitened microphone signal and the whitened loudspeaker signal. The so-called background adaptive filter  $\hat{\mathbf{f}}_B(t)$  is operated with a larger step size ( $\mu_B$ ) than the foreground filter  $\hat{\mathbf{f}}(t)$  (step size  $\mu$ ). For example:  $\mu_B = 10\mu$ . The residual energies  $E\{\varepsilon_B^2(t)\}$  of the whitened background filter residuals and  $E\{\varepsilon^2(t)\}$  of the whitened foreground filter residuals are estimated as  $\sigma_{T,\varepsilon_B}^2(t)$

and  $\sigma_{T,\varepsilon}^2(t)$ , respectively, over an exponential window,

$$\sigma_{T,\varepsilon}^2(t) = \lambda_T \sigma_{T,\varepsilon}^2(t-1) + (1 - \lambda_T) \varepsilon^2(t), \quad (10.21)$$

$$\sigma_{T,\varepsilon_B}^2(t) = \lambda_T \sigma_{T,\varepsilon_B}^2(t-1) + (1 - \lambda_T) \varepsilon_B^2(t), \quad (10.22)$$

and similarly the energy  $E\{\tilde{y}^2(t)\}$  of the whitened microphone signal  $\tilde{y}(t) = \hat{\mathbf{a}}^T(t)\mathbf{y}(t)$  is estimated as

$$\sigma_{T,\tilde{y}}^2(t) = \lambda_T \sigma_{T,\tilde{y}}^2(t-1) + (1 - \lambda_T) \tilde{y}^2(t). \quad (10.23)$$

Here  $0 \ll \lambda_T < 1$  is a forgetting factor. At discrete time intervals  $t = n\Delta_T$  with  $n$  integer, these energy estimates are compared, and the coefficients  $\hat{\mathbf{f}}_B(t)$  are copied to  $\hat{\mathbf{f}}(t)$  if the condition

$$\sigma_{T,\varepsilon_B}^2(n\Delta_T) < \gamma_1 \sigma_{T,\varepsilon}^2(n\Delta_T) \quad (10.24)$$

is met (optionally an additional condition

$$\sigma_{T,\varepsilon_B}^2(n\Delta_T) < \gamma_2 \sigma_{T,\tilde{y}}^2(n\Delta_T) \quad (10.25)$$

can be imposed). If on the other hand

$$\sigma_{T,\varepsilon_B}^2(n\Delta_T) > \gamma_3 \sigma_{T,\varepsilon}^2(n\Delta_T), \quad (10.26)$$

the coefficients  $\hat{\mathbf{f}}(t)$  are copied to  $\hat{\mathbf{f}}_B(t)$ , to avoid divergence of  $\hat{\mathbf{f}}_B(t)$ . In these conditions,  $\gamma_{1,2,3}$  are parameters with  $0 < \gamma_{1,2,3} < 1$  and  $\gamma_1 \leq \gamma_3$ .

The above procedure effectively combines the advantages of a large step size (fast convergence in case of a large change in the room impulse response) and of a small step size (low-variance steady-state behaviour). This strategy can be applied in a time domain NLMS approach, in a subband approach (see Section 10.3.1), as well as in a frequency domain NLMS approach (Section 10.3.2).

## 10.2.4 Notch Filtering

### Problem Statement

In the PEM-AFROW algorithm, an unbiased estimate  $\hat{\mathbf{f}}(t)$  of the RIR is formed. The loudspeaker signal  $u(t)$  is filtered by this RIR model, and the result is subtracted from the microphone signal  $y(t)$ , effectively cancelling the transfer function  $\mathbf{f}(t)$  between loudspeaker and microphone. This means that essentially no signal distortion occurs in this setup (assuming a perfect estimate), more specifically that the signal is not attenuated in certain frequency regions, and that the gain level  $g$  which is set on the amplifier, is effectively obtained, *even when this means that the corresponding loop gain is larger than unity*. An obvious issue is that if the tracking behaviour of the PEM-AFROW algorithm is non-perfect, an unstable system may result as soon as the RIR changes.

Traditional approaches to feedback cancellation in PA applications use notch filtering. Disadvantages of this approach are the reactive character and the fact that signal distortion occurs: the gain is reduced in frequency regions where howling is likely to occur, which not only gives the signal an audible distortion, but which also results in the fact that the gain level which is set on the amplifier, is not reached whenever the system becomes unstable.

PEM-AFROW was shown to be very robust against changes in the acoustic feedback path due to moving objects. These changes have the largest influence on the RIR (and require a good tracking behaviour of the algorithm) when they occur in the vicinity of the loudspeaker or microphone, e.g., head movements of the speaker. Experiments were performed with a real-time system with loudspeaker and microphone spaced 2 to 3 meters apart, in an office room environment, and it was shown that howling cannot be induced by fast head movements, by partially covering up the microphone or loudspeaker, nor by moving large objects in the space between loudspeaker and microphone. Hence for these cases robustness is guaranteed by the PEM-AFROW algorithm.

On the other hand, when the microphone or loudspeaker position is changed, the RIR “shifts” over the time axis. This is a change of the RIR which the PEM-AFROW algorithm can not track fast enough: it is less robust than notch-filtering-based techniques when the loudspeaker or microphone itself is moved.

### Proactive Notch Filtering

We have simulated a RIR [17] for a room of  $10 \times 5 \times 4$  m with reflection coefficients of the walls equal to 0.8, with a loudspeaker on position (3.21, 3.21, 2.5) m and a microphone on positions (1, 2, 1) m and (1, 1.95, 1) m, hence 5 cm apart. The RIR for the first microphone position and the difference between the RIRs for both microphones is plotted in Fig. 10.4. The difference has the same order of magnitude as the impulse responses themselves. It is obvious that a PEM-AFROW-based approach, which has to track the impulse response, may need a considerable amount of time before it has compensated for the difference between the impulse responses. On the other hand, the position of many of the peaks in the loop gain frequency spectrum shown in Fig. 10.5, does not change much, which indicates that a notch-filtering-based approach can be more robust against the microphone displacement. Depending on the room acoustics, only a limited number of modes is available that can give rise to instabilities, and so a small change in the distance between loudspeaker and microphone may lead to a small change in the center frequencies of the oscillations (unless a completely different mode becomes active), while it can lead to large changes in the RIR, see Fig. 10.4. Since notch filtering is robust against small changes in the center frequencies, it is desirable to combine the PEM-AFROW provided estimates of the RIR (proactive) with the notch filtering technique (robust).



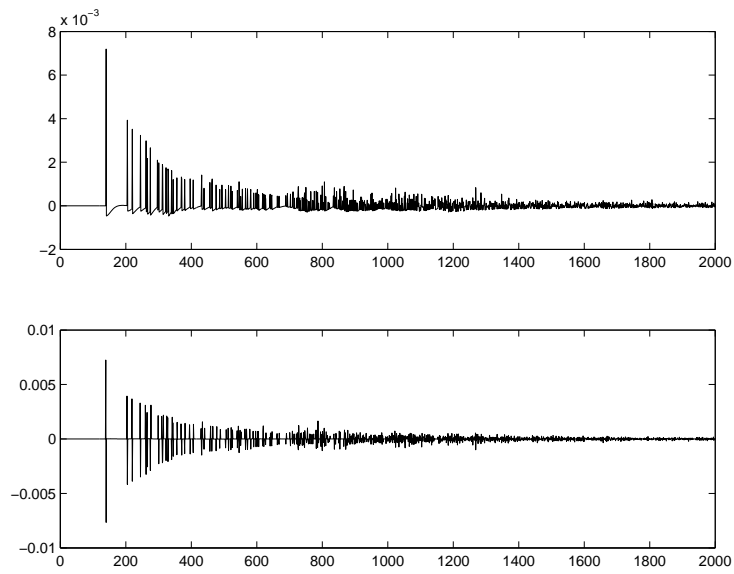


Figure 10.4: RIR vs. time index for first microphone position, and the (large) RIR difference between the two microphones.

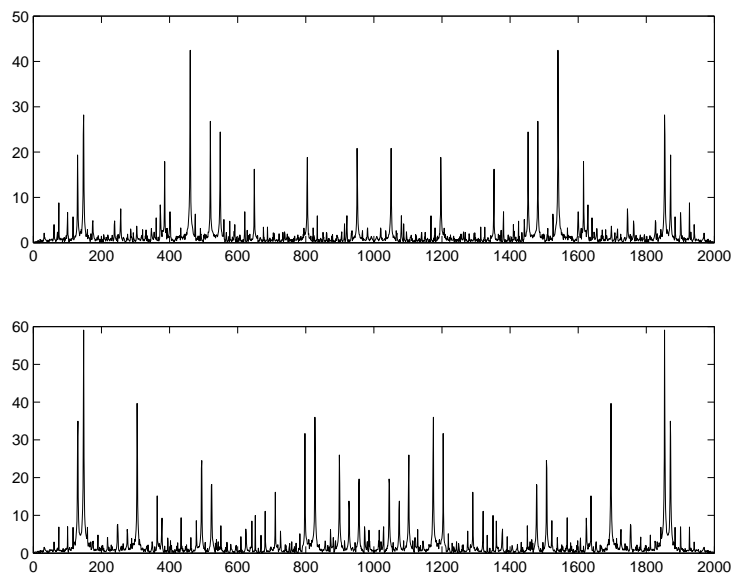


Figure 10.5: Loop gain vs. frequency index for two slightly different microphone positions (10 cm apart).

Table 10.1: Notch filter design, starting from the PEM-AFROW room impulse response estimate.

---

```

1. for  $i = 1, \dots, \text{NrActive}$ 
    if  $G_l(\omega_i) < T_{\min}$ 
        decrease depth on  $\omega_i$ 
    end if
end for
2. while maxima can be found
    (a) find  $\omega_{\max} = \arg \max_{\omega} G_l(\omega)$  where  $\omega_{\max} > T_{\max}$ 
    (b) if  $\omega_{\max}$  close to existing notch  $\omega_F$ 
         $\omega_F = \omega_{\max}$ 
        increase depth on  $\omega_F$ 
    elseif  $\text{NrActive} < \text{Max}$ 
        new notch on  $\omega_{\max}$ 
    end if
end while

```

---

The condition for instability is fulfilled when the loop gain  $G_l(\omega) \geq 1$  on frequencies where the loop phase  $\varphi_l(\omega) = n2\pi$ , with  $n$  an integer number. The frequencies  $\omega_i$  for which this condition holds, can be found from the estimate of the RIR which is provided by PEM-AFROW, and the knowledge of the forward gain  $K$ :  $G_l(\omega) = K|\hat{F}(\omega)|$ , where  $|\hat{F}(\omega)|$  denotes the magnitude response of  $\hat{\mathbf{f}}$ .

The algorithm we have used to design the notch filters is given in Table 10.1. It places notch filters on frequencies  $\omega_i$  where  $G_l(\omega_i) > T_{\max}$  with threshold  $T_{\max} < 1$ , independently of the phase characteristic (although this would be easy to incorporate), and it removes notch filters on frequencies where  $G_l(\omega) < T_{\min}$ . The notch filter design procedure is repeated regularly (a few times per second), and each time only a limited number of notch filters are designed. Then the PEM-AFROW algorithm is allowed to adapt to the new acoustic impulse response *with the notch filters included*. This results in an update of  $G_l(\omega)$ . The notch filter depth on a certain frequency may then be increased or decreased, depending on the new value of  $G_l(\omega)$ .

The notch filters are IIR filters with two poles and two zeros each. From a numerical point of view, it is important *not* to combine all of the notch filters into one IIR filter, but to apply all of the so-called “biquads” individually to the signal, as otherwise numerical instability may result if the poles are close together. A design procedure for notch filters is described in [18].

At first sight it may seem advantageous not to include the notch filters in the

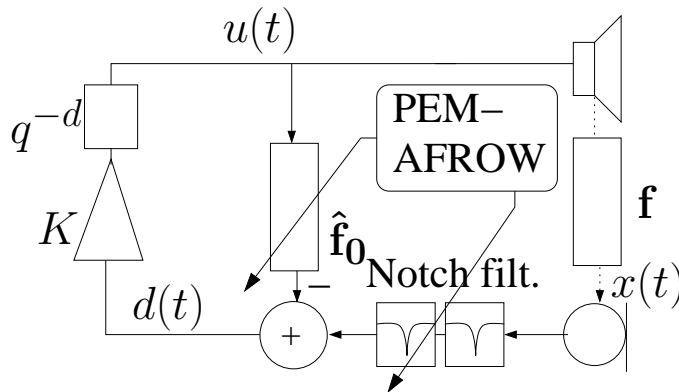


Figure 10.6: Combined system with proactive notch filtering and a controller  $\hat{\mathbf{f}}_0(t)$ .

feedback path (the part of the loop which is modeled by the PEM-AFROW algorithm), but instead place them in the forward path (the part containing the delay). That way, the PEM-AFROW adaptive filter would not have to model the changing notch filter characteristics. But on the other hand, the FNR would become very low on the notch frequencies when the notch filters are placed in the forward path (the loudspeaker signal is attenuated, while the near-end signal is not), and a large variance on the estimation of the loop gain would occur on the notch frequencies. Simulations show that this leads to wrong decisions on where to place (or remove) notch filters.

The notch-filtering-based acoustic feedback controller is now combined with a PEM-AFROW-based controller, as shown in Fig. 10.6. The PEM-AFROW algorithm estimates the cascade of the notch filters with the RIR and this estimate is then used in a controller  $\hat{\mathbf{f}}_0(t)$  which cancels the remaining loudspeaker component in the microphone signal. The advantage compared to the PEM-AFROW-only algorithm [4],[5] is that the notch filters –when chosen appropriately wide– guarantee robustness for path changes (in exchange for some distortion), and the controller removes the residual reverb-like sounds.

If  $T_{\max}$  is chosen smaller than 1, the above technique leads to a proactive system, because new notch filters will be designed before the system becomes unstable at the corresponding frequencies. The system can be made robust to path changes by choosing an appropriate width for the notch filters. The parameters for the notch filters should be chosen to avoid audible distortion. It is important to note that due to the additional PEM-AFROW-based controller  $\hat{\mathbf{f}}_0(t)$ , which removes residual loudspeaker components from the microphone signal, the loop gain on the frequencies  $\omega_i$  is not limited to  $T_{\max}$  as in traditional, notch-filtering-only techniques.

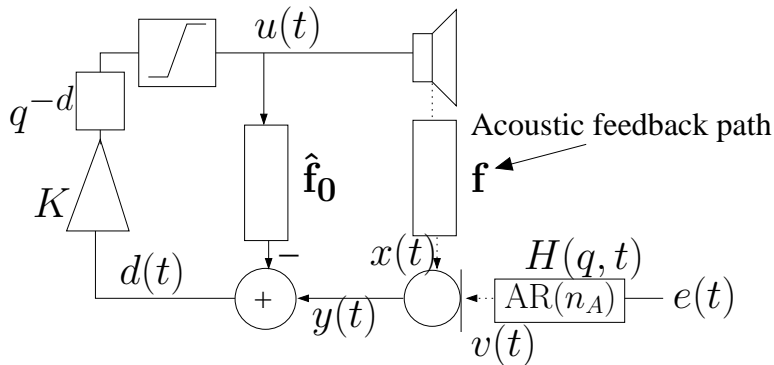


Figure 10.7: A limiter or clipper should be added to avoid clipping in the loudspeaker/amplifier.

### 10.2.5 Nonlinearities

If large signal levels are applied to the loudspeaker/amplifier (e.g., in case the Larsen-effect occurs), the loudspeaker or amplifier may exhibit nonlinear behavior (clipping). A PEM-AFROW-based acoustic feedback cancellation scheme provides a linear estimate of the transfer function for the system with input signal  $u(t)$  and output signal  $x(t)$  (see Fig. 10.1). This system comprises the amplifier (which is not explicitly drawn in the figure, but in practice is placed just before the loudspeaker), the loudspeaker, the acoustic feedback path and the microphone (we will refer to this system as the RIR). The linear estimate, which is used in the controller  $\hat{\mathbf{f}}_0(t)$ , will not be a correct model when nonlinearities due to loudspeaker or amplifier clipping occur in the system. Because of the mismatch between the (linear) controller and the (nonlinear) system, instabilities (howling) will occur.

A simple and efficient method to avoid clipping in the room impulse response is by limiting the maximum amplitude of  $u(t)$  as shown in Fig. 10.7, i.e., by adding a limiter or clipper on the left hand side of the controller, before its input.

In several applications, nonlinear operations are required in the signal path (e.g., dynamics processing). These operations should be applied before the input of both the controller and the RIR system, such that the identification algorithm does not see these nonlinearities. Note that also the gain control  $K$  should preferably be placed in the left hand part of the scheme, as shown in the figure, such that variations and nonlinearities (e.g., dynamics compression) in  $K$  do not have to be tracked by the adaptive algorithm.

### 10.2.6 Highpass Filtering

Often the low cut-off frequency of the loudspeaker or amplifier is higher than that of the microphone. This means that for low-frequency near-end signals (e.g., generated by displacing the microphone stand), the FNR may become very small. Due to the large variance on the estimation resulting from this, the system may become unstable. A simple solution is to use a highpass filter on the microphone signal with a cut-off frequency (depending on the loudspeaker characteristics) of, e.g., 100 Hz. The influence on the speech quality will be very small, while low-frequency near-end signals are suppressed.

## 10.3 Subband and Frequency Domain Implementations

In PA applications, typically long impulse responses are involved, and high sampling frequencies are desired. This leads to a high computational complexity in time domain algorithms. We propose two different frequency domain implementations. The first one (Section 10.3.1) is based on a subband implementation of the time domain algorithm. A filterbank is used to split the time domain signals into independent frequency bands, and processing is done independently in each of these bands. The advantage (apart from a complexity reduction) will be that the filter length can be chosen independently for each of the frequency bands. The downside of this approach is the latency introduced by the filterbank operation.

The second frequency domain implementation reduces this latency since it is based on the partitioned block frequency domain adaptive filtering (PBFDAF) algorithm (Section 10.3.2).

### 10.3.1 Oversampled DFT-Modulated Filterbank Implementation

We first propose an implementation based on a subband implementation of the NLMS algorithm. A DFT-modulated perfect reconstruction filterbank [19] divides the signals in subbands and then the PEM-AFROW-based algorithm is applied in each subband. This is advantageous from a performance and complexity point of view.

It is well known that the performance of the NLMS-based algorithms which run in each band is better when the eigenvalue spread of their input signal is smaller. Since for a typical speech signal the bandfiltered signals in the subbands are whiter than the fullband signal, the eigenvalue spread is indeed

smaller.

The complexity is also reduced compared to time domain processing: in an  $M$ -band filterbank with  $N$ -fold downsampling, the filter length of the adaptive filters in each band is roughly  $N$  times smaller than for the corresponding time domain setup, and the adaptive filters also operate at an  $N$  times lower sampling rate. The adaptive filters in only  $\frac{M}{2} + 1$  of the bands have to be computed, since the other bands are complex conjugated. So –neglecting the complexity of the filterbank operations– the complexity in the subband scheme is  $\frac{M}{2N^2}$  smaller compared to the corresponding time domain scheme.

In order to avoid aliasing, it is important to use an oversampled filterbank (which means that  $N < M$ ). The filterbank prototype filter will never have ideal properties (it is not a brick wall filter), and hence for  $M = N$ , frequency components above the Nyquist frequency would be present in the subband signals before the downsampling. The “perfect reconstruction” property of the filterbank would indeed compensate for the aliasing even in that case, but *only* if no processing occurs in the subbands. This is not the case in our scheme. So  $N$  should be chosen small enough, such that no aliasing occurs even with non-ideal bandpass filters in the filterbank.

In the subband processing context, complexity can even further be reduced thanks to the properties of the physical environment. It can be observed that for a typical room (or hall), the absorption of the high-frequency components is much higher than for the low-frequency components. We propose to estimate the maximum required number of filter taps for each subband individually in a calibration stage. It should be noted that calibration should be performed in an empty room, and that (due to absorption in the clothing) the impulse response of rooms is always shortened when people (the audience) enter the room (note that a similar approach could also be used in acoustic echo cancelling scenarios).

Fig. 10.8 shows the setup of this system. Straightforward implementation is shown in the upper part where a PEM-AFROW algorithm is performed in each of the subbands independently. In the lower part of Fig. 10.8, we show how one analysis filter bank can be removed by directly reusing the subband signals as inputs for the adaptive filters. Since the synthesis/analysis filterbank pair in the lower part of Fig. 10.8 introduces a delay  $q^{-d_0}$ , and the subband filters have to estimate the subband path between loudspeaker and microphone, this delay must also be included before the subband filters.

Since the individual PEM-AFROW algorithms in each of the subbands run at an  $N$  times lower sampling frequency, their parameters should be adjusted compared to the fullband version of the algorithm. This means that the AR order, the frame length and the forward delay are all  $N$  times smaller compared to the fullband version.

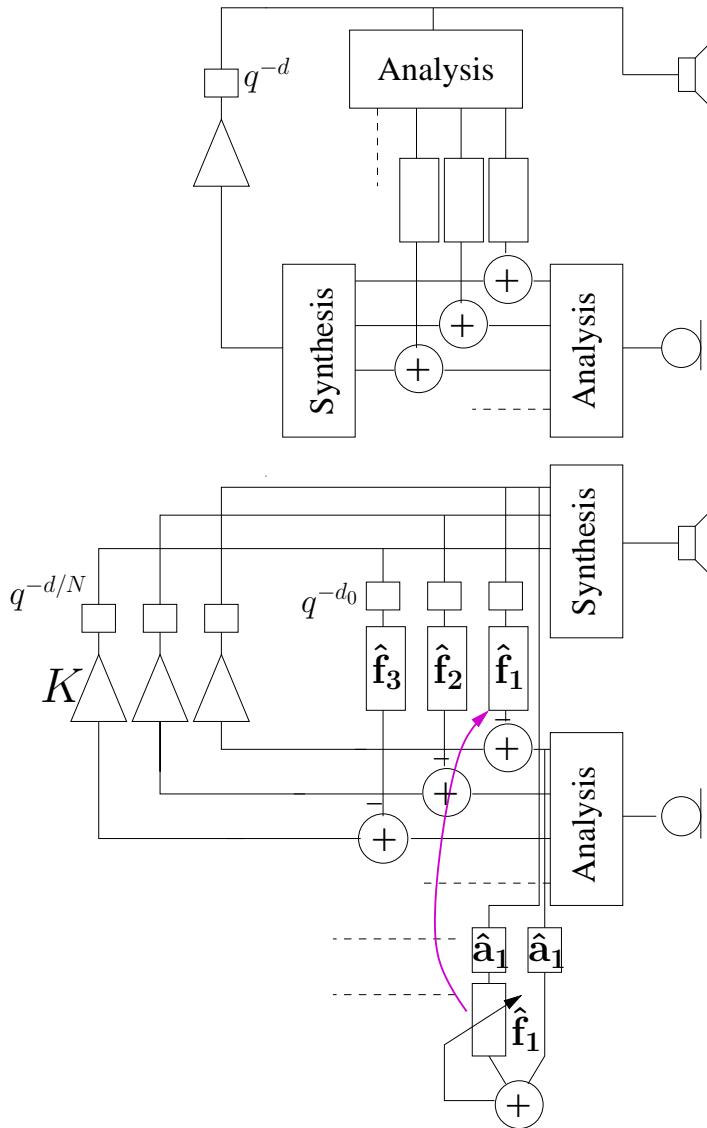


Figure 10.8: Filterbank implementation. Upper: straightforward implementation. Lower: implementation with one analysis bank less (identification part shown only for one subband).

The resulting algorithm, implemented in C++ without any specific optimization effort, runs in real time on a Pentium Mobile 1.7 GHz with a Linux operating system, and Jack<sup>1</sup> as the sound device interface, for a 16-band filterbank, with 2-fold oversampling, and with 600 filter taps per band.

### 10.3.2 Frequency Domain Implementation

A major disadvantage of the above filterbank approach, is the delay introduced by the filterbanks. This delay becomes larger when the number of subbands (and hence the complexity reduction) is larger. Although a certain amount of delay is required in the PEM-AFROW algorithm, the sum of a large filterbank delay and the delay due to the A/D-D/A buffering may become prohibitive for certain applications. In [20], the authors use a partitioned block frequency domain adaptive filtering algorithm (PBFDAF, also known as the MDF algorithm [21]) to overcome this disadvantage.

It should be noted though that in the PEM-AFROW algorithm, the input vector of the adaptive filter  $\hat{\mathbf{f}}$  has to be recalculated each time when the estimated AR coefficients  $\hat{\mathbf{a}}$  change. This amounts to a matrix-vector multiplication with dimension  $(n_F + 1) \times (n_A + 1)$ , which occurs at least every speech frame, but which may occur more often due to the nondeterministic nature of the long term predictor [5]. In an implementation with PBFDAF, this results in the FFTs of all of the input vector partitions having to be recalculated each time this occurs. The computational complexity for such a PBFDAF implementation will hence become comparable to the complexity of an FDAF implementation with FFT size equal to  $2 \times (n_F + 1)$  and a shift size equal to the frame length.

In practical applications, it is often advisable to use the PBFDAF or FDAF algorithm to achieve a low latency, instead of the subband approach of Section 10.3.1. PBFDAF and FDAF combined with PEM-AFROW both have comparable complexity, but the FDAF algorithm results in a more simple implementation.

## 10.4 Simulations

In this section, we will first focus on the individual benefits achieved by onset detection, the use of both a foreground and a background filter, and proactive notch filtering, and then we will review a combined system which incorporates all of the techniques described above.

---

<sup>1</sup><http://www.jackaudio.org>



### 10.4.1 Onset Detection

The simulation environment consists of an artificial RIR of 2000 taps (obtained with the mirror image method [17]) for a room of  $10 \times 5 \times 4$  m, the reflection coefficients of the walls are 0.8, with a loudspeaker on position (3.21, 3.21, 2.5) m and a microphone on position (1, 2, 1) m. A sampling frequency of 16 kHz is used. A speech signal contains several different types of onsets, and hence in order to obtain reproducible results, we use a white noise input signal with intermittently a level of -46 dB during 1 second, and a level of 0 dB during 1 second. The adaptive filter in the PEM-AFROW algorithm has a (sufficient) order of  $n_F + 1 = 2000$  taps. The frame length for energy comparison (Section 10.2.1) is 20 ms, parameter  $c$  defined in Section 10.2.1 is 5, and the adaptation inhibition time after detection of an onset (Section 10.2.1) is  $\Delta = 2(n_F + 1)$ . The added gain is set to 5 dB.

For the algorithm without onset detection, audible (transient) instabilities are present at the onsets of the louder signal blocks, even with small step sizes for the adaptive NLMS algorithm ( $\mu = 0.005$ ), which corresponds to an unacceptably low convergence and tracking speed. With  $\mu = 0.05$ , these transients even take more than a second to disappear. With the onset detection switched on, the stepsize can be as large as  $\mu = 0.2$  (which obviously results in an acceptable tracking speed), without audible instabilities.

These results are confirmed when a speech input signal is used instead of a white noise signal. The values for the maximum of  $\mu$  when onset detection is off may vary, because of the different types of onsets in a speech signal. But the value of  $\mu = 0.2$  as a maximum value when onset detection is switched on, is consistent with the white noise experiment.

### 10.4.2 Foreground/Background Filter

The same simulation setup is used, without the onset detection algorithm. The step size for the foreground filter is chosen 10 times smaller than for the background filter. The values for the parameters mentioned in Section 10.2.3 are  $\gamma_1 = \gamma_3 = 0.86$  and condition (10.25) is omitted. We evaluate the behavior for the previously mentioned input signal. Large stepsizes can be used for the background filter (e.g.,  $\mu = 0.9$ ), which allows for very fast convergence, while a small variance can be obtained due to the behavior of the foreground filter with the smaller step size. When  $\mu$  is chosen not too large, then for high loop gains (high FNR), e.g., with an added gain of 10 dB, the foreground/background filter structure does not exhibit instabilities at onsets, because the slow foreground filter “bounds” the divergence of the faster background filter. For low loop gains (e.g., with negative added gains) or higher values of  $\mu$ , onset detection should be used to prevent coefficients from drifting from the optimal solution.

The foreground/background filter structure is not only important for robustness related to signal properties, but also for robustness against system changes (i.e., changes of the RIR). The simulations use the room and geometry setup as detailed in Section 10.2.4. In the simulations, abrupt switches between two artificial impulse responses are performed, one corresponding to microphone position (1, 2, 1) m, and one corresponding to microphone position (1, 1.95, 1) m. In this case a real speech input signal is used. As a reference, we consider a system with a single adaptive filter with  $\mu = 0.09$ , resulting in the same variance as the foreground filter in the twin filter structure. In the twin filter setup, for an added gain of 5 dB, no audible transient is present, while a clear transient is audible in the reference system. For an added gain of 10 dB, a short transient is audible during 0.5 s. For an added gain of 13 dB, a transient instability is audible during 1 s.

All these results were also verified with a real-time (16 kHz) implementation with a real speech input signal, in a real acoustic environment (office room of  $7 \times 3.5 \times 3.6$  m, a reverberation time (T60) of 120 ms, and a distance between loudspeaker and microphone of 2 m).

### 10.4.3 Proactive Notch Filtering

Real-time experiments at a 16 kHz sampling rate in a real environment were performed with added gains of about 5 dB, using only the notch filtering section (proactive, designed based on the PEM-AFROW estimate), and without the additional controller  $\hat{\mathbf{f}}_0(t)$ . It is known that this is a typical stable added gain for notch-filtering-based feedback suppression. Fast movements (about 1 Hz) of the microphone over a distance of 30 cm, at 1 m distance from the loudspeaker, do not result in instability for this proactive notch filtering approach, while for the PEM-AFROW-based feedback suppression algorithm in [4],[5], instability does indeed sometimes occur under these conditions. Note however that, for constant microphone-loudspeaker distances, much higher added gains can be achieved with the approach in [4],[5] (up to 15 dB), even with large changes in the RIR due to movements of, e.g., people in the vicinity of the loudspeaker or microphone.

In addition to this experiment, and in order to provide reproducible results, simulations were performed with a room geometry as detailed in Section 10.2.4, for a combined system as in Fig. 10.6. Sudden changes between the two RIRs are applied (which are worse conditions than the 1 Hz movements in the real-environment experiment described above). Then it is evaluated for which maximum gain level, no instability or audible artifacts occur upon changes of the impulse responses. The results are provided in Table 10.2. The step size for the NLMS filter in the PEM-AFROW algorithm is 0.05. For comparison: the stable added gain when only the notch filtering section is used is 5 dB.

Table 10.2: Simulation results for different notch filter parameters. Bandwidths are normalized ( $1 = \text{Nyquist frequency}$ ).

Bandwidth	Depth	Added gain
3dB BW = $1/20$	12 dB	10 dB
3dB BW = $1/10$	12 dB	13.9 dB
1dB BW = $1/5$	3 dB	12.9 dB

#### 10.4.4 Highpass Filter

Though difficult to quantify numerically, real-time experiments show a huge improvement of stability in case the microphone is displaced when a highpass filter is available, see Section 10.2.6.

#### 10.4.5 Combined Real-Time System

The combined frequency domain system is implemented for use in a real-time environment, in C++ on a Linux PC (Pentium M, 1.7 MHz). A graphical user interface is available for setting parameters of different building blocks, and for switching on and off the different modules. The filter length is 2048 taps, and the sampling frequency 16 kHz. In this way, a RIR of 128 ms can be modeled. The system is set up in a real environment (office room of  $7 \times 3.5 \times 3.6$  m, with a reverberation time (T60) of 120 ms, and a distance between loudspeaker and microphone of 2 m), and maintains stability far better than the PEM-AFROW-based system without the robustness add-ons.

As a general result from real-time experiments, we can state that the most important elements for increased robustness are the highpass filter, the foreground/background filter structure, and onset detection. Proactive notch filtering can increase robustness somewhat, but has the disadvantage of introducing signal distortion.

## 10.5 Conclusions

In this paper, we have shown a number of procedures to increase the robustness and to reduce the complexity of a PEM-AFROW-based feedback cancellation system. Concerning robustness, we have shown how onset detection avoids instability at onsets of the near-end signal. We have derived an NLMS-type algorithm which allows for prior knowledge of the RIR to be incorporated in the adaptation rule which is particularly useful at system startup. Then a twin filter structure (adopted from acoustic echo cancellation) was shown to be also

effective in acoustic feedback cancellation. On the one hand RIR changes can be tracked quickly while on the other hand a small variance on the estimate of the RIR can be maintained during steady-state operation. A highpass filter is included to avoid high variance on the low-frequency part of the RIR estimate. Finally, the combination of notch filtering and PEM-AFROW was shown to provide a more robust solution to feedback path changes if some distortion is allowable. Concerning computational complexity we have demonstrated how the PEM-AFROW algorithm can be implemented in subbands, where extra complexity reduction can be obtained due to the physical characteristics of reverberation in a typical environment, and as an alternative a low-latency frequency domain implementation can be used.

## Bibliography

- [1] G. Nicollini, Y. Mazoyer, C. Crippa, S. Pernici, and P. Confalonieri, "A CMOS fully integrated antilarson system for digital telephones," *IEEE J. Solid-State Circuits*, vol. 28, no. 5, pp. 605–609, May 1993.
- [2] R. Porayath and D. J. Mapes-Riordan, "Acoustic feedback elimination using adaptive notch filter algorithm," U.S. Patent 5,999,631, Dec., 1999.
- [3] M. P. Lewis, T. J. Tucker, and D. M. Oster, "Method and apparatus for adaptive audio resonant frequency filtering," U.S. Patent 5,245,665, Sep., 1993.
- [4] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, "Acoustic feedback suppression for long acoustic paths using a nonstationary source model," in *Proc. 13th European Signal Process. Conf. (EUSIPCO '05)*, Antalya, Turkey, Sep. 2005.
- [5] —, "Acoustic feedback suppression for long acoustic paths using a nonstationary source model," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.
- [6] S. Kamerling, K. Janse, and F. van der Meulen, "A new way of acoustic feedback suppression," in *Preprints AES 104th Convention*, Amsterdam, The Netherlands, May 1998, AES Preprint 4735.
- [7] C. P. Janse and H. J. W. Belt, "Sound reinforcement system having an echo suppressor and loudspeaker beamformer," WIPO Patent Application WO/2003/010996, Feb., 2003.
- [8] C. P. Janse and P. A. A. Timmermans, "Signal amplifier system with improved echo cancellation," U.S. Patent 5,748,751, May, 1998.

- [9] M. G. Siqueira and A. Alwan, "Steady-state analysis of continuous adaptation in acoustic feedback reduction systems for hearing-aids," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 4, pp. 443–453, Jul. 2000.
- [10] J. Hellgren and U. Forssell, "Bias of feedback cancellation algorithms in hearing aids based on direct closed loop identification," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 7, pp. 906–913, Nov. 2001.
- [11] A. Spriet, I. Proudler, M. Moonen, and J. Wouters, "Adaptive feedback cancellation in hearing aids with linear prediction of the desired signal," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3749–3763, Oct. 2005.
- [12] T. van Waterschoot, G. Rombouts, and M. Moonen, "Optimally regularized adaptive filtering algorithms for room acoustic signal enhancement," *Signal Processing*, vol. 88, no. 3, pp. 594–611, Mar. 2008.
- [13] —, "MSE optimal regularization of APA and NLMS algorithms in room acoustic applications," in *Proc. 2006 Int. Workshop Acoustic Echo Noise Control (IWAENC '06)*, Paris, France, Sep. 2006.
- [14] J. M. Kates, "Constrained adaptation for feedback cancellation in hearing aids," *J. Acoust. Soc. Amer.*, vol. 106, no. 2, pp. 1010–1019, Aug. 1999.
- [15] K. Ochiai, T. Araseki, and T. Ogihara, "Echo canceler with two echo path models," *IEEE Trans. Commun.*, vol. 25, no. 6, pp. 589–595, Jun. 1977.
- [16] J. Prado and E. Moulines, "Frequency-domain adaptive filtering with applications to acoustic echo cancellation," *Ann. Télécommun.*, vol. 49, no. 7–8, pp. 414–428, Jul./Aug. 1994.
- [17] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943–950, Apr. 1979.
- [18] T. van Waterschoot and M. Moonen, "A pole-zero placement technique for designing second-order IIR parametric equalizer filters," *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 8, pp. 2561–2565, Nov. 2007.
- [19] P. P. Vaidyanathan, *Multirate systems and filter banks*. Englewood Cliffs, New Jersey: Prentice-Hall, 1993.
- [20] A. Spriet, G. Rombouts, M. Moonen, and J. Wouters, "Adaptive feedback cancellation in hearing aids," *J. Franklin Inst.*, vol. 343, no. 6, pp. 545–573, Sep. 2005.
- [21] J.-S. Soo and K. K. Pang, "Multidelay block frequency domain adaptive filter," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-38, no. 2, pp. 373–376, Feb. 1990.



## Chapter 11

# Linear Prediction of Audio Signals

Comparison of linear prediction models for audio signals

Toon van Waterschoot and Marc Moonen

Published in *EURASIP J. Audio, Speech, Music Process.*, vol. 2008, Article ID 706935, 24 pages, 2008. doi:10.1155/2008/706935

### Contributions of first author

- literature study
- co-development of theoretical framework, signal models, and LP criterion
- co-analysis of conventional LP model
- co-analysis of alternative LP models
- co-design of evaluation experiments
- software implementation and computer simulations
- co-interpretation of simulation results
- co-formulation of conclusion
- text redaction and editing



## Abstract

While linear prediction (LP) has become immensely popular in speech modeling, it does not seem to provide a good approach for modeling audio signals. This is somewhat surprising, since a tonal signal consisting of a number of sinusoids can be perfectly predicted based on an (all-pole) LP model with a model order that is twice the number of sinusoids. We provide an explanation why this result cannot simply be extrapolated to LP of audio signals. If noise is taken into account in the tonal signal model, a low-order all-pole model appears to be only appropriate when the tonal components are uniformly distributed in the Nyquist interval. Based on this observation, different alternatives to the conventional LP model can be suggested. Either the model should be changed to a pole-zero, a high-order all-pole, or a pitch prediction model, or the conventional LP model should be preceded by an appropriate frequency transform, such as a frequency warping or downsampling. By comparing these alternative LP models to the conventional LP model in terms of frequency estimation accuracy, residual spectral flatness, and perceptual frequency resolution, we obtain several new and promising approaches to LP-based audio modeling.

## 11.1 Introduction

Linear prediction (LP) is a widely used and well understood technique for the analysis, modeling, and coding of speech signals [1]. Its success can be attributed to its correspondence with the speech generation process. The vocal tract can be modeled as a slowly time-varying, low-order all-pole filter, while the glottal excitation can be represented either by a white noise sequence (for unvoiced sounds), or by an impulse train generated by periodic vibrations of the vocal chords (for voiced sounds). By using this so-called source-filter model, a speech segment can be whitened with a cascade of a formant predictor for removing short-term correlation, and a pitch predictor for removing long-term correlation [2].

The source-filter model is much less popular in audio analysis than in speech analysis. First of all, the generation of musical sounds is highly dependent on the instruments used, hence it is hard to propose a generic audio signal generation model. Second, from a physical point of view, polyphonic audio signals should be analyzed using multiple source-filter models, which seems rather impractical. Finally, the enormous success of perceptual audio coders [3] and the recent advent of parametric coders based on the sinusoidal model [4], originally proposed for speech analysis and synthesis [5], have shifted the research interest in audio analysis away from the LP approach. Nevertheless, some audio coding algorithms still rely on LP [6]-[15], which is then usually performed on a warped frequency scale [16]. Also, in audio signal processing applications

other than coding, prediction error filters obtained with LP are used for the whitening of audio signals, e.g., to produce robust and fast converging acoustic echo and feedback cancelers [17]-[20].

Since many audio signals exhibit a large degree of tonality, i.e., their frequency spectrum is characterized by a finite number of dominant frequency components, it is useful to analyze LP of audio signals in the frequency domain, i.e., from a spectral estimation point of view. Intuitively, one could expect that performing LP using a model order that is twice the number of tonal components, leads to a signal estimate in which each of the spectral peaks is modeled with a complex conjugate pole pair close to (but inside) the unit circle. In practice, however, this does not seem to be the case, and very often a poor LP signal estimate is obtained. The fundamental problem when performing LP of an audio signal, is that apart from the tonal components, a broadband noise term should generally also be incorporated in the tonal model. The noise term can either account for imperfections in the signal's tonal behavior, or for noise introduced when working with finite-length data windows. Whereas a sum of  $N$  sinusoids can be perfectly modeled using an AR( $2N$ ) model, i.e., an autoregressive or all-pole model of order  $2N$ , a sum of  $N$  sinusoids plus (white) noise, should instead be modeled using an ARMA( $2N,2N$ ) model, i.e., an autoregressive moving-average or pole-zero model with  $2N$  zeros and  $2N$  poles [21]-[25].

A first consequence of incorporating a noise term in the tonal signal model, is that the LP spectral estimate is smoothed [22],[26], due to the fact that the estimated poles are drawn towards the origin of the  $z$ -plane [22],[27]. A second consequence, which to our knowledge has not been recognized up till now, is that the estimated poles tend to be equally distributed around the unit circle when noise is present, even at high signal-to-noise ratios and for low AR model orders. From this observation, it follows that signals with tonal components that are approximately equally distributed in the Nyquist interval can be better represented with an all-pole model than signals that have their tonal components concentrated in a selected region of the Nyquist interval. Unfortunately, audio signals tend to belong to the latter class of signals, since they are typically sampled at a sampling frequency that is much higher than the frequency of their dominating tonal components.

In [28], it was shown that audio signals having their dominating tonal components in a frequency region that is small compared to the entire signal bandwidth may exhibit a large autocorrelation matrix eigenvalue spread and hence tend to produce inaccurate LP models due to numerical instability. A stabilization method based on a selective LP (SLP) model [1] was proposed, which reduces the LP model bandwidth to the frequency region of interest. The influence of the signal's frequency distribution on LP performance was also recognized with the development of so-called frequency-warped linear prediction (WLP) [12],[16]. The warping operation is a non-uniform frequency transform

which is usually designed to approximate the constant- $Q$  frequency scale [29], and also provides a good match with the Bark or ERB psychoacoustic scales, provided that the warping parameter is chosen properly [30]. In [12], WLP was shown to outperform conventional LP in terms of resolving adjacent peaks in the signal spectrum, however, no gain in spectral flatness of the LP residual was obtained. We will review the SLP and WLP models, as well as three other LP models that appear to be suited for tonal audio signals, and show how all of these models are capable of solving the frequency distribution issue described above. More specifically, we will also consider high-order all-pole models [22], constrained pole-zero models [24],[25],[31]-[37] and pitch prediction models. Pitch prediction (PLP), also known as long-term prediction, was originally proposed for speech modeling and coding, and was more recently applied to audio signal modeling in the context of the MPEG-4 advanced audio coder (AAC) [38],[39]. High-order (HOLP) and pole-zero (PZLP) linear prediction models have not been applied to audio modeling before, however, some speech analysis techniques rely on a PZLP model [40]-[42]. All considered approaches result in stable LP models, and some outperform the WLP model both in terms of conventional measures, such as frequency estimation error and residual spectral flatness [43, Ch. 6], and in terms of perceptually motivated measures, such as inter-peak dip depth (IDD) [12]. Moreover, many of these alternative models perform even better when cascaded with a conventional LP model. The LP models described in this paper were evaluated and compared experimentally for a synthetic audio signal in [44]. This work is extended here by also performing a mathematical analysis of the different LP models, and describing additional simulation results for synthetic signals and true monophonic and polyphonic audio signals.

This paper is organized as follows. Section 11.2 provides some background material on the signal model and the LP criterion. In Section 11.3, we analyze the performance of the conventional LP model, and illustrate the influence of the distribution of the tonal components in the analyzed signal. In Section 11.4, five alternative LP models are reviewed and interpreted as potential solutions to the observed frequency distribution problem. The emphasis is on the influence of using models other than the conventional low-order all-pole model, and not on how the model parameters are estimated. However, for each LP model, references to existing estimation methods are provided. LP model pole-zero plots and magnitude responses for a synthetic audio signal are presented throughout Sections 11.3 and 11.4. A detailed analysis is only provided for the pole-zero LP model, since all other alternative LP models are all-pole models, which can be analyzed using an approach similar to the conventional LP model analysis in Section 11.3. In Section 11.5, we provide LP model pole-zero plots and magnitude responses for true monophonic and polyphonic audio signals. Furthermore, the conventional and alternative LP models are compared in terms of frequency estimation accuracy, residual spectral flatness, and perceptual frequency resolution, both for synthetic and true audio signals.

Finally, Section 11.6 concludes the paper.

## 11.2 Preliminaries

### 11.2.1 Tonal Audio Signal Model

We will only consider tonal audio signals, i.e., signals having a continuous spectrum containing a finite number of dominant frequency components. In this way, the majority of audio signals is covered, except for the class of percussive sounds. The performance of the different LP models described below will be evaluated for three types of audio signals: synthetic audio signals consisting of a sum of harmonic sinusoids in white noise, true monophonic audio signals, and true polyphonic audio signals.

The fundamental frequency of monophonic audio signals is usually, i.e., for most musical instruments, in the range 100–1000 Hz. The number of relevant harmonics (i.e., frequency components at multiples of the fundamental frequency, having a magnitude that is significantly larger than the average signal power) is typically between 10 and 20. It can thus be seen that most dominating frequency components in audio signals, sampled at  $f_s = 44.1$  kHz, lie in the lower half of the Nyquist interval, i.e., between 0 and 11025 Hz (corresponding to the angular frequency range from 0 to  $\pi/2$ ). This property will be a key issue in the rest of the paper.

Like for speech signals, we can also assume short-term stationarity for audio signals. Monophonic audio signals can typically be divided in musical notes of different duration. Each note can then be subdivided in four parts: the attack, decay, sustain, and release parts. The sustain part is usually the longest part of the note, and exhibits the highest degree of stationarity. The attack and decay parts are shortest and may show transient behavior, such that stationarity can only be assumed on very short time windows (a few ms). Whereas LP of speech signals is typically performed on time windows of around 20 ms, longer windows appear to be beneficial for LP of audio signals. In our examples, a time window of 46.4 ms is used, corresponding to  $L = 2048$  samples at  $f_s = 44.1$  kHz, or, in musical terms, 1/32 note at 161.5 beats per minute. In our theoretical derivations, however, we will assume  $L \rightarrow \infty$  to avoid window end effects.

The underlying signal model that is assumed for all audio signals throughout this paper is the following:

$$y(t) = \sum_{n=1}^N \alpha_n \cos(\omega_n t + \phi_n) + r(t), \quad t = 1, \dots, L \quad (11.1)$$

where, for ease of notation, the time index  $t$  has been normalized w.r.t. the sampling period  $T_s = 1/f_s$ . This signal model is referred to as the *tonal signal*

*model*, and may differ from the sinusoidal model [5] used in speech and audio coding in that only the tonal components in the observed audio signal  $y(t)$  are modeled by sinusoids, while the non-tonal components are contained in the noise term  $r(t)$ . The tonal components correspond to the fundamental frequencies and their relevant harmonics, and are characterized by their amplitudes  $\alpha_n$ , (radial) frequencies  $\omega_n \in [0, \pi]$  and phases  $\phi_n \in [0, 2\pi)$ ,  $n = 1, \dots, N$ . The noise term  $r(t)$  will generally have a non-white, continuous spectrum, and may also contain low-power harmonics.

Two special cases of the tonal signal model are of particular interest in audio signal modeling. In the *monophonic signal model*, it is assumed that all tonal components are harmonically related to a single fundamental frequency  $\omega_0$ , i.e.,

$$y(t) = \sum_{n=1}^N \alpha_n \cos(n\omega_0 t + \phi_n) + r(t), \quad t = 1, \dots, L. \quad (11.2)$$

In the *polyphonic signal model*, the signal is assumed to contain multiple sets of harmonically related sinusoids, with multiple fundamental frequencies  $\omega_{0,n}$ ,  $n = 1, \dots, N$ :

$$y(t) = \sum_{n=1}^N \left( \sum_{m=1}^{M_n} \alpha_{n,m} \cos(m\omega_{0,n} t + \phi_{n,m}) \right) + r(t), \quad t = 1, \dots, L. \quad (11.3)$$

Note that the number of relevant harmonics ( $M_n - 1$ ) may differ for each of the  $N$  fundamental frequencies  $\omega_{0,n}$ , and that only one overall noise term is added.

The monophonic signal model in (11.2) is an harmonic signal model, while the tonal and polyphonic signal models in (11.1) and (11.3) are not. We should stress that of all LP models described below, the pitch prediction model described in Section 11.4.3, is the only model in which the harmonicity property is exploited. The other models do not rely on harmonicity, although the calculation of the LP model parameters may be simplified by taking harmonicity into account.

**Example 11.1** [Synthetic audio signal] A synthetic audio signal, generated from the monophonic signal model in (11.2), is well suited for examining the properties of the LP models presented below, since it provides exact knowledge of the fundamental frequency  $f_0 = \omega_0(f_s/2\pi)$  and the number of harmonics. In the examples throughout Section 11.3 and 11.4, a synthetic audio signal is used with  $L = 2048$  samples,  $N = 15$  tonal components and random, uniformly distributed amplitudes  $\alpha_n \in [0, 1]$  and phases  $\phi_n \in [0, 2\pi)$ . The synthetic audio signal and its magnitude spectrum are shown in Figs. 11.1(a) and 11.1(b), respectively. The radial fundamental frequency was chosen to be  $\omega_0 = 2\pi/64$ , i.e., with 64 samples per period  $T_0$ , such that, at  $f_s = 44.1$  kHz, the fundamental frequency  $f_0 \approx 689.1$  Hz is in the mid-range of musical notes (i.e., slightly

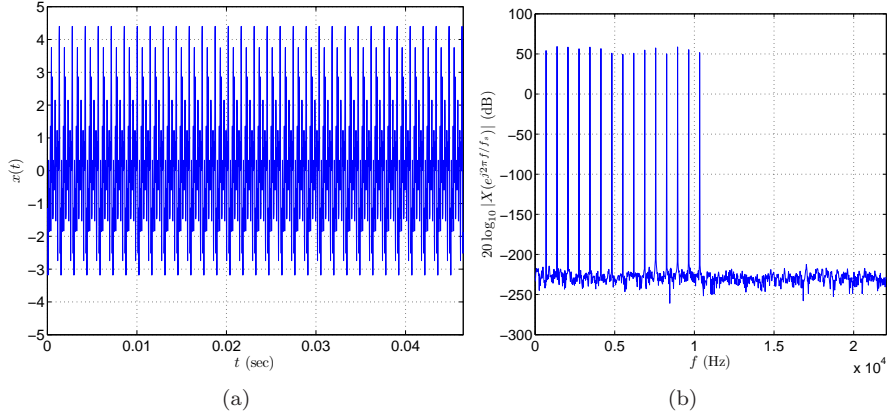


Figure 11.1: Synthetic audio signal: (a) time-domain waveform, (b) magnitude spectrum.

lower than F5). The fundamental frequency and its harmonics are then also in the discrete set of frequencies at which the length- $L$  discrete Fourier transform (DFT) is evaluated (see Fig. 11.1(b)). The pitch period  $T_0$  being equal to an integer number of sampling periods ( $T_0 = 64T_s$ ), will allow us to clearly illustrate the effect of pitch prediction in Section 11.4.3. Finally,  $T_0$  also being an integer multiple of  $2(N+1)T_s$  will yield an integer downsampling operation in the SLP method in Section 11.4.5.

### 11.2.2 Linear Prediction Criterion

The aim of LP is to obtain a linear parametric model  $G(z)$  that predicts the observed signal  $y(t)$  up to an uncorrelated residual  $e(t, \xi)$ ,

$$Y(z) = G(z)E(z, \xi) \quad (11.4)$$

or

$$E(z, \xi) = H(z)Y(z) \quad (11.5)$$

where  $\xi$  represents a vector that contains the LP model parameters,  $Y(z)$  and  $E(z, \xi)$  denote the  $z$ -transform of the observed and residual signal, respectively, and  $H(z) = G^{-1}(z)$  corresponds to the prediction error filter (PEF), which has the property of whitening the input signal  $y(t)$ . The PEF transfer function  $H(z)$  is required to be stable, while the LP model transfer function  $G(z)$  is not. In fact, when modeling sinusoidal components in the observed signal  $y(t)$ , an unstable LP model  $G(z)$  having poles on the unit circle can be very useful.

The LP model is generally an infinite impulse response (IIR) model, i.e.,

$$G(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \dots + b_{2Q} z^{-2Q}}{1 + a_1 z^{-1} + \dots + a_{2P} z^{-2P}} \quad (11.6)$$

with the numerator and denominator order defined as  $2Q$  and  $2P$ , respectively. While in conventional LP,  $G(z)$  is an all-pole model (i.e.,  $B(z) \equiv 1$ ), in this paper we also consider pole-zero LP models. For analyzing the LP performance for tonal input signals, it will be useful to consider the radial representation of  $G(z)$ ,

$$G(z) = \frac{b_0 \prod_{l=1}^Q (1 - \rho_l e^{j\zeta_l} z^{-1})(1 - \rho_l e^{-j\zeta_l} z^{-1})}{\prod_{l=1}^P (1 - \nu_l e^{j\theta_l} z^{-1})(1 - \nu_l e^{-j\theta_l} z^{-1})} \quad (11.7)$$

$$= \frac{b_0 \prod_{l=1}^Q (1 - 2\rho_l \cos \zeta_l z^{-1} + \rho_l^2 z^{-2})}{\prod_{l=1}^P (1 - 2\nu_l \cos \theta_l z^{-1} + \nu_l^2 z^{-2})} \quad (11.8)$$

with  $\rho_l, \nu_l$  denoting the zero and pole radii, and  $\zeta_l, \theta_l$  the numerator and denominator resonance frequencies, respectively. In the sequel we will assume  $b_0 = 1$ , such that the LP model parameter vector can be defined as follows:

$$\boldsymbol{\xi} = [\theta_1, \dots, \theta_P, \nu_1, \dots, \nu_P, \zeta_1, \dots, \zeta_Q, \rho_1, \dots, \rho_Q]^T. \quad (11.9)$$

From a spectral estimation point of view, the parameter vector  $\boldsymbol{\xi}$  should be estimated such that the LP residual  $e(t, \boldsymbol{\xi})$  has an approximately flat spectrum [1]. In the case of audio LP, the residual does not have to be a white noise signal, as is often assumed in other LP applications, but it can also be a Dirac impulse, which also has a flat spectrum. The parameter vector estimate is the result of minimizing a least-squares (LS) criterion, which can be expressed in the time domain as well as in the frequency domain, following Parseval's theorem:

$$\min_{\boldsymbol{\xi}} V(\boldsymbol{\xi}) = \min_{\boldsymbol{\xi}} \sum_{t=1}^L e^2(t, \boldsymbol{\xi}) = \min_{\boldsymbol{\xi}} \frac{1}{L} \sum_{k=0}^{L-1} |E(e^{j\frac{2\pi k}{L}}, \boldsymbol{\xi})|^2 \quad (11.10)$$

with  $E(e^{j\frac{2\pi k}{L}}, \boldsymbol{\xi})$ ,  $k = 0, \dots, L-1$  the  $L$ -point discrete Fourier transform (DFT) of the LP residual.

In the theoretical analysis, we will assume an infinitely long observation window

( $L \rightarrow \infty$ ), such that (11.10) becomes

$$\min_{\xi} V(\xi) = \min_{\xi} \frac{1}{2\pi} \int_0^{2\pi} |E(e^{j\omega}, \xi)|^2 d\omega = \min_{\xi} \frac{1}{2\pi} \int_0^{2\pi} |H(e^{j\omega})|^2 |Y(e^{j\omega})|^2 d\omega \quad (11.11)$$

using (11.5) to obtain the second equality, in which  $|H(e^{j\omega})|^2$  denotes the PEF magnitude response and  $|Y(e^{j\omega})|^2$  is the power spectrum of  $y(t)$ . From the tonal signal model in (11.1), and assuming that the cross-spectrum of the tonal part and the noise part of  $y(t)$  is zero, we obtain

$$|Y(e^{j\omega})|^2 = \sum_{n=1}^N \frac{\alpha_n^2}{4} (\delta(\omega - \omega_n) + \delta(\omega + \omega_n)) + |R(e^{j\omega})|^2 \quad (11.12)$$

such that (11.11) can be rewritten, using  $|H(e^{j\omega_n})|^2 = |H(e^{-j\omega_n})|^2$ , as

$$\min_{\xi} V(\xi) = \min_{\xi} \left[ \sum_{n=1}^N \frac{\alpha_n^2}{2} |H(e^{j\omega_n})|^2 + \frac{1}{2\pi} \int_0^{2\pi} |H(e^{j\omega})|^2 |R(e^{j\omega})|^2 d\omega \right]. \quad (11.13)$$

To simplify the analysis, we assume that the noise term  $r(t)$  in the tonal signal model has a flat spectrum, i.e.,  $|R(e^{j\omega})|^2 = \sigma_r^2$ ,  $\forall \omega$ , such that

$$\min_{\xi} V(\xi) = \min_{\xi} \left[ \sum_{n=1}^N \frac{\alpha_n^2}{2} |H(e^{j\omega_n})|^2 + \frac{\sigma_r^2}{2\pi} \int_0^{2\pi} |H(e^{j\omega})|^2 d\omega \right]. \quad (11.14)$$

This approximation can be justified in the LP analysis by noting that the noise term in the tonal signal model is spectrally much flatter than the tonal part of the observed signal.

### 11.3 Conventional Linear Prediction Model

We now analyze the minimization of the LP criterion in (11.14) for a conventional, all-pole LP model. The PEF is in this case an all-zero filter,

$$H(z) = \prod_{l=1}^P (1 - 2\nu_l \cos \theta_l z^{-1} + \nu_l^2 z^{-2}). \quad (11.15)$$

We will examine the effect of setting  $P = N$ , since we know that an AR( $2N$ ) model should be capable of perfectly modeling a noiseless sum of  $N$  sinusoids [25]. However, in the tonal signal model (11.1), a noise term is also present, hence the solution to the LP estimation problem will be a compromise of attenuating the tonal components, while increasing (or maintaining) the flatness



of the noise spectrum. In [22], this compromise was analyzed w.r.t. its effect on the radii  $\{\nu_l\}_{l=1}^P$  of the PEF zeros, while disregarding the effect on the PEF zero angles  $\{\theta_l\}_{l=1}^P$ . In our analysis, we will focus on the effect of the noise on the estimated PEF zero angles.

The LP model parameters in  $\boldsymbol{\xi} = [\theta_1, \dots, \theta_P, \nu_1, \dots, \nu_P]^T$  can be obtained as the solution to a system of  $2P$  equations, that are obtained by differentiating the LP criterion in (11.14) w.r.t.  $\{\theta_l\}_{l=1}^P$  and  $\{\nu_l\}_{l=1}^P$ , i.e.,

$$\left\{ \frac{\partial}{\partial \theta_l} \left\{ \sum_{n=1}^N \frac{\alpha_n^2}{2} |H(e^{j\omega_n})|^2 + \frac{\sigma_r^2}{2\pi} \int_0^{2\pi} |H(e^{j\omega})|^2 d\omega \right\} = 0, \quad l = 1, \dots, P \right. \quad (11.16)$$

$$\left. \left\{ \frac{\partial}{\partial \nu_l} \left\{ \sum_{n=1}^N \frac{\alpha_n^2}{2} |H(e^{j\omega_n})|^2 + \frac{\sigma_r^2}{2\pi} \int_0^{2\pi} |H(e^{j\omega})|^2 d\omega \right\} = 0, \quad l = 1, \dots, P \right. \right. \quad (11.17)$$

We will first consider the case in which the noise term is equal to zero, i.e.,  $\sigma_r^2 = 0$ . In this case, the LP estimation problem can be formulated as follows:

$$\min_{\boldsymbol{\xi}} V(\boldsymbol{\xi}) = \min_{\boldsymbol{\xi}} \sum_{n=1}^N \frac{\alpha_n^2}{2} |H(e^{j\omega_n})|^2 \quad (11.18)$$

which leads to the following system of equations:

$$\left\{ \sum_{n=1}^N \frac{\alpha_n^2}{2} \left[ \frac{\partial}{\partial \theta_l} |H(e^{j\omega})|^2 \right]_{\omega=\omega_n} = 0, \quad l = 1, \dots, P \right. \quad (11.19)$$

$$\left. \left\{ \sum_{n=1}^N \frac{\alpha_n^2}{2} \left[ \frac{\partial}{\partial \nu_l} |H(e^{j\omega})|^2 \right]_{\omega=\omega_n} = 0, \quad l = 1, \dots, P. \right. \right. \quad (11.20)$$

From the PEF transfer function in (11.15), we can calculate the PEF magnitude response, and its partial derivatives w.r.t. the parameters  $\theta_l, \nu_l$ ,  $l = 1, \dots, P$ :

$$|H(e^{j\omega})|^2 = \prod_{l=1}^P \left[ (1 - \nu_l^2)^2 + 4\nu_l^2 (\cos \omega - \cos \theta_l)^2 - 4\nu_l (1 - \nu_l)^2 \cos \theta_l \cos \omega \right] \quad (11.21)$$

$$\begin{aligned} \frac{\partial}{\partial \theta_l} |H(e^{j\omega})|^2 &= 4\nu_l \sin \theta_l [(1 + \nu_l^2) \cos \omega - 2\nu_l \cos \theta_l] \\ &\times \prod_{\substack{k=1 \\ k \neq l}}^P \left[ (1 - \nu_k^2)^2 + 4\nu_k^2 (\cos \omega - \cos \theta_k)^2 - 4\nu_k (1 - \nu_k)^2 \cos \theta_k \cos \omega \right] \end{aligned} \quad (11.22)$$

$$\frac{\partial}{\partial \nu_l} |H(e^{j\omega})|^2 = 4[\nu_l^3 - (3\nu_l^2 + 1) \cos \theta_l \cos \omega + \nu_l (\cos^2 \omega - \sin^2 \omega + 2 \cos^2 \theta_l)]$$

$$\times \prod_{\substack{k=1 \\ k \neq l}}^P \left[ (1 - \nu_k^2)^2 + 4\nu_k^2 (\cos \omega - \cos \theta_k)^2 - 4\nu_k(1 - \nu_k)^2 \cos \theta_k \cos \omega \right] \quad (11.23)$$

The system of equations (11.19)-(11.20) with (11.22)-(11.23) generally has multiple solutions, even when the PEF zero angles  $\{\theta_l\}_{l=1}^P$  are constrained to lie in  $[0, \pi]$ , which correspond to (local) minima of the LP criterion. The global minimum  $V(\boldsymbol{\xi}) = 0$  in case  $P = N$  is obtained for the parameter values

$$\begin{cases} \theta_l = \omega_l, & l = 1, \dots, P \\ \nu_l = 1, & l = 1, \dots, P. \end{cases} \quad (11.24)$$

$$(11.25)$$

The PEF thus behaves as a cascade of second-order all-zero notch filters, with all the zeros on the unit circle and with the notch frequencies equal to the frequencies of the tonal components. Note that the corresponding LP model transfer function  $G(z) = H^{-1}(z)$  is in this case unstable.

Next, we will illustrate the influence of a non-zero noise term on the solution (11.24)-(11.25) obtained in the noiseless case. The second term in the LP criterion (11.14), which is due to the noise, can be rewritten using Parseval's theorem as follows:

$$\frac{\sigma_r^2}{2\pi} \int_0^{2\pi} |H(e^{j\omega})|^2 d\omega = \sigma_r^2 \left( 1 + \sum_{i=1}^{2P} a_i^2 \right). \quad (11.26)$$

It can hence be seen that this term acts as a minimum norm constraint in the LP criterion, in the sense that it penalizes the squared norm of the PEF impulse response coefficient vector

$$\mathbf{a} = [1 \quad a_1 \quad \dots \quad a_{2P}]^T. \quad (11.27)$$

This minimum norm constraint has two effects on the solution (11.24)-(11.25) that was obtained in the noiseless case. A first effect, which was investigated in [22], is that the estimated PEF zeros are drawn towards the origin of the  $z$ -plane and hence the estimated PEF zero radii  $\{\nu_l\}_{l=1}^P$  are less than one. A second effect is related to the estimated PEF zero angles  $\{\theta_l\}_{l=1}^P$ . Consider the following constrained estimation problem,

$$\min_{\boldsymbol{\xi}} V(\boldsymbol{\xi}) = \min_{\boldsymbol{\xi}} \sigma_r^2 \left( 1 + \sum_{i=1}^{2P} a_i^2 \right) \quad \text{s.t.} \quad \nu_l > 0, \quad l = 1, \dots, P. \quad (11.28)$$

In this estimation problem, the squared norm of the PEF impulse response coefficient vector is minimized under a constraint that rules out the trivial solution  $a_1 = \dots = a_{2P} = 0$ . It is straightforward to see that the solution to (11.28) can be obtained by setting  $a_1 = \dots = a_{2P-1} = 0$  and  $a_{2P} = \beta$  with  $|\beta| > 0$ , which results in a PEF that behaves as a comb filter. The PEF zeros are

then uniformly distributed on a circle with radius  $\sqrt[2P]{\beta}$ , and with an angle  $\pi/P$  between neighbouring zeros. In case  $\beta > 0$ , the PEF zero angles in the Nyquist interval correspond to  $\theta_l = \frac{\pi}{2P} + (l-1)\frac{\pi}{P}$ ,  $l = 1, \dots, P$ , while if  $\beta < 0$ , the PEF has  $P + 1$  zeros in the Nyquist interval, i.e.,  $\theta_l = (l-1)\frac{\pi}{P}$ ,  $l = 1, \dots, P + 1$ . The latter case corresponds to a one-tap pitch prediction filter (see Section 11.4.3), which in fact deviates from the conventional LP model in (11.15) since the zeros at DC and at the Nyquist frequency do not have a corresponding complex conjugate zero.

We can therefore expect that when noise is present, the estimated PEF zeros are both shifted towards the origin and rotated around the origin, hence tending to a uniform angular distribution. The extent to which the zeros are displaced as compared to the noiseless solution, depends on the noise power  $\sigma_r^2$ , which determines the relative importance of the minimum norm constraint in the LP criterion (11.14). The angular effect described above can also be observed in the noiseless case when the LP model order  $2P > 2N$ , in which case the  $2P - 2N$  “extraneous” PEF zeros tend to be uniformly distributed around the unit circle if a minimum norm constraint is incorporated in the LP criterion [45].

**Example 11.2** [Conventional LP of synthetic audio signal] When we estimate a conventional LP model of order  $2P = 2N = 30$  for the synthetic audio signal defined in Ex. 11.1, using the covariance method [1] to calculate the model parameters, we obtain a PEF as illustrated by the pole-zero plot and magnitude response in Figs. 11.2(a) and 11.2(b), respectively. The conventional LP model nearly succeeds at correctly modeling all the tonal components in the synthetic audio signal. However, if we add Gaussian white noise to the observed signal, the covariance method yields the estimated conventional LP model shown in Figs. 11.3(a) and 11.3(b), for a signal-to-noise ratio (SNR) of 25 dB. The PEF zero configuration is in this case clearly a compromise between the LP solutions to the tonal part and the noise part of the signal: the PEF has 9 complex conjugate zero pairs in the sum of sinusoids’ frequency region, and another 6 complex conjugate zero pairs which are nearly uniformly distributed in the upper half of the Nyquist interval. A similar result is obtained when we use the autocorrelation method [1] instead of the covariance method to predict the noiseless synthetic audio signal. Indeed, the autocorrelation method introduces noise in the autocorrelation domain by distorting the signal’s periodicity due to zero padding. This example illustrates the above statement that for conventional LP models, the PEF zero configuration is a trade-off between suppressing the tonal components and keeping the noise spectrum as flat as possible. Note that in the absence of noise (Fig. 11.2(b)), the PEF high-frequency response may become extremely large.

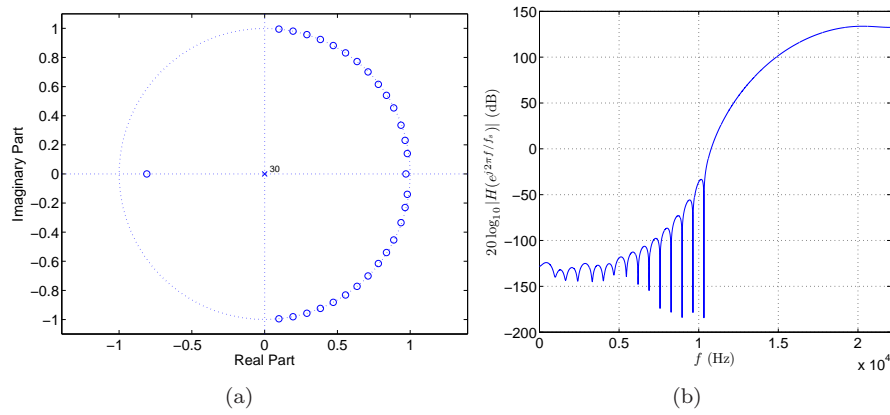


Figure 11.2: Conventional LP model of synthetic audio signal with order  $2P = 30$  and covariance method: (a) PEF pole-zero plot, (b) PEF magnitude response.

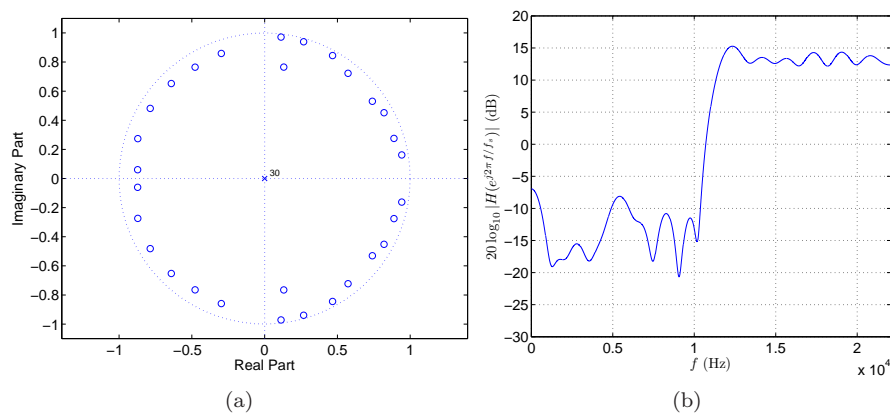


Figure 11.3: Conventional LP model of synthetic audio signal plus noise (SNR = 25 dB) with order  $2P = 30$  and covariance method: (a) PEF pole-zero plot, (b) PEF magnitude response.

## 11.4 Alternative Linear Prediction Models

In this section, we present five existing alternative LP models, and we illustrate how all these models attempt to compensate for the shortcomings of the conventional LP model, described in Section 11.3, when the input signal's tonal components are concentrated in the lower half of the Nyquist interval. In the first three alternative LP models, namely the constrained pole-zero LP (PZLP) model, the high-order LP (HOLP) model, and the pitch prediction (PLP) model, the influence of the input signal's frequency distribution is decreased by using a model different from the conventional low-order all-pole model. In the last two alternative LP models, namely the warped LP (WLP) model and the selective LP (SLP) model, the performance of the conventional low-order all-pole model is increased by first transforming the input signal such that its tonal components are spread in the entire Nyquist interval. As stated earlier, we will mainly focus on the alternative LP models, and not on how the model parameters can be estimated.

### 11.4.1 Constrained Pole-Zero LP Model

It is well known that, whereas a sum of  $N$  sinusoids can be exactly modeled using an AR( $2N$ ) model, a sum of  $N$  sinusoids plus white noise should be modeled using an ARMA( $2N, 2N$ ) model [21]-[24] with equal coefficients in the AR and MA parts, i.e., the zeros coinciding with the poles [23],[25]. This observation can be extended to a sum of (finite-bandwidth) damped sinusoids plus white noise, but in this case the zeros should be slightly displaced towards the origin, remaining on the same radial line as the poles [24], [25]. The LP model in (11.8) can then be simplified to a constrained pole-zero LP (PZLP) model with an equal number of poles and zeros,

$$G(z) = \prod_{l=1}^P \frac{(1 - 2\rho_l \cos \theta_l z^{-1} + \rho_l^2 z^{-2})}{(1 - 2\nu_l \cos \theta_l z^{-1} + \nu_l^2 z^{-2})} \quad (11.29)$$

the constraint being that the poles and zeros are on the same radial lines, i.e.,  $\zeta_l = \theta_l$ ,  $l = 1, \dots, P$ , with the poles positioned between the zeros and the unit circle, i.e.,  $0 \ll \rho_l < \nu_l \leq 1$ ,  $l = 1, \dots, P$ .

We now analyze the PZLP model performance for predicting tonal signals corresponding to the signal model (11.1), when  $P = N$ , by substituting the PEF magnitude response  $|H(e^{j\omega})|^2$ , obtained by inverting the magnitude response of  $G(z)$  in (11.29), in the LP criterion (11.14). First, we evaluate the second term of the LP criterion (11.14). Using the direct-form representation of the PZLP model in (11.6), with  $Q = P$  and  $b_0 = 1$ , the PEF magnitude response can be calculated as

$$|H(e^{j\omega})|^2 = \frac{|A(e^{j\omega})|^2}{|B(e^{j\omega})|^2} \quad (11.30)$$

$$= \frac{r_a(0) + 2 \sum_{i=1}^{2P} \cos(i\omega) r_a(i)}{r_b(0) + 2 \sum_{i=1}^{2P} \cos(i\omega) r_b(i)} \quad (11.31)$$

with  $r_a(i) = \sum_{p=i}^{2P} a_p a_{p-i}$  and  $r_b(i) = \sum_{p=i}^{2P} b_p b_{p-i}$  the autocorrelation functions of the PEF numerator and denominator coefficients, respectively. Note that when predicting tonal signals, the PEF poles and zeros are typically very close to the unit circle, and the PEF zeros are allowed to lie on the unit circle. We can then approximately state that the PEF pole radii are equal, i.e.,  $\rho_1 = \dots = \rho_P = \rho$  and likewise that the PEF zero radii are equal, i.e.,  $\nu_1 = \dots = \nu_P = \nu$ . In this case, the numerator and denominator of the PEF transfer function admit a particular structure, as shown in [31],

$$H(z) = \frac{1 + \nu g_1 z^{-1} + \dots + \nu^{P-1} g_{P-1} z^{-P+1} + \nu^P g_P z^{-P}}{1 + \rho g_1 z^{-1} + \dots + \rho^{P-1} g_{P-1} z^{-P+1} + \rho^P g_P z^{-P}} \cdots \quad (11.32)$$

$$\cdots \frac{+\nu^{P+1} g_{P-1} z^{-P-1} + \dots + \nu^{2P-1} g_1 z^{-2P+1} + \nu^{2P} z^{-2P}}{+\rho^{P+1} g_{P-1} z^{-P-1} + \dots + \rho^{2P-1} g_1 z^{-2P+1} + \rho^{2P} z^{-2P}}$$

and, as a consequence, the autocorrelation function of the PEF numerator coefficients can be rewritten, for  $i = 0, \dots, 2P$ , as

$$r_a(i) = \begin{cases} \sum_{p=0}^{P-i} g_p g_{p+i} (\nu^{2p+i} + \nu^{4P-(2p+i)}) + \sum_{p=1}^{\frac{i-1}{2}} g_{P-p} g_{P-i+p} (\nu^{2P-i} + \nu^{2P+i}), & i = \text{odd} \\ \sum_{p=0}^{P-i} g_p g_{p+i} (\nu^{2p+i} + \nu^{4P-(2p+i)}) + \sum_{p=1}^{\frac{i}{2}-1} g_{P-p} g_{P-i+p} (\nu^{2P-i} + \nu^{2P+i}) \\ \quad + g_{P-\frac{i}{2}}^2 \nu^{2P}, & i = \text{even} \end{cases} \quad (11.33)$$

and similarly for  $r_b(i)$ ,  $i = 0, \dots, 2P$ , by replacing  $\nu$  with  $\rho$  in (11.33). Since  $\nu$  and  $\rho$  are assumed to be close to 1, we can make the following approximations,

$$\begin{cases} \nu^{2p+i} + \nu^{4P-(2p+i)} \approx 2\nu^{2P} & i = 0, \dots, 2P, p = 0, \dots, P-i \quad (11.34) \\ \nu^{2P-i} + \nu^{2P+i} \approx 2\nu^{2P} & i = 0, \dots, 2P, p = 1, \dots, \lfloor \frac{i-1}{2} \rfloor \quad (11.35) \\ \rho^{2p+i} + \rho^{4P-(2p+i)} \approx 2\rho^{2P} & i = 0, \dots, 2P, p = 0, \dots, P-i \quad (11.36) \\ \rho^{2P-i} + \rho^{2P+i} \approx 2\rho^{2P} & i = 0, \dots, 2P, p = 1, \dots, \lfloor \frac{i-1}{2} \rfloor \quad (11.37) \end{cases}$$

where  $\lfloor x \rfloor$  denotes the floor function, which returns the highest integer less than or equal to  $x$ . We can hence rewrite  $r_a(i)$  in (11.33) and  $r_b(i)$  as

$$\begin{cases} r_a(i) = \nu^{2P} \gamma_i & i = 0, \dots, 2P \quad (11.38) \\ r_b(i) = \rho^{2P} \gamma_i & i = 0, \dots, 2P \quad (11.39) \end{cases}$$

with

$$\gamma_i = \begin{cases} 2 \sum_{p=0}^{P-i} g_p g_{p+i} + 2 \sum_{p=1}^{\frac{i-1}{2}} g_{P-p} g_{P-i+p}, & i = \text{odd} \\ 2 \sum_{p=0}^{P-i} g_p g_{p+i} + 2 \sum_{p=1}^{\frac{i}{2}-1} g_{P-p} g_{P-i+p} + g_{P-\frac{i}{2}}^2, & i = \text{even.} \end{cases} \quad (11.40)$$

Substituting (11.38) and (11.39) in (11.31) yields

$$|H(e^{j\omega})|^2 = \frac{\nu^{2P} \left( \gamma_0 + 2 \sum_{i=1}^{2P} \cos(i\omega) \gamma_i \right)}{\rho^{2P} \left( \gamma_0 + 2 \sum_{i=1}^{2P} \cos(i\omega) \gamma_i \right)} = \frac{\nu^{2P}}{\rho^{2P}} \quad (11.41)$$

which is expected to be a good approximation except in the close neighborhood of the PEF pole-zero angles  $\theta_l$ ,  $l = 1, \dots, P$ , where the PEF magnitude response approaches zero because the PEF zeros are closer to the unit circle than the poles. However, when integrating the PEF magnitude response over the entire frequency range  $[0, 2\pi)$ , the notches in  $|H(e^{j\omega})|^2$  at  $\omega = \theta_l$  are negligible, such that the second term in the LP criterion (11.14) can be written as

$$\frac{\sigma_r^2}{2\pi} \int_0^{2\pi} |H(e^{j\omega})|^2 d\omega = \sigma_r^2 \frac{\nu^{2P}}{\rho^{2P}}. \quad (11.42)$$

We now consider the minimization of the LP criterion (11.14) for the PZLP model (11.29), assuming that  $\nu_1 = \dots = \nu_P = \nu$  and  $\rho_1 = \dots = \rho_P = \rho$  with  $0 \ll \rho < \nu \leq 1$  and using the approximation (11.34)-(11.37) such that the result in (11.42) can be applied. Since  $\nu$  and  $\rho$  are close to each other, they cannot be treated as independent variables, and minimizing the LP criterion w.r.t.  $\nu$  and  $\rho$  can be achieved by setting the total derivative w.r.t.  $\nu$  and  $\rho$  to zero, which leads to the following system of equations,

$$\left\{ \begin{array}{l} \frac{\partial V(\boldsymbol{\xi})}{\partial \theta_l} = \sum_{n=1}^N \frac{\alpha_n^2}{2} \left[ \frac{\partial}{\partial \theta_l} |H(e^{j\omega})|^2 \right]_{\omega=\omega_n} + \frac{\partial}{\partial \theta_l} \left( \sigma_r^2 \frac{\nu^{2P}}{\rho^{2P}} \right) = 0, \quad l = 1, \dots, P \end{array} \right. \quad (11.43)$$

$$\left\{ \begin{array}{l} \frac{dV(\boldsymbol{\xi})}{d\nu} = \frac{\partial V(\boldsymbol{\xi})}{\partial \nu} + \frac{\partial V(\boldsymbol{\xi})}{\partial \rho} \frac{d\rho}{d\nu} = 0 \end{array} \right. \quad (11.44)$$

$$\left\{ \begin{array}{l} \frac{dV(\boldsymbol{\xi})}{d\rho} = \frac{\partial V(\boldsymbol{\xi})}{\partial \rho} + \frac{\partial V(\boldsymbol{\xi})}{\partial \nu} \frac{d\nu}{d\rho} = 0 \end{array} \right. \quad (11.45)$$

with

$$\frac{\partial V(\boldsymbol{\xi})}{\partial \nu} = \sum_{n=1}^N \frac{\alpha_n^2}{2} \left[ \frac{\partial}{\partial \nu} |H(e^{j\omega})|^2 \right]_{\omega=\omega_n} + \frac{\partial}{\partial \nu} \left( \sigma_r^2 \frac{\nu^{2P}}{\rho^{2P}} \right) = 0 \quad (11.46)$$

$$\frac{\partial V(\xi)}{\partial \rho} = \sum_{n=1}^N \frac{\alpha_n^2}{2} \left[ \frac{\partial}{\partial \rho} |H(e^{j\omega})|^2 \right]_{\omega=\omega_n} + \frac{\partial}{\partial \rho} \left( \sigma_r^2 \frac{\nu^{2P}}{\rho^{2P}} \right) = 0. \quad (11.47)$$

Since  $\nu$  and  $\rho$  are close to each other, we can assume

$$\frac{d\rho}{d\nu} \approx \frac{d\nu}{d\rho} \approx 1 \quad (11.48)$$

and moreover

$$\frac{\partial}{\partial \nu} \left( \sigma_r^2 \frac{\nu^{2P}}{\rho^{2P}} \right) \approx -\frac{\partial}{\partial \rho} \left( \sigma_r^2 \frac{\nu^{2P}}{\rho^{2P}} \right). \quad (11.49)$$

Substituting (11.46)-(11.49) in (11.44) and (11.45), and noting that the expression in (11.42) does not depend on the PEF pole-zero angles  $\theta_l$ , we can see that all the terms in the system of equations (11.43)-(11.45) that are due to the noise component in the observed signal cancel out. In other words, if the PEF poles and zeros are close to the unit circle, then the solution to the LP estimation problem using the PZLP model is insensitive to (white) noise in the observed signal. This is the main strength of the PZLP model as compared to the conventional LP model, which was shown in Section 11.3 to be much more sensitive to noise when predicting tonal signals.

It remains to show that the PEF angles calculated from (11.43)-(11.45) converge to the frequencies of the tonal components. The PZLP PEF magnitude response and its partial derivatives w.r.t.  $\theta_l$ ,  $l = 1, \dots, P$ ,  $\nu$ , and  $\rho$  can be calculated as

$$|H(e^{j\omega})|^2 = \prod_{l=1}^P \frac{|A_l(e^{j\omega})|^2}{|B_l(e^{j\omega})|^2} \quad (11.50)$$

$$\begin{aligned} &= \prod_{l=1}^P \frac{(1 - \nu^2)^2 + 4\nu^2(\cos \omega - \cos \theta_l)^2 - 4\nu(1 - \nu)^2 \cos \theta_l \cos \omega}{(1 - \rho^2)^2 + 4\rho^2(\cos \omega - \cos \theta_l)^2 - 4\rho(1 - \rho)^2 \cos \theta_l \cos \omega} \\ \frac{\partial}{\partial \theta_l} |H(e^{j\omega})|^2 &= \frac{|B_l(e^{j\omega})|^2 \frac{\partial}{\partial \theta_l} |A_l(e^{j\omega})|^2 - |A_l(e^{j\omega})|^2 \frac{\partial}{\partial \theta_l} |B_l(e^{j\omega})|^2}{|B_l(e^{j\omega})|^4} \\ &\quad \times \prod_{\substack{k=1 \\ k \neq l}}^P \frac{|A_k(e^{j\omega})|^2}{|B_k(e^{j\omega})|^2} \end{aligned} \quad (11.51)$$

$$\frac{\partial}{\partial \nu} |H(e^{j\omega})|^2 = \sum_{l=1}^P \left\{ \frac{\frac{\partial}{\partial \nu} |A_l(e^{j\omega})|^2}{|B_l(e^{j\omega})|^2} \prod_{\substack{k=1 \\ k \neq l}}^P \frac{|A_k(e^{j\omega})|^2}{|B_k(e^{j\omega})|^2} \right\} \quad (11.52)$$

$$\frac{\partial}{\partial \rho} |H(e^{j\omega})|^2 = -\sum_{l=1}^P \left\{ \frac{|A_l(e^{j\omega})|^2 \frac{\partial}{\partial \rho} |B_l(e^{j\omega})|^2}{|B_l(e^{j\omega})|^4} \prod_{\substack{k=1 \\ k \neq l}}^P \frac{|A_k(e^{j\omega})|^2}{|B_k(e^{j\omega})|^2} \right\} \quad (11.53)$$

with

$$\frac{\partial}{\partial \theta_l} |A_l(e^{j\omega})|^2 = 4\nu \sin \theta_l [(1 + \nu^2) \cos \omega - 2\nu \cos \theta_l] \quad (11.54)$$



$$\frac{\partial}{\partial \theta_l} |B_l(e^{j\omega})|^2 = 4\rho \sin \theta_l [(1 + \rho^2) \cos \omega - 2\rho \cos \theta_l] \quad (11.55)$$

$$\frac{\partial}{\partial \nu} |A_l(e^{j\omega})|^2 = 4[2\nu(\cos \omega - \cos \theta_l)^2 - (1 - \nu)(1 - 3\nu) \cos \theta_l \cos \omega - \nu(1 - \nu^2)]$$

$$\frac{\partial}{\partial \rho} |B_l(e^{j\omega})|^2 = 4[2\rho(\cos \omega - \cos \theta_l)^2 - (1 - \rho)(1 - 3\rho) \cos \theta_l \cos \omega - \rho(1 - \rho^2)].$$

The global minimum of (11.14) with  $P = N$ , corresponding to  $V(\boldsymbol{\xi}) = \sigma_r^2$ , is obtained when

$$\begin{cases} |A_l(e^{j\omega_l})|^2 = 0, & l = 1, \dots, P & (11.56) \\ \frac{\partial}{\partial \theta_l} |A_l(e^{j\omega_l})|^2 = 0, & l = 1, \dots, P & (11.57) \\ \frac{\partial}{\partial \nu} |A_l(e^{j\omega_l})|^2 = 0, & & (11.58) \end{cases}$$

or, equivalently,

$$\begin{cases} \theta_l = \omega_l, & l = 1, \dots, P & (11.59) \\ \nu = 1 & & (11.60) \end{cases}$$

and hence, following the assumption that the PEF poles are close to the zeros,  $\rho \rightarrow 1$ .

**Example 11.3** [Constrained pole-zero LP of synthetic audio signal] The PZLP model parameters can be estimated, either using an adaptive notch filtering (ANF) algorithm, for which several implementations have been suggested [24],[25],[31]-[35], or using the constrained pole-zero linear prediction (CPZLP) algorithm for multi-tone frequency estimation [36],[37]. Alternatively, if the PEF pole and zero radii are fixed a priori, any existing frequency estimation algorithm may be used to estimate the unknown PEF angles. When harmonicity can be assumed, i.e., for monophonic audio signals, an adaptive comb filter (ACF) may be a useful alternative to the ANF, as it relies on only one unknown parameter (i.e., the fundamental frequency) [32],[35]. Similarly, a comb-filter-based variant of the CPZLP algorithm has been described in [37].

Figs. 11.4(a) and 11.4(b) show the PEF pole-zero plot and magnitude response of a PZLP model of the synthetic audio signal introduced in Example 11.1, and with additive Gaussian white noise (SNR = 25 dB). The PZLP model parameters were calculated using the CPZLP algorithm with a comb filter model [37] of order  $2P = 30$ , pole radius  $\rho = 0.95$ , and zero radius  $\nu = 1$ , and with a numerical line search method using the BFGS quasi-Newton algorithm with initial fundamental frequency estimate  $\omega_0^{(0)} = 0.001$  and line search parameters as suggested in [36]. It can be seen that the PEF magnitude response exhibits a notch filter behavior at the frequencies of the tonal components, while being approximately flat in the remainder of the Nyquist interval.

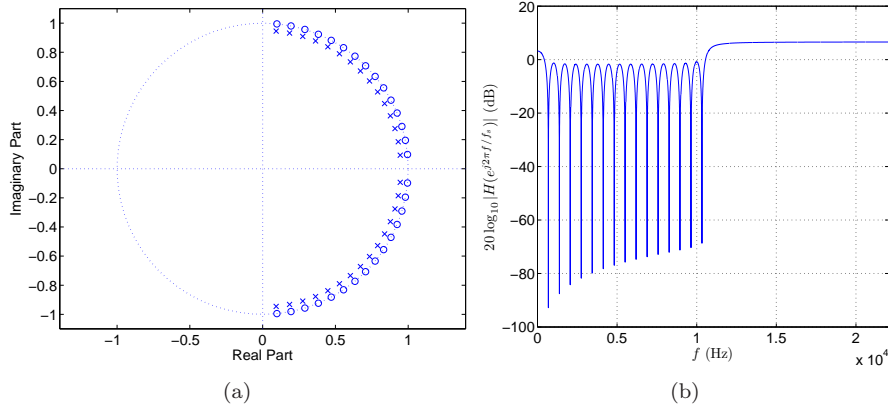


Figure 11.4: Constrained pole-zero LP model of synthetic audio signal plus noise (SNR = 25 dB) with order  $2P = 30$  and CPZLP algorithm: (a) PEF pole-zero plot, (b) PEF magnitude response.

### 11.4.2 High-Order LP model

It is well known that a pole-zero model can be arbitrarily closely approximated with an all-pole model, provided that the model order is chosen large enough. This means that a noisy sum of sinusoids can also be modeled using a high-order all-pole model instead of a pole-zero model [22]. In Section 11.3, the LP minimization problem (11.14) was analyzed for the case of an all-pole model of order  $P = N$ . When noise is present in the observed signal, the LP solution was shown to be a compromise between cancelling the tonal components and maintaining a flat high-frequency residual spectrum. By increasing the model order, the density of the zeros near the unit circle is increased accordingly, and hence the frequency resolution in the tonal components frequency range improves without sacrificing high-frequency residual spectral flatness. However, as the LP model order  $2P$  approaches the observation window length  $L$ , the variance of the estimated model parameters may be unacceptably large, leading to spurious peaks in the signal's spectral estimate [22]. It has been suggested that the order  $2P$  of a high-order LP (HOLP) model should be chosen in the interval  $L/3 \leq 2P \leq L/2$  to obtain the best spectral estimate for a noisy sum of sinusoids [22],[46].

**Example 11.4** [High-order LP of synthetic audio signal] Performing a  $L/2 = 1024$ th order LP of the noisy synthetic audio signal fragment defined before, using the autocorrelation method to estimate the model parameters, we obtain a PEF pole-zero plot and magnitude response as shown in Figs. 11.5(a) and 11.5(b). Examining the distribution of the PEF zeros in the complex plane reveals that this approach produces approximately  $1024 - 2N$  zeros, lying on and

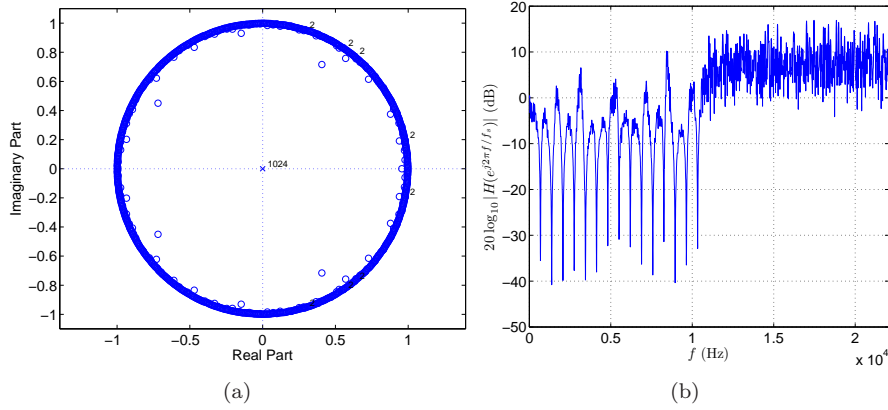


Figure 11.5: High-order LP model of synthetic audio signal plus noise (SNR = 25 dB) with order  $2P = 1024$  and autocorrelation method: (a) PEF pole-zero plot, (b) PEF magnitude response.

nearly equally spaced around the unit circle (to provide overall spectral flatness of the PEF magnitude response), and  $2N$  additional zeros at the frequencies  $\pm n\omega_0, n = 1, \dots, N$  of the tonal components (to provide the notch filter behavior). Note that when applying the covariance method to the estimation of the HOLP model parameters, a similar result is obtained.

### 11.4.3 Pitch Prediction Model

In LP of speech signals, the conventional LP model is usually cascaded with a so-called pitch prediction (PLP) model, with the aim of removing the long-term correlation from the signal. This technique can also be used to remove the (quasi-)periodicity from monophonic audio signals, since it implicitly relies on the harmonicity of the observed signal. If we consider a sum of harmonic sinusoids having a pitch period  $T_0$  that corresponds to an integer number of sampling periods  $KT_s$ , where  $K$  is referred to as the pitch lag, then perfect prediction can be obtained by using a one-tap pitch predictor, of which the PEF transfer function is given by

$$H(z) = 1 - z^{-K} = 1 - z^{-T_0/T_s} = 1 - z^{-2\pi/\omega_0}. \quad (11.61)$$

The PEF magnitude response corresponding to (11.61) is

$$|H(e^{j\omega})|^2 = 2 \left[ 1 - \cos\left(\frac{2\pi\omega}{\omega_0}\right) \right]. \quad (11.62)$$

It can be seen that  $|H(e^{j\omega})|^2 = 0$  at  $\omega = k\omega_0, \forall k \in \mathbb{Z}$ , which corresponds to a comb filter behavior, i.e., the PEF zeros are positioned on and equally

spaced around the unit circle, at angles corresponding to integer multiples of the fundamental frequency  $\omega_0$ . In other words, referring to the analysis in Section 11.3, the requirements of having the PEF zeros on the unit circle at angles  $n\omega_0$ ,  $n = 1, \dots, N$  (for cancelling the tonal components) and uniformly distributed on the unit circle (for maintaining the LP residual spectral flatness) are both fulfilled with the PLP model in (11.61).

However, for the PLP model to be capable of producing a good spectral estimate of a monophonic audio signal, we should improve the model in (11.61) in two ways. First of all, in audio signals the amplitudes of the harmonics  $n\omega_0$  typically decrease with increasing  $n$  (see, e.g., Figs. 11.11(b) and 11.14(b) in Section 11.5). This effect requires the PEF magnitude response to be spectrally shaped such that the comb filter notch depth decreases for increasing frequency. This can be achieved by using a multi-tap PLP model [47], which features multiple non-zero filter coefficients centered around the pitch lag value. In speech processing, a 3-tap PLP model is often applied, since this configuration usually provides enough flexibility in terms of spectral shaping,

$$H(z) = 1 + a_{K-1}z^{-(K-1)} + a_K z^{-K} + a_{K+1}z^{-(K+1)}. \quad (11.63)$$

From the 3-tap PEF magnitude response

$$\begin{aligned} |H(e^{j\omega})|^2 &= (\cos K\omega + a_K + (a_{K-1} + a_{K+1}) \cos \omega)^2 \\ &\quad + (\sin K\omega + (a_{K-1} - a_{K+1}) \sin \omega)^2 \end{aligned} \quad (11.64)$$

it can be derived that the desired spectral shaping for our application, i.e., a decreasing notch depth for increasing frequency, is obtained when  $-1 \leq a_K < (a_{K-1} + a_{K+1}) < 0$  [47].

Secondly, the PLP model in (11.62) is based on the assumption that the pitch lag  $K = T_0/T_s$  is an integer number, which is generally not the case. Non-integer pitch lags can be incorporated in the PLP model in two ways: either by using a multi-tap PLP model for interpolation (see, e.g., [2]), or by using a fractional delay filter [48], for which numerous design methods exist [49]. We prefer to combine both approaches, such that the multi-tap structure may be primarily used for spectral shaping, whereas interpolation for non-integer pitch lags is achieved with a fractional delay filter. A combined fractional multi-tap PLP model has been proposed in [47], with

$$H(z) = 1 + \sum_{l=K-1}^{K+1} a_l z^{-l} \left( \sum_{i=-I}^{I-1} w_h(I + f/D) \text{sinc}(I + f/D) z^i \right). \quad (11.65)$$

The fractional delay interpolation filter is a Hamming-windowed, truncated (length- $2I$ ) approximation of the ideal sinc-like interpolation filter [49], with  $w_h(t)$  denoting the Hamming window (centered at  $t = 0$ ). In (11.65),  $D$  is the interpolation ratio (where  $1/D$  is referred to as the pitch resolution), and  $f = 0, 1, \dots, D - 1$  denotes the fractional phase.

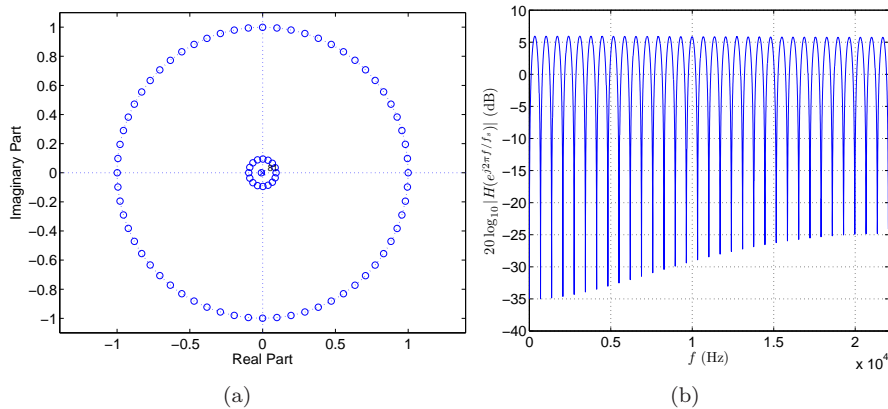


Figure 11.6: Fractional 3-tap PLP model of synthetic audio signal plus noise (SNR = 25 dB): (a) PEF pole-zero plot, (b) PEF magnitude response.

Typically, for estimating the PLP model parameters, in a first step the optimal pitch lag  $K$  and fractional phase  $f$  are estimated by an exhaustive search of the minimal fractional 1-tap PLP residual power over the interval  $K \in [K_{\min}, K_{\max}]$  and  $f \in [0, D - 1]$ . In speech analysis, the pitch lag limits correspond to the highest-pitched (female) and lowest-pitched (male) voices being analyzed and are typically chosen in the range  $K_{\min} = 20, \dots, 40$  and  $K_{\max} = 120, \dots, 160$  samples, at  $f_s = 8$  kHz. For pitch analysis of audio signals, we propose to set the pitch lag range such that it corresponds to a fundamental frequency range of  $100, \dots, 1000$  Hz, i.e., at  $f_s = 44.1$  kHz,  $K \in [44, 441]$ . In a second step, the fractional 3-tap PLP model parameters  $a_l$ ,  $l \in [K - 1, K + 1]$  are estimated using the estimated pitch lag and fractional phase from the first step. Some useful approximations for efficiently calculating the 3-tap PLP model parameters from the input signal autocorrelation function have been suggested in [2].

**Example 11.5** [Pitch prediction of synthetic audio signal] The parameters of the fractional 3-tap PLP model given in (11.65) were estimated for the noisy synthetic audio signal defined earlier using the method proposed in [47], with an interpolation filter of length  $2I = 32$  and an interpolation ratio  $D = 8$ , and by forcing the input correlation matrix to be Toeplitz [2]. The resulting PEF magnitude response and pole-zero plot are shown in Figs. 11.6(a) and 11.6(b). Note the additional circle of zeros around the origin in Fig. 11.6(a), which is due to the fractional part of the PEF transfer function, and the spectral shaping effect in Fig. 11.6(b), which is obtained by using multiple taps in the PLP model.

### 11.4.4 Warped LP Model

Warped linear prediction (WLP) is probably the most well-known technique for LP of audio signals, see [12] and references therein. In WLP, the input signal undergoes a non-uniform frequency transformation before a conventional LP is performed, with the aim of enhancing the frequency resolution in certain frequency regions. The frequency transformation is usually defined by an all-pass bilinear transform in the  $z$ -domain, which maps the unit circle onto itself,

$$z^{-1} \mapsto \tilde{z}^{-1} = \frac{z^{-1} - \lambda}{1 - \lambda z^{-1}}. \quad (11.66)$$

The so-called warping parameter  $\lambda$  is typically chosen such that the corresponding frequency mapping

$$\omega \mapsto \tilde{\omega} = \omega + 2 \arctan\left(\frac{\lambda \sin \omega}{1 - \lambda \cos \omega}\right) \quad (11.67)$$

approximates the Bark auditory scale [30], i.e., when the sampling rate  $f_s$  is expressed in kHz,

$$\lambda_{\text{Bark}}(f_s) = 1.0674 \sqrt{\frac{2}{\pi} \arctan(0.06583 f_s)} - 0.1916. \quad (11.68)$$

Since  $\lambda_{\text{Bark}}(44.1) > 0$ , the warping operation tends to spread out the tonal components in the observed signal over the entire Nyquist interval. From the conventional LP analysis in Section 11.3, it can hence be expected that applying a conventional, i.e., low-order all-pole LP model to the warped signal will yield a better prediction than a conventional LP model of the original signal. The optimal prediction is obtained when the frequency transformation produces a uniform spreading of the tonal components in the Nyquist interval. For monophonic audio signals this is never the case, since the bilinear frequency warping in (11.66)-(11.67) disturbs the harmonicity of the signal. For this class of signals, the frequency transformation of the selective LP model described in Section 11.4.5 appears to be better suited. However, for polyphonic audio signals, the above bilinear frequency warping may be a near-optimal mapping, since in this case the different fundamental frequencies are approximately related to each other according to the Bark scale (see also the simulation results in Section 11.5.3).

**Example 11.6** [Warped LP of synthetic audio signal] The warped spectrum of the noisy synthetic audio signal defined before, is shown in Fig. 11.7(a) for  $\lambda = \lambda_{\text{Bark}}(44.1) = 0.75641$ . Figs. 11.7(b) and 11.7(c) illustrate the PEF pole-zero plot and magnitude response on a warped frequency scale  $\tilde{f} = \tilde{\omega}(f_s/2\pi)$ , when a  $2N$ th order WLP model is calculated using the autocorrelation method. The frequency resolution of the signal's WLP spectral estimate is very good for

the five lowest tonal components  $n\omega_0$ ,  $n = 1, \dots, 5$ , while the higher harmonics are modeled less accurately because they are too closely spaced on the warped frequency scale. The PEF transfer function can be unwarped to the original frequency scale, but then the PEF impulse response is of infinite duration. The PEF pole-zero plot and magnitude response on the original frequency scale, obtained by truncating the unwarped PEF impulse response to a length of  $L/4 = 512$  samples, are shown in Figs. 11.7(d) and 11.7(e). The pole-zero plot on the original frequency scale clearly illustrates that the WLP model succeeds both at cancelling the (low-frequency) tonal components (by placing a few zeros approximately on the unit circle at the lower tonal component frequencies), and at preserving the overall spectral flatness of the residual (by placing a large number of zeros uniformly spaced around and close to the unit circle).

Note that the WLP residual  $e(t, \xi)$  can be calculated without unwarping the PEF transfer function, but instead by considering the PEF as a warped FIR filter [50]. Moreover, before feeding the WLP residual to a synthesis filter or calculating its spectral flatness (see Section 11.5), it should be post-filtered with a high-pass filter defined as [12]

$$D_0^{-1}(z) = \frac{1 - \lambda z^{-1}}{\sqrt{1 - \lambda^2}}. \quad (11.69)$$

### 11.4.5 Selective LP Model

In some cases, e.g., when dealing with monophonic audio signals, a uniform frequency mapping may be more useful than a non-uniform mapping such as the warping operation described in Section 11.4.4, since it preserves the harmonic relation between the tonal components. A uniform mapping which allows to “zoom in” on a certain frequency region  $\omega_1 \leq \omega \leq \omega_2$ , is accomplished by

$$\omega \mapsto \tilde{\omega} = \pi \frac{\omega - \omega_1}{\omega_2 - \omega_1} \quad (11.70)$$

which, when combined with a conventional LP model, is known as a selective LP (SLP) model [1].

To obtain a uniform spreading of the tonal components over the entire Nyquist interval, we should choose  $\omega_1 = 0$  and  $\omega_2 = \omega_1 + \omega_N$ , with  $\omega_1$  and  $\omega_N$  the frequencies of the lowest and highest tonal components, see (11.1). This leads to

$$\omega \mapsto \tilde{\omega} = \Gamma \omega \quad (11.71)$$

with

$$\Gamma = \frac{\pi}{\omega_1 + \omega_N}. \quad (11.72)$$

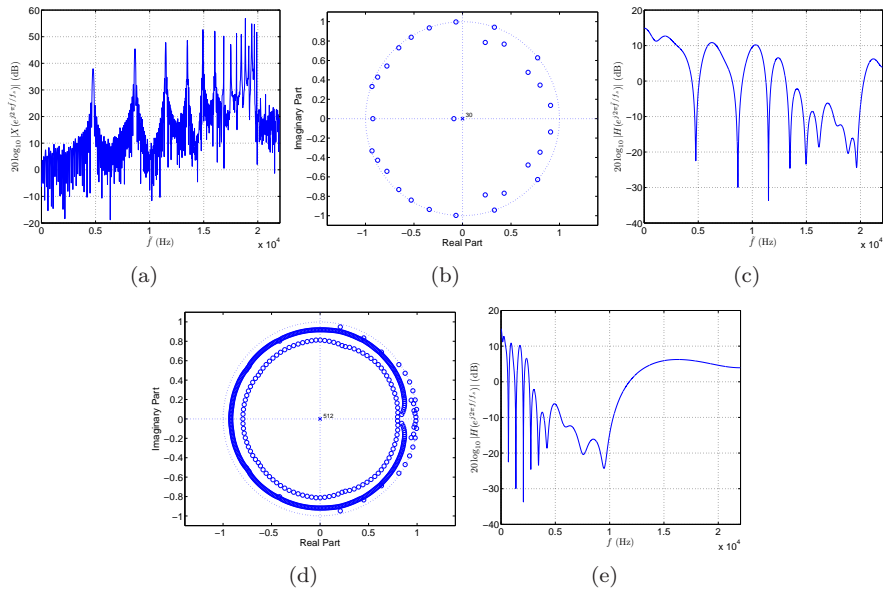


Figure 11.7: Warped LP model of synthetic audio signal plus noise (SNR = 25 dB) with order  $2P = 30$ , warping parameter  $\lambda = \lambda_{\text{Bark}}(44.1)$ , and autocorrelation method: (a) Noisy synthetic audio signal magnitude spectrum (warped scale), (b) PEF pole-zero plot (warped scale), (c) PEF magnitude response (warped scale), (d) PEF pole-zero plot (original scale), (e) PEF magnitude response (original scale).



In the  $z$ -domain this corresponds to the mapping

$$z^{-1} \mapsto \tilde{z}^{-1} = z^{-\Gamma} \quad (11.73)$$

which is a downsampling operation with downsampling factor  $\Gamma$ . In the case of a monophonic audio signal, the downsampling factor can be rewritten using (11.2),

$$\Gamma = \frac{\pi}{(N+1)\omega_0} = \frac{f_s}{2(N+1)f_0} \quad (11.74)$$

and in the polyphonic case, using (11.3),

$$\Gamma = \frac{\pi}{\omega_{0,1} + M_N \omega_{0,N}} = \frac{f_s}{2(f_{0,1} + M_N f_{0,N})}. \quad (11.75)$$

Note that the optimal downsampling factor  $\Gamma$ , given in (11.72), is highly signal-dependent, and non-integer downsampling is required in general. These difficulties can be easily avoided by using an approximate, integer downsampling factor (see Section 11.5) which is chosen to be fixed for the entire signal analysis. It should then typically be chosen in the range  $\Gamma = 2, \dots, 10$ , if possible using some prior knowledge about the frequency range of the instrument generating the audio signal being analyzed.

**Example 11.7** [Selective LP of synthetic audio signal] The spectrum of the noisy synthetic audio signal defined before, downsampled with a factor  $\Gamma = 2$  (obtained from (11.74) with  $\omega_0 = 2\pi/64$  and  $N = 15$ ), is shown in Fig. 11.8(a), and the PEF pole-zero plot and magnitude response, resulting from using a  $2N$ th order SLP model, calculated with the autocorrelation method, are plotted on the downsampled frequency scale in Figs. 11.8(b) and 11.8(c). The PEF zeros are nearly perfectly distributed in a uniform way around the unit circle, with exactly one complex conjugate zero pair for each tonal component in the downsampled signal. After upsampling, the PEF pole-zero plot and magnitude response shown in Figs. 11.8(d) and 11.8(e) are obtained. The PEF behavior on the original frequency scale is comparable to the PLP model PEF behavior, i.e., nearly perfect cancellation of the tonal components is achieved, at the cost of having additional notches in the upper half of the Nyquist interval, which may result in a non-smooth high-frequency residual spectrum. The LP residual can either be calculated on the downsampled or on the original time scale.

## 11.5 Simulation Results

In this section, we evaluate the conventional and alternative LP models described in Sections 11.3 and 11.4 in terms of frequency estimation accuracy,

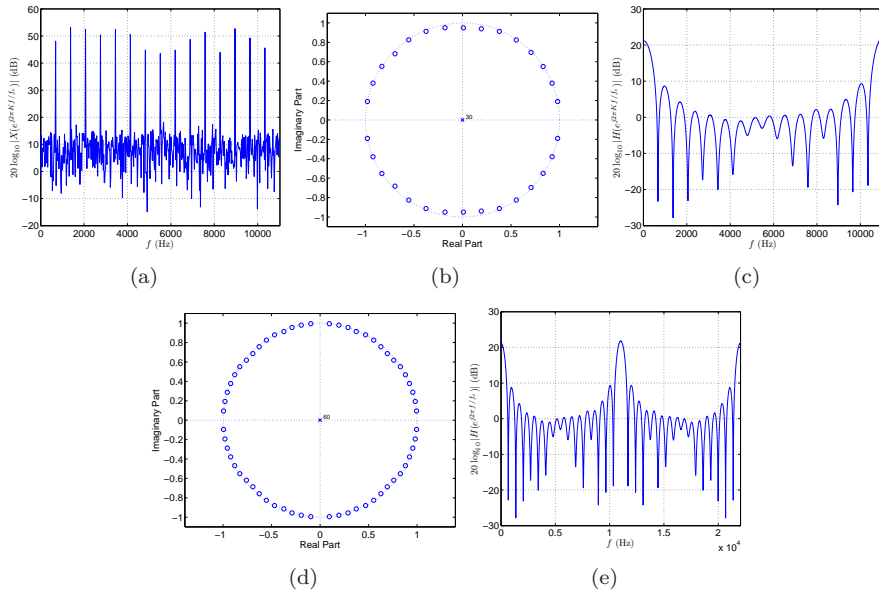


Figure 11.8: Selective LP model of synthetic audio signal plus noise (SNR = 25 dB) with order  $2P = 30$ , downsampling factor  $\Gamma = 2$ , and autocorrelation method: (a) Noisy synthetic audio signal magnitude spectrum (downsampled scale), (b) PEF pole-zero plot (downsampled scale), (c) PEF magnitude response (downsampled scale), (d) PEF pole-zero plot (original scale), (e) PEF magnitude response (original scale).

residual spectral flatness, and perceptual frequency resolution for a synthetic harmonic audio signal with varying fundamental frequency and SNR. Afterwards, we apply the different LP models to true monophonic and polyphonic audio signals, and we analyze the PEF behavior by examining the pole-zero plots and magnitude responses. Residual spectral flatness figures are given for true audio signals as a function of pitch and time offset of the analysis window within the signal.

We should stress that the aim is to compare different LP models, and not the algorithms that can be used to estimate the model parameters. Some models come with parameter estimation algorithms that are well established (e.g., covariance method or autocorrelation method with Levinson-Durbin algorithm [51, Ch. 6] for all-pole models), yet other models do not. In particular, PZLP models typically result in a non-convex parameter estimation problem that is solved either in an adaptive or iterative way. As a consequence, the performance of the corresponding estimation algorithms (e.g., ANF or CPZLP) depends heavily on the initial conditions. In the simulation results presented below, the initial conditions are chosen in the neighborhood of the true fundamental frequencies in the observed audio signal, such that the PZLP estimation algorithms yield a solution that corresponds with high probability to the global solution. In this way, the emphasis is on the model performance rather than on the estimation algorithm performance. For the same reason, knowledge of the true fundamental frequencies is also assumed when determining the optimal downsampling factor in the SLP estimation algorithms, and for designing a PLP model for polyphonic audio signals. For the conventional LP model, the performance may differ substantially for the autocorrelation and covariance estimation methods, hence the results for both methods are included.

### 11.5.1 Synthetic Audio Signal

Throughout Ex. 11.2-11.7, the performance of conventional and alternative LP models was illustrated by inspecting the PEF pole-zero plots and magnitude responses, resulting from the prediction of a noisy synthetic audio signal with fundamental frequency  $f_0 \approx 689.1$  Hz and SNR = 25 dB. We also present a more quantitative evaluation of the different LP models, for a synthetic audio signal with variable fundamental frequency and SNR.

A first performance measure is the Mean Square Frequency Error (MSFE), which is defined with the aim of evaluating the frequency estimation accuracy of the different LP models,

$$\text{MSFE} = \frac{1}{N} \sum_{n=1}^N (\theta_{l(n)} - \omega_n)^2 \quad (11.76)$$

with

$$l(n) = \arg \min_l \|\nu_l e^{j\theta_l} - e^{j\omega_n}\|^2 \quad (11.77)$$

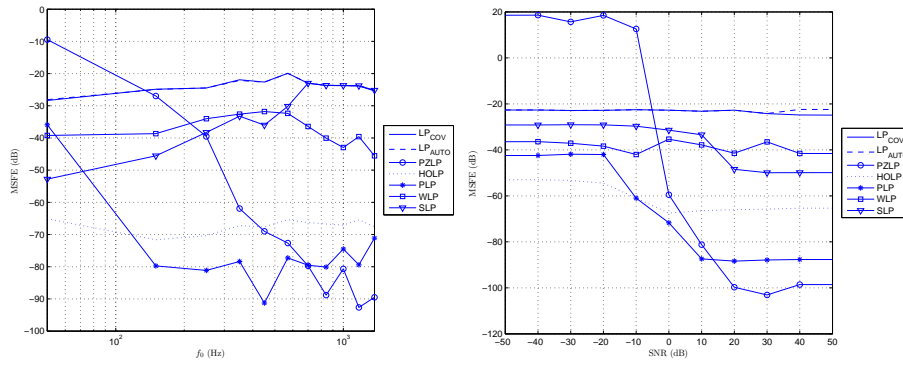
$$= \arg \min_l (1 + \nu_l^2 - 2\nu_l \cos(\theta_l - \omega_n)). \quad (11.78)$$

In other words, the MSFE is calculated as the mean square difference between each of the frequencies  $\omega_n$  of the  $N$  tonal components in the observed signal, and the angle of the PEF zero  $\nu_{l(n)} e^{j\theta_{l(n)}}$  that is closest to the point  $e^{j\omega_n}$  in the complex plane. The MSFE was calculated for a synthetic audio signal with  $N$ ,  $L$ ,  $f_s$ ,  $\alpha_n$ , and  $\phi_n$  as in Ex. 11.2, with additive Gaussian white noise resulting in an SNR = 25 dB, and with varying fundamental frequency, equal to the first 11 center frequencies of the Bark scale [52]. A second experiment was conducted with similar signals having a fixed fundamental frequency  $f_0 \approx 689.1$  Hz, and an SNR varying between -50 dB and 50 dB in steps of 10 dB. The MSFE results, averaged over 100 Monte Carlo trials for different realizations of the Gaussian white noise sequence, are shown in Figs. 11.9(a) and 11.9(b). The MSFE of the low-order all-pole models (LP<sub>AUTO</sub>, LP<sub>COV</sub>, WLP, and SLP) appears to be more or less invariant w.r.t. varying fundamental frequency and SNR, with MSFE values varying between -50 and -20 dB, the highest of which is obtained with the conventional LP model. It can be observed that models for which the PEF zeros are on (PLP and PZLP) or very close to (HOLP) the unit circle generally provide a higher frequency estimation accuracy. The HOLP model produces MSFE values between -70 and -50 dB, which are invariant with varying fundamental frequency, and slightly lower for high than for low SNRs. At sufficiently high fundamental frequency and SNR values, the PLP and PZLP models achieve an MSFE as low as -90 (PLP) and -100 (PZLP) dB. However, the PLP and PZLP model MSFE performance is seen to be worse for lower fundamental frequencies and SNR values. The sensitivity of these models to the fundamental frequency is presumably related to the fact that these are the only models that explicitly rely on the harmonicity of the observed signal (since in the PZLP case, the comb filter model is used). The performance drop of the PZLP model at low SNR values is due to the accuracy of the CPZLP algorithm, which is known to be relatively poor at SNR values below -5 dB [37].

A second performance measure is the Spectral Flatness Measure (SFM) of the LP residual, defined as [43, Ch. 6]

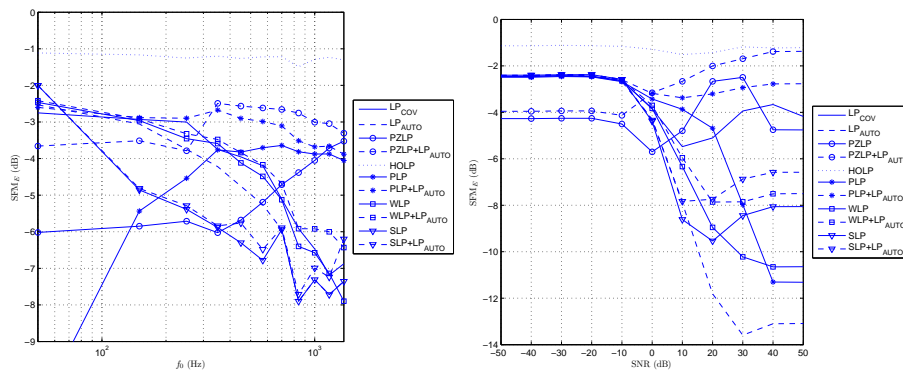
$$\text{SFM}_E = \frac{\exp\left[\frac{1}{L} \sum_{k=0}^{L-1} \ln |E(e^{j\frac{2\pi k}{L}}, \xi)|\right]}{\frac{1}{L} \sum_{k=0}^{L-1} |E(e^{j\frac{2\pi k}{L}}, \xi)|} \quad (11.79)$$

with  $E(e^{j\frac{2\pi k}{L}}, \xi)$ ,  $k = 0, \dots, L-1$  the  $L$ -point DFT of the LP residual  $e(t, \xi)$ . The SFM is a real number between 0 and 1, with SFM = 1 corresponding to a



(a) MSFE (dB) vs. fundamental frequency  $f_0$  (Hz)

(b) MSFE (dB) vs. SNR (dB)



(c) Residual SFM (dB) vs. fundamental frequency  $f_0$  (Hz)

(d) Residual SFM (dB) vs. SNR (dB)

Figure 11.9: Mean square frequency error (MSFE) and residual SFM curves of Monte Carlo simulations for a synthetic audio signal with variable fundamental frequency and SNR.

flat spectrum, and is often expressed on a dB-scale (0 dB corresponding to a flat spectrum). Monte Carlo simulation results of the residual SFM after prediction of the synthetic audio signals with varying fundamental frequency and SNR described above, are shown in Figs. 11.9(c) and 11.9(d). The residual SFM of the low-order all-pole models ( $LP_{\text{AUTO}}$ ,  $LP_{\text{COV}}$ , WLP, and SLP) decreases with increasing fundamental frequency and increasing SNR. The first observation can be explained by noting that at low fundamental frequency values, the low-order all-pole models tend to model multiple tonal components with one complex conjugate pole pair, while the remaining poles are used to model the high-frequency noise spectrum. As a consequence, most of the poles are located relatively far away from the unit circle, hence resulting in a smoother spectral behavior. The residual SFM drop at high SNR values should not be surprising, since the low-order all-zero PEFs generally do not succeed at completely cancelling the tonal components from the observed signal. On the other hand, the residual SFM of the PLP and PZLP models can be seen to increase with increasing fundamental frequency and decreases (PLP) or remains quasi constant (PZLP) with increasing SNR. The HOLP model residual SFM is the highest among all LP models, and appears to be independent of both fundamental frequency and SNR. The SFM of the synthetic audio signals before LP was on average -10 dB in the varying fundamental frequency case, and -35 dB in the varying SNR case. A relevant extension to the low-order alternative LP models described in Section 11.4, is to cascade them with a conventional LP model. Such a cascaded model can be motivated by noting that for true audio signals, the noise term in the tonal signal models (11.1)-(11.3) may be non-white. Hence an alternative LP model could be applied first for predicting the tonal components, and in a second step a conventional LP model could be used for whitening both the noise and the unpredicted tonal components in the residual of the alternative LP model. This cascaded structure appears to be beneficial for the low-order alternative LP models (PZLP, PLP, WLP, and SLP) in terms of increasing the residual SFM, especially at high SNR values and, for the PZLP and PLP models, also at low fundamental frequency values.

Finally, the third performance measure we will use is the Inter-peak Dip Depth (IDD) [12], a perceptually motivated measure which reflects the separability of spectral peaks for a certain model. It is defined for an LP model of a length- $L$  sum of two sinusoids at frequencies  $f_1$  and  $f_2$  Hz, separated by two times the equivalent rectangular bandwidth (ERB) [53] at the lower frequency  $f_1$ , i.e.,  $f_2 = f_1 + 2(0.108f_1 + 24.7)$ , as

$$\text{IDD} = \frac{L_1 + L_2}{2L_d} \quad (11.80)$$

with  $L_1$  and  $L_2$  corresponding to the amplitude of the two peaks in the LP model magnitude response, and  $L_d$  to the minimal amplitude between the two peaks. The higher the IDD, the better the perceptual frequency resolution of the model is expected to be. The IDD was measured for all LP models except

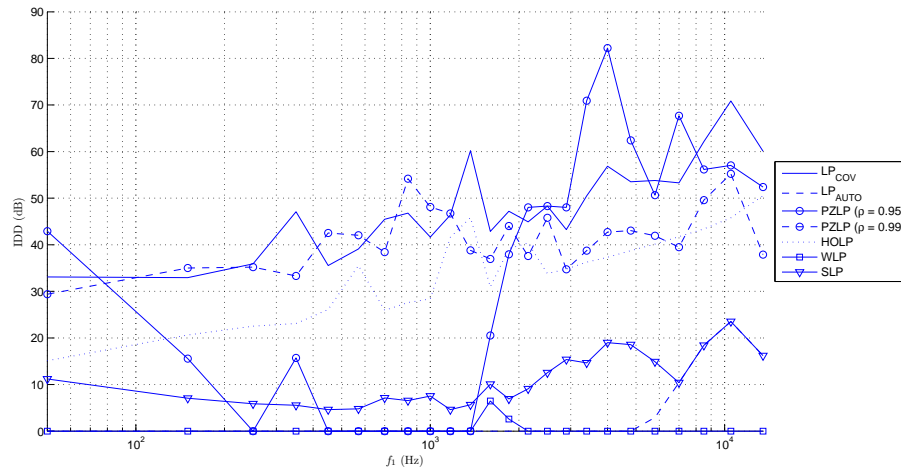


Figure 11.10: IDD results for two-tone signal with frequencies  $f_1$  and  $f_2 = f_1 + 2\text{ERB}(f_1)$ .

the PLP model, for 24 sets of two sinusoids, with  $f_1$  corresponding to the center frequency of the 24 Bark scale bands [52]. The PLP model is not appropriate for this type of signal, since the sinusoid frequencies are not harmonically related. The IDD results for the conventional LP, PZLP, WLP and SLP models with order  $2P = 2N = 4$  and for the HOLP model with order  $2P = L/2 = 1024$  are shown in Fig. 11.10. The low-order all-pole models perform poorly, except for the conventional LP model with the covariance estimation method, which has a very high IDD even in the low-frequency region. For true audio signals, however, the  $\text{LP}_{\text{COV}}$  model will perform worse in terms of perceptual frequency resolution since the estimated model parameters can strongly differ for noise-free and noisy sinusoidal signals, see Figs. 11.2(a) and 11.3(a). The HOLP model IDD exhibits a similar trend as the  $\text{LP}_{\text{COV}}$  model IDD, as it slightly increases with increasing frequency, remaining on average 14 dB below the  $\text{LP}_{\text{COV}}$  model IDD curve. The PZLP model can be seen to produce high IDD values at low and high frequencies, but performs poorly in the mid-frequency range (250 to 1370 Hz), which is exactly the frequency range of interest in audio applications. Of course, the IDD performance of an LP model is strongly related to the bandwidth of the spectral peaks that it can produce. As a consequence, the PZLP model IDD performance can be improved by increasing the pole radius (e.g.,  $\rho = 0.99$ , see Fig. 11.10), which is equivalent to reducing the smallest achievable bandwidth [54], however, when dealing with true audio signals a lower value of the pole radius is expected to be more appropriate for taking into account the damping of the tonal components.

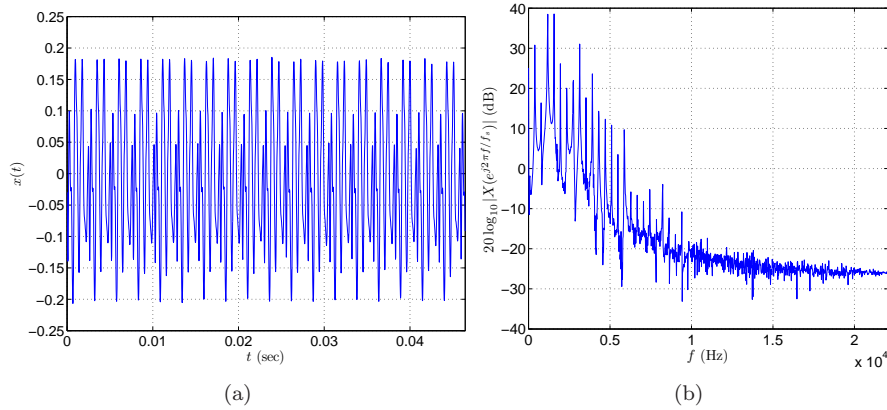


Figure 11.11: Monophonic audio signal: (a) Time-domain waveform, (b) magnitude spectrum.

### 11.5.2 Monophonic Audio Signal

A length- $L$  monophonic audio fragment was extracted from a Bb clarinet sound recording in the McGill University Master Samples collection [55]. The fragment, which corresponds to the samples 70001 to 72048 of the G4 note recording, is shown in Fig. 11.11(a), along with its magnitude spectrum in Fig. 11.11(b). The fundamental frequency corresponds to  $f_0 = 387.6$  Hz, and the number of relevant harmonics is chosen to be  $N = 15$ . A conventional LP model of order  $2P = 30$ , calculated using the autocorrelation method, produces a PEF as illustrated in Figs. 11.12(a) and 11.12(d), which is again a compromise between cancelling the tonal components and keeping the residual spectrum relatively flat. A better resolution is obtained using a PZLP model with  $2P = 30$ ,  $\rho_l = 0.95$  and  $\nu_l = 1$ ,  $l = 1, \dots, P$ , as shown in Figs. 11.12(b) and 11.12(e), and using a HOLP model with  $2P = 1024$ , see Figs. 11.12(c) and 11.12(f). A fractional 3-tap PLP model was calculated using the method proposed in [47], with the algorithm parameters given in Ex. 11.5, resulting in the PEF shown in Figs. 11.12(g) and 11.12(j), in which the spectral shaping capability of the 3-tap PLP model is clearly exploited. A WLP model with  $2P = 30$  and  $\lambda = \lambda_{\text{Bark}}(44.1)$  produces an unwarped PEF as shown in Figs. 11.12(h) and 11.12(k). Finally, the SLP model with  $2P = 30$ , for which the optimal downsampling factor from (11.72) was rounded to  $\Gamma = 4$ , has a PEF after upsampling which is given in Figs. 11.12(i) and 11.12(l).

The residual SFM values obtained with the different LP models were calculated for 2048-sample fragments taken from the sustain part of the Bb clarinet recordings in [55] with varying pitch, ranging from D3 to D6 (corresponding to  $f_0 = 146.8$  Hz to 1174.7 Hz), and are shown in Fig. 11.13(a). The original signal fragments have an average SFM value of -31 dB. The residual SFM curves



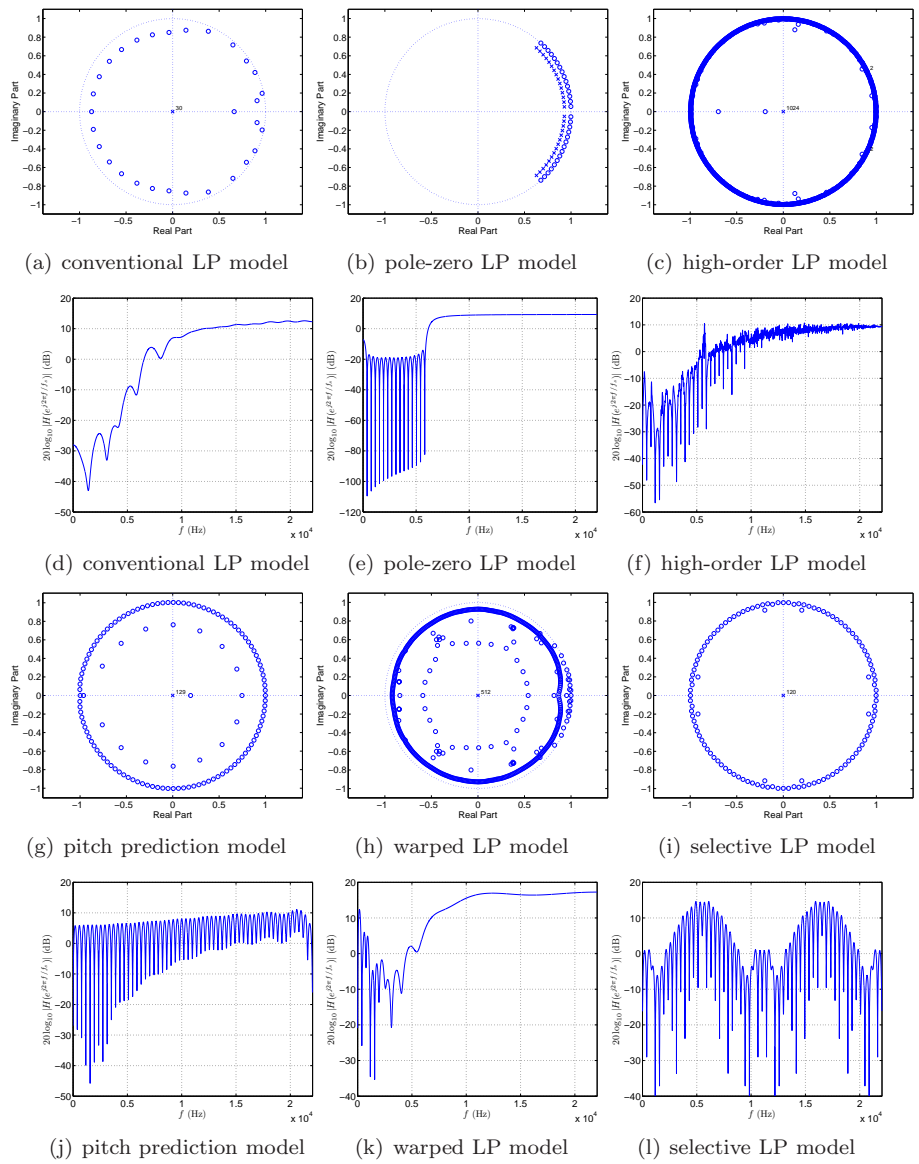
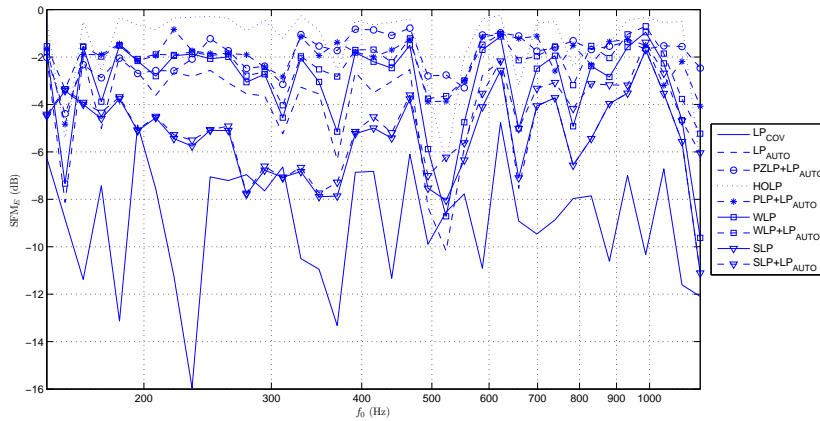


Figure 11.12: Monophonic audio signal: PEF pole-zero plots (first and third row) and PEF magnitude responses (second and fourth row) for conventional and alternative LP models.

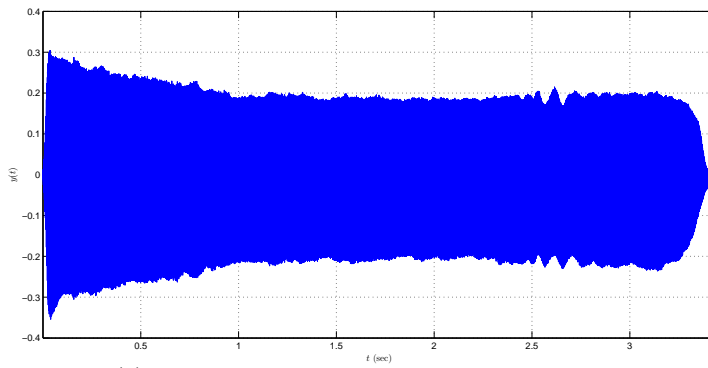
for the PZLP and PLP models are not shown, as they are (partially) outside the displayed SFM range, with an average residual SFM of -12 and -19 dB, respectively. Fig. 11.13(c) contains the residual SFM results when the analysis window time offset is varied in steps of 2048 samples from the onset till the end of the Bb clarinet G4 note in [55], which is plotted in Fig. 11.13(b). Again, the PZLP and PLP curves are omitted, with an average residual SFM of -10 and -19 dB, respectively, while the original signal fragments have an average SFM of -29 dB. From Fig. 11.13(a), we can observe that the residual SFM does not exhibit a notable trend with varying fundamental frequency for any of the LP models, which is somewhat contradictory with the results obtained for synthetic signals (see Fig. 11.9(c)). This can be explained by suggesting that the residual SFM value for true audio signals is primarily determined by the (low-power) harmonics which are modeled as noise components instead of tonal components. This undermodeling effect is generally independent of the fundamental frequency, but rather depends on which musical instrument is considered. Figs. 11.13(b) and 11.13(c) show that the LP model performance is comparable in the decay, sustain, and release part of the note, but somewhat worse in the attack part. This is mainly due to the fact that the attack part exhibits much less stationarity than the other signal parts. In both experiments, the HOLP model and the PZLP and PLP models cascaded with a conventional LP model, provide the best residual SFM results, which is consistent with the results obtained for synthetic signals (see Figs. 11.9(c) and 11.9(d)). The WLP model, potentially cascaded with a conventional LP model, performs somewhat worse yet still outperforms the  $LP_{\text{AUTO}}$  model, while the SLP and  $LP_{\text{COV}}$  models yield significantly poorer results.

### 11.5.3 Polyphonic Audio Signal

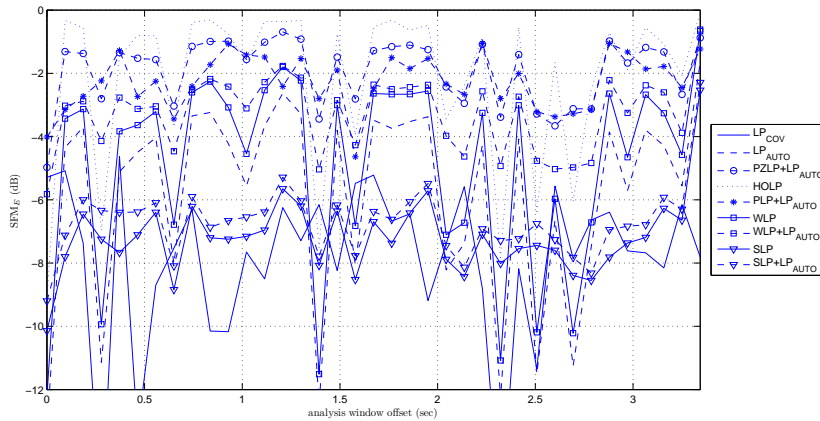
From the concert hall Steinway recordings in [55], a polyphonic audio signal was generated by adding four monophonic piano sounds. The samples 2001 to 4048 of the C4, E4, G4, and C5 note recordings were added to obtain a length- $L$  C major chord, plotted in Figs. 11.14(a) and 11.14(b). The four fundamental frequencies are  $f_{0,n} = \{258.4, 323, 387.6, 516.8\}$  Hz, and each of the monophonic components has 7 relevant harmonics, i.e.,  $M_n = 7$ ,  $n = 1, \dots, 4$ . The PEF obtained with a conventional LP model of order  $2P = 2 \sum_{n=1}^4 M_n = 54$  is shown in Figs. 11.15(a) and 11.15(d). It can be seen that the PEF has only one low-frequency notch and an overall high-pass shape. The PZLP model with  $2P = 54$ ,  $\rho_l = 0.95$  and  $\nu_l = 1$ ,  $l = 1, \dots, P$ , produces exactly as many PEF notches as there are non-overlapping tonal components, as can be seen in Figs. 11.15(b) and 11.15(e). The same holds true for the HOLP model with  $2P = 1024$ , of which the PEF is shown in Figs. 11.15(c) and 11.15(f). The PLP model does not seem to be suited for predicting polyphonic signals since the tonal components do not obey an integer harmonic relation. An alternative PLP approach could exist in cascading as many PLP models as there



(a) Residual SFM (dB) vs. fundamental frequency  $f_0$  (Hz)



(b) Time-domain waveform of analyzed Bb clarinet G4 note



(c) Residual SFM (dB) vs. analysis window time offset (s)

Figure 11.13: Residual SFM curves for a true monophonic audio signal with variable fundamental frequency and analysis window time offset.

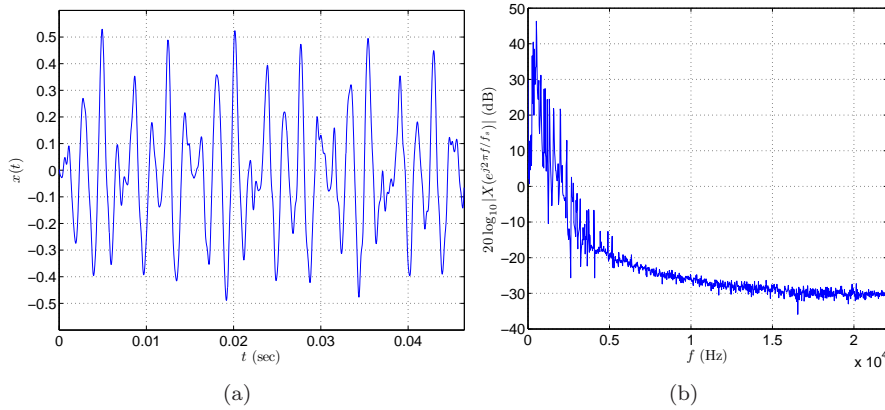


Figure 11.14: Polyphonic audio signal: (a) Time-domain waveform, (b) magnitude spectrum.

are different fundamental frequencies in the polyphonic signal, but this does not yield good results. Another alternative PLP approach may be based on the fractional harmonic relations which exist between the fundamental frequencies in a musical chord, e.g., for a major chord (consisting of dominant, third, fifth, and octave) it can be verified that  $f_{0,2} = (5/4)f_{0,1}$ ,  $f_{0,3} = (3/2)f_{0,1}$ , and  $f_{0,4} = 2f_{0,1}$ . As a consequence, a fractional PLP model with pitch lag  $K = 4(T_{0,1}/T_s)$  samples would produce PEF notches at all the tonal components in the polyphonic signal. However, allowing such large pitch lags deteriorates the performance of the algorithm for calculating the PLP model parameters, since the allowable pitch lag search space  $[K_{\min}, K_{\max}]$  becomes very large, rendering the algorithm slower and less reliable. Moreover, the large number of spurious notches in the PEF frequency response leads to an extremely non-smooth residual spectrum. As an example, a fractional pseudo-3-tap PLP model [47], assuming knowledge of the pitch lag  $K = 4(T_{0,1}/T_s) = 682.6625$  samples, was constructed by setting  $a_{K-1} = a_{K+1} = -0.05$  and  $a_K = -0.9$ . The resulting PEF when  $2I = 32$  and  $D = 8$  is shown in Figs. 11.15(g) and 11.15(j). Finally, the WLP and SLP models were applied to the polyphonic signal, both with  $2P = 54$ , a warping parameter  $\lambda = \lambda_{\text{Bark}}(44.1)$  resulting in the unwarped PEF in Figs. 11.15(h) and 11.15(k), and a downsampling factor  $\Gamma = 6$  (rounded from the optimal value in (11.75)) resulting in the upsampled PEF shown in Figs. 11.15(i) and 11.15(l).

Two similar experiments as in the monophonic case were performed, for calculating the residual SFM values after prediction of a polyphonic audio signal with varying pitch and analysis window time offset. Fig. 11.16(a) shows the residual SFM results for LP of a 4-note major chord (consisting of dominant, third, fifth, and octave) created from the concert hall Steinway recordings in

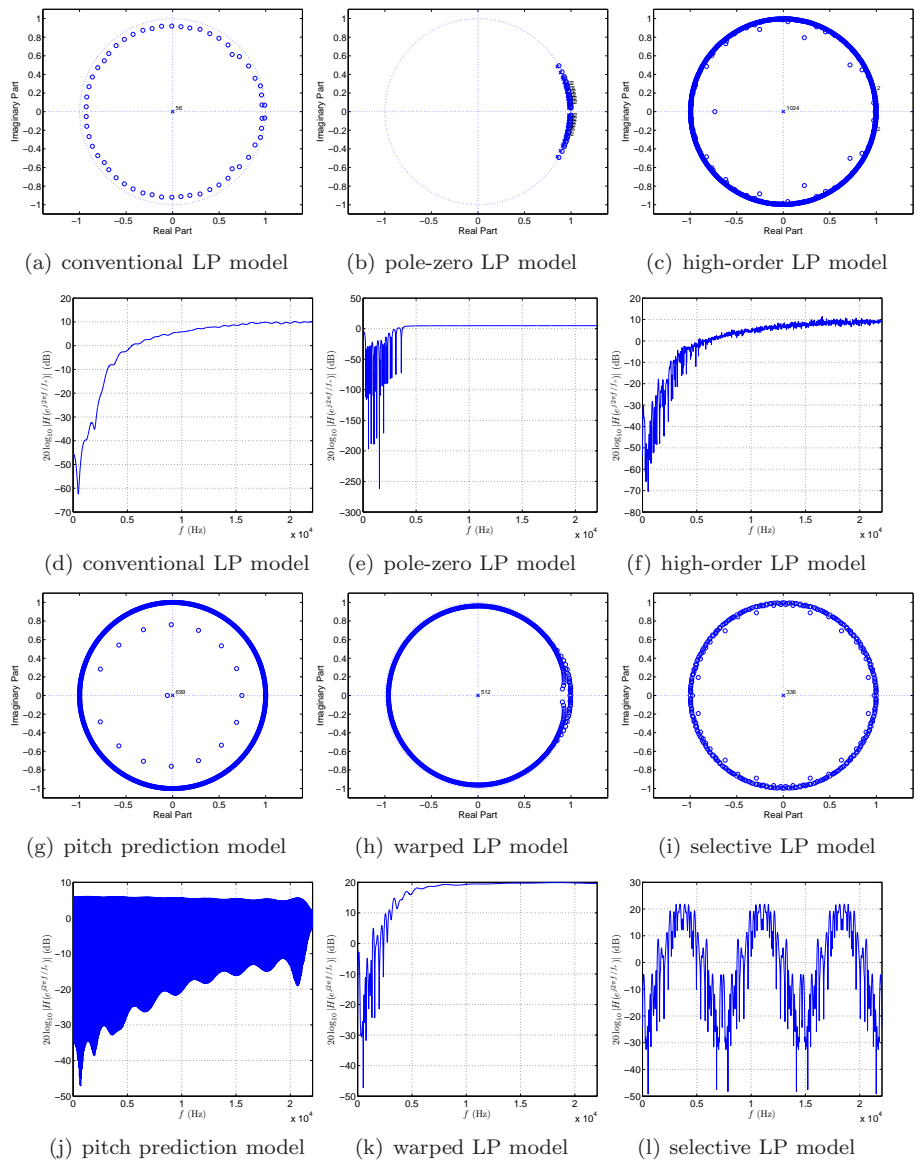
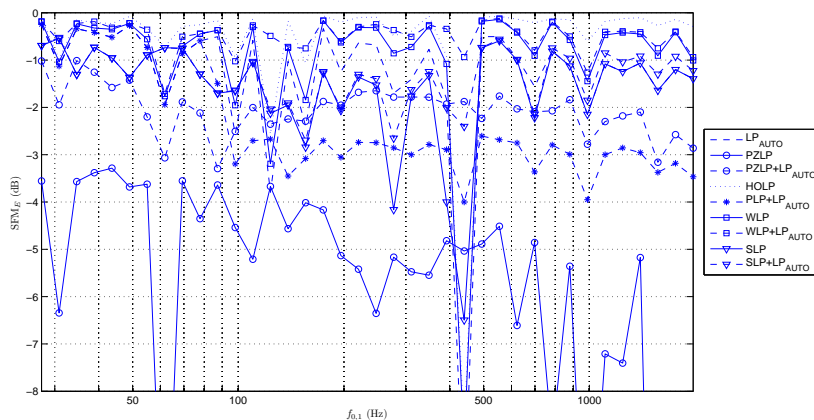


Figure 11.15: Polyphonic audio signal: PEF pole-zero plots (first and third row) and PEF magnitude responses (second and fourth row) for conventional and alternative LP models.

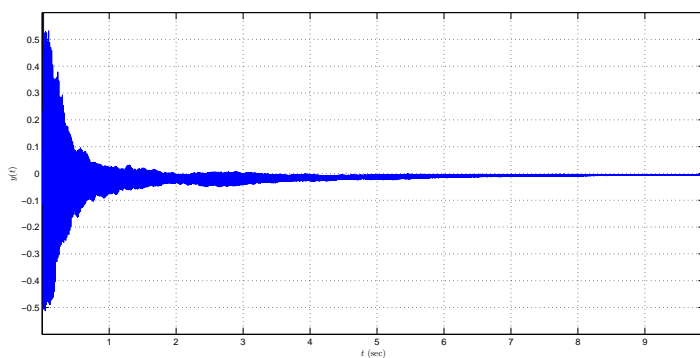
[55], in which the dominant varies from A0 to C7 (corresponding to  $f_0 = 27.5$  Hz to 2093 Hz) and the analysis window is in the release part of the chord. The  $LP_{COV}$  and PLP curves are not shown, since they are partially below the displayed residual SFM range, having a residual SFM value of -11 and -30 dB, respectively. The original polyphonic signals have an average SFM of -32 dB. At very low pitched chords, the  $LP_{AUTO}$ , HOLP, WLP, SLP models and the PZLP and PLP models cascaded with a conventional LP model, are quite competitive, however, towards higher pitch values, the HOLP and WLP models outperform the other models. The superior performance of the WLP model as compared to the other low-order models should not be a surprise. As noted in Section 11.4.4, the tonal components in a polyphonic signal are approximately distributed according to the Bark scale and are hence mapped to a nearly uniform frequency distribution after frequency warping. The  $LP_{AUTO}$  and SLP models still perform reasonably well for high pitched chords, while the cascaded PZLP and PLP models perform worse. It appears that the approach of decomposing the polyphonic signal into a number of harmonic signals (which is what the PZLP and PLP models attempt to do) is not beneficial in terms of residual spectral flatness. In Fig. 11.16(b) the 4-note major chord with dominant C4 is plotted, for which the residual SFM results of LP with a variable analysis window time offset are shown in Fig. 11.16(c). During the attack part of the chord (analysis window offset = 0 s), all LP models perform poorly. In the next 5 positions of the analysis window, which correspond to the decay and sustain parts, the residual SFM performance is the best. Again, the HOLP and WLP models yield better results than the  $LP_{AUTO}$  and SLP models, which in turn outperform the PZLP and PLP models, cascaded with a conventional LP model. In the release part of the chord (analysis window offset = ca. 0.6 s to 9.8 s), the residual SFM performance is highly fluctuating for all models, and particularly, the cascaded PZLP model residual SFM curve exhibits a decreasing trend towards the end of the chord, due to the decreasing SNR. The original C major chord has an average SFM of -37 dB, and the  $LP_{COV}$  and PLP models, resulting in an average residual SFM of -12 and -28 dB, respectively, are not shown in the graph.

## 11.6 Conclusion

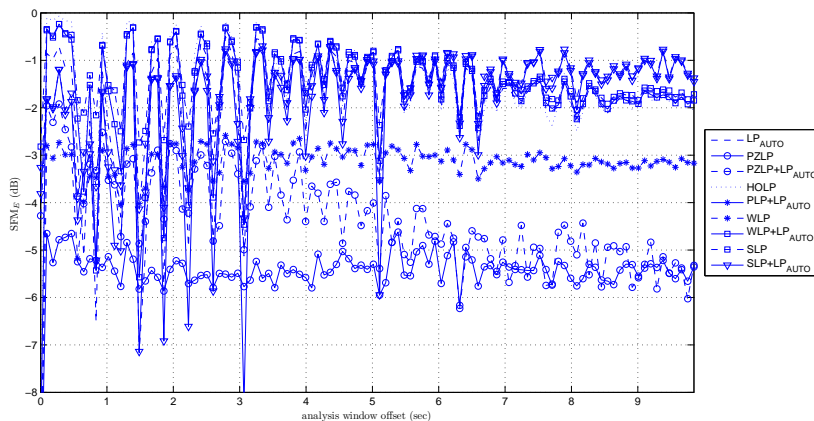
In this paper, we have analyzed the performance of the conventional LP model when applied to tonal audio signals, and illustrated how the quality of this model depends on the distribution of the signal's tonal components in the Nyquist interval. It was shown that the conventional LP model, with a model order equal to two times the number of tonal components, and calculated by minimizing an LS criterion, produces a PEF that features a trade-off between cancelling the tonal components and keeping the residual spectrum as flat as possible. This trade-off occurs since the tonal components in an audio signal,



(a) Residual SFM (dB) vs. lower fundamental frequency  $f_{0,1}$  (Hz)



(b) Time-domain waveform of analyzed C major piano chord



(c) Residual SFM (dB) vs. analysis window time offset (s)

Figure 11.16: Residual SFM curves for a true polyphonic audio signal with variable fundamental frequency and analysis window time offset.

sampled at  $f_s = 44.1$  kHz, are typically located in the lower half of the Nyquist interval.

Five existing alternative LP models were described, applied to tonal audio signals, and interpreted in terms of relieving the trade-off inherent in the conventional LP model. The first three alternative LP approaches solve the frequency distribution problem by considering a model different from the low-order all-pole model, namely a (constrained) pole-zero (PZLP) model, a high-order all-pole (HOLP) model, or a pitch prediction (PLP) model. Two other alternative approaches aim at improving the low-order all-pole model's performance, by first transforming the input signal and hence altering the distribution of its tonal components. If an all-pass bilinear transform is used, we end up with the warped all-pole (WLP) model, whereas a linear frequency transform leads to the selective all-pole (SLP) model.

Extensive simulation results were reported with the aim of assessing the performance of the conventional and alternative LP models. Summarizing, we can state that a high-order all-pole model appears to be better suited to the audio LP problem than a conventional, low-order all-pole model. However, the HOLP model, which typically has half as many model parameters as the number of samples in the analysis window, is impractically complex in many applications. It could hence be expected that the PZLP model is a good alternative, since it can approximate the HOLP PEF impulse response with fewer parameters. This seems to be true only for monophonic audio signals, and even in this case, estimating the model parameters without prior knowledge on the fundamental frequency range is not a trivial task. Another good alternative to the HOLP model in the case of monophonic signals is the PLP model, especially when cascaded with a conventional LP model, as is common use in speech analysis. Finally, for polyphonic audio LP, the WLP model performance comes very close to the optimal HOLP model performance, however, the WLP model performs poorly in terms of perceptual frequency resolution, unless its model order is chosen to be an order of magnitude larger than the number of tonal components in the observed signal [12].

## Bibliography

- [1] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, no. 4, pp. 561–578, Apr. 1975.
- [2] R. P. Ramachandran and P. Kabal, "Pitch prediction filters in speech coding," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-37, no. 4, pp. 467–478, Apr. 1989.
- [3] K. Brandenburg and G. Stoll, "ISO/MPEG-1 audio: A generic standard



- for coding of high-quality digital audio,” *J. Audio Eng. Soc.*, vol. 42, no. 10, pp. 780–792, Oct. 1994.
- [4] ISO/IEC, “IS 14496-4:2004/Amd 13:2007: Parametric coding for high quality audio conformance,” International Organization for Standardization, Geneva, Switzerland., Tech. Rep., Jan 2007.
- [5] R. J. McAulay and T. F. Quatieri, “Speech analysis/synthesis based on a sinusoidal representation,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 4, pp. 744–754, Aug. 1986.
- [6] A. Härmä, U. K. Laine, and M. Karjalainen, “Warped linear prediction in audio coding,” in *Proc. IEEE Nordic Signal Proc. Symp. (NORSIG '96)*, Espoo, Finland, 1996, pp. 447–450.
- [7] N. Iwakami and T. Moriya, “Transform-domain weighted interleave vector quantization,” in *Preprints AES 101st Convention*, Los Angeles, CA, USA, Nov. 1996, AES Preprint 4377.
- [8] B. Bessette, R. Salami, C. Laflamme, and R. Lefebvre, “A wideband speech and audio codec at 16/24/32 kbit/s using hybrid ACELP/TCX techniques,” in *Proc. 1999 IEEE Workshop Speech Coding*, Porvoo, Finland, Jun. 1999, pp. 7–9.
- [9] A. Härmä and U. K. Laine, “Warped low delay CELP for wideband audio coding,” in *Proc. AES 17th Int. Conf. High-Quality Audio Coding*, Florence, Italy, Sep. 1999, pp. 207–215.
- [10] Y. Rongshan and K. C. Chung, “High quality audio coding using a novel hybrid WLP-subband coding algorithm,” in *Proc. 5th Int. Symp. Signal Process. Appl. (ISSPA '99)*, vol. 1, Brisbane, Australia, Aug. 1999, pp. 483–486.
- [11] B. Edler, C. Faller, and G. Schuller, “Perceptual audio coding using a time-varying linear pre- and post-filter,” in *Preprints AES 109th Convention*, Los Angeles, CA, USA, Sep. 2000, AES Preprint 5274.
- [12] A. Härmä and U. K. Laine, “A comparison of warped and conventional linear predictive coding,” *IEEE Trans. Speech Audio Process.*, vol. 9, no. 5, pp. 579–588, Jul. 2001.
- [13] M. Deriche and D. Ning, “A novel audio coding scheme using warped linear prediction model and the discrete wavelet transform,” *IEEE Trans. Audio Speech Lang. Process.*, vol. 14, no. 6, pp. 2039–2048, Nov. 2006.
- [14] A. Biswas and A. C. den Brinker, “Perceptually biased linear prediction,” *J. Audio Eng. Soc.*, vol. 54, no. 12, pp. 1179–1188, Dec. 2006.

- [15] Y. Nakatoh and H. Matsumoto, "A low-bit-rate audio codec using mel-scaled linear predictive analysis," *Acoust. Sci. & Tech.*, vol. 28, no. 3, pp. 147–152, May 2007, special issue on "Applied Systems".
- [16] H. W. Strube, "Linear prediction on a warped frequency scale," *J. Acoust. Soc. Amer.*, vol. 68, no. 4, pp. 1071–1076, Oct. 1980.
- [17] T. van Waterschoot, G. Rombouts, P. Verhoeve, and M. Moonen, "Double-talk-robust prediction error identification algorithms for acoustic echo cancellation," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 846–858, Mar. 2007.
- [18] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, "Acoustic feedback suppression for long acoustic paths using a nonstationary source model," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.
- [19] T. van Waterschoot and M. Moonen, "Adaptive feedback cancellation for audio signals using a warped all-pole near-end signal model," in *Proc. 2008 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '08)*, Las Vegas, NV, USA, Apr. 2008, pp. 269–272.
- [20] —, "Adaptive feedback cancellation for audio applications," *Signal Processing*, conditionally accepted for publication, Dec. 2008, ESAT-SISTA Technical Report TR 07-30, Katholieke Universiteit Leuven, Belgium. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/07-30.html>
- [21] M. Pagano, "Estimation of autoregressive signals plus white noise," *Ann. Statist.*, vol. 2, no. 1, pp. 97–108, 1974.
- [22] S. M. Kay, "The effects of noise on the autoregressive spectral estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 5, pp. 478–485, Oct. 1979.
- [23] Y. T. Chan, J. M. M. Lavoie, and J. B. Plant, "A parameter estimation approach to estimation of frequencies of sinusoids," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 2, pp. 214–219, Apr. 1981.
- [24] D. V. B. Rao and S.-Y. Kung, "Adaptive notch filter for the retrieval of sinusoids in noise," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, no. 4, pp. 791–802, Aug. 1984.
- [25] W. J. Fitzgerald and R. Geere, "Class of constrained ARMA models for line enhancement using real-time QR implementation," *IEE Electronics Lett.*, vol. 27, no. 24, pp. 2230–2231, Nov. 1991.
- [26] V. F. Pisarenko, "The retrieval of harmonics from a covariance function," *Geophys. J. Roy. Astron. Soc.*, vol. 33, pp. 347–366, 1973.

- [27] L. B. Jackson, D. W. Tufts, F. K. Soong, and R. M. Rao, "Frequency estimation by linear prediction," in *Proc. 1978 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '78)*, Tulsa, OK, USA, Apr. 1978, pp. 352–356.
- [28] S. H. Nam, "Stabilizing discrete spectral modeling of audio signals," *IEEE Signal Process. Lett.*, vol. 9, no. 9, pp. 292–294, Sep. 2002.
- [29] A. V. Oppenheim, D. H. Johnson, and K. Steiglitz, "Computation of spectra with unequal resolution using the fast Fourier transform," *Proc. IEEE*, vol. 59, no. 2, pp. 299–301, Feb. 1971.
- [30] J. O. Smith and J. S. Abel, "Bark and ERB bilinear transforms," *IEEE Trans. Speech Audio Process.*, vol. 7, no. 6, pp. 697–708, Nov. 1999.
- [31] A. Nehorai, "A minimal parameter adaptive notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 4, pp. 983–996, Aug. 1985.
- [32] A. Nehorai and B. Porat, "Adaptive comb filtering for harmonic signal enhancement," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 5, pp. 1124–1138, Oct. 1986.
- [33] T. S. Ng, "Some aspects of an adaptive digital notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 2, pp. 158–161, Feb. 1987.
- [34] J. M. Travassos-Romano and M. Bellanger, "Fast least squares adaptive notch filtering," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-36, no. 9, pp. 1536–1540, Sep. 1988.
- [35] G. Li, "A stable and efficient adaptive notch filter for direct frequency estimation," *IEEE Trans. Signal Process.*, vol. 45, no. 8, pp. 2001–2009, Aug. 1997.
- [36] T. van Waterschoot and M. Moonen, "Constrained pole-zero linear prediction: an efficient and near-optimal method for multi-tone frequency estimation," in *Proc. 16th European Signal Process. Conf. (EUSIPCO '08)*, Lausanne, Switzerland, Aug. 2008.
- [37] T. van Waterschoot, M. Diehl, and M. Moonen, "Constrained pole-zero linear prediction: Optimization of cascaded biquadratic notch filters for multi-tone and multi-pitch estimation," to be submitted for publication, Mar. 2009, ESAT-SISTA Technical Report TR 07-115, Katholieke Universiteit Leuven, Belgium, Feb. 2008.
- [38] J. Ojanperä, M. Väänänen, and L. Yin, "Long term predictor for transform domain perceptual audio coding," in *Preprints AES 107th Convention*, New York, NY, USA, Sep. 1999, AES Preprint 5036.

- [39] J. Herre and B. Grill, "Overview of MPEG-4 audio and its applications in mobile communications," in *Proc. 5th Int. Conf. Signal Process. (WCCC-ICSP '00)*, vol. 1, Beijing, China, Aug. 2000, pp. 11–20.
- [40] G. E. Kopec, A. V. Oppenheim, and J. M. Tribolet, "Speech analysis by homomorphic prediction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-25, no. 1, pp. 40–49, Feb. 1977.
- [41] K. Steiglitz, "On the simultaneous estimation of poles and zeros in speech analysis," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-25, no. 3, pp. 229–234, Jun. 1977.
- [42] L. Mitiche, B. Derras, and A. B. H. Adamou-Mitiche, "Efficient low-order auto regressive moving average (ARMA) models for speech signals," *Acoust. Res. Lett. Online*, vol. 5, no. 2, pp. 75–81, Apr. 2004.
- [43] J. D. Markel and A. H. Gray, Jr., *Linear prediction of speech*. New York: Springer-Verlag, 1976.
- [44] T. van Waterschoot and M. Moonen, "Linear prediction of audio signals," in *Proc. Interspeech 2007*, Antwerp, Belgium, Aug. 2007, pp. 518–521.
- [45] R. Kumaresan, "On the zeros of the linear prediction-error filter for deterministic signals," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, no. 1, pp. 217–220, Feb. 1983.
- [46] T. J. Ulrych and T. N. Bishop, "Maximum entropy spectral analysis and autoregressive decomposition," *Rev. Geophys. Space Phys.*, vol. 13, pp. 183–200, Feb. 1978.
- [47] Y. Qian, G. Chahine, and P. Kabal, "Pseudo-multi-tap pitch filters in a low bit-rate CELP speech coder," *Speech Commun.*, vol. 14, no. 4, pp. 339–358, Sep. 1994.
- [48] P. Kroon and B. S. Atal, "Pitch predictors with high temporal resolution," in *Proc. 1990 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '90)*, vol. 2, Albuquerque, NM, USA, Apr. 1990, pp. 661–664.
- [49] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine, "Splitting the unit delay: Tools for fractional delay filter design," *IEEE Signal Process. Mag.*, vol. 13, no. 1, pp. 30–60, Jan. 1996.
- [50] A. Härmä, M. Karjalainen, L. Savioja, V. Välimäki, U. K. Laine, and J. Huopaniemi, "Frequency-warped signal processing for audio applications," *J. Audio Eng. Soc.*, vol. 48, no. 11, pp. 1011–1031, Nov. 2000.
- [51] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1996.

- [52] E. Zwicker and H. Fastl, *Psychoacoustics, Facts and Models*. Berlin: Springer-Verlag, 1990.
- [53] B. C. J. Moore and B. R. Glasberg, "A revision of Zwicker's loudness model," *Acta Acoustica*, vol. 82, pp. 335–345, 1996.
- [54] T. van Waterschoot and M. Moonen, "A pole-zero placement technique for designing second-order IIR parametric equalizer filters," *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 8, pp. 2561–2565, Nov. 2007.
- [55] F. Opolko and J. Wapnick, *McGill University Master Samples*. Montreal, QC, Canada: McGill University, 2006, DVD edition.



## Chapter 12

# AFC for Audio Applications

Adaptive feedback cancellation for audio applications

Toon van Waterschoot and Marc Moonen

Conditionally accepted for publication in *Signal Processing*,  
Mar. 2009.

### Contributions of first author

- literature study
- co-analysis of acoustic feedback problem
- co-development of AFC concept for audio signals using cascaded near-end signal models
- co-development of prediction error identification algorithm structure
- co-analysis of identifiability conditions
- co-development of PEM-based AFC algorithms with cascaded near-end signal models
- co-analysis of computational complexity and modeling approximations
- co-design of evaluation experiments
- software implementation and computer simulations
- co-interpretation of simulation results
- co-formulation of conclusion
- text redaction and editing



## Abstract

Acoustic feedback occurs in many audio applications involving musical sound signals. However, research efforts in acoustic feedback control have mainly been focused on speech applications. Since sound quality is of prime importance in audio applications, a proactive approach to acoustic feedback control is preferred to avoid ringing, howling, and excessive reverberation. Adaptive feedback cancellation (AFC) using a prediction-error-method (PEM)-based approach is a promising proactive solution, but existing algorithms are again designed for speech applications only. We propose to replace the all-pole near-end speech signal model in the PEM-based approach with a cascade of two near-end signal models: a tonal components model and a noise components model. We derive the identifiability conditions for joint identification of the acoustic feedback path and the cascaded near-end signal models. Depending on the model structure that is used for the near-end tonal components, five different PEM-based AFC algorithms are considered. By applying some relevant model approximations, the computational overhead of the proposed algorithms compared to the normalized least mean squares (NLMS) algorithm can be reduced to 25 % of the NLMS complexity. Simulation results for both room acoustic and hearing aid scenarios indicate a significant performance improvement in terms of the misadjustment and the maximum stable gain increase.

## 12.1 Introduction

Acoustic feedback is a physical phenomenon arising in several speech and audio applications, which may severely degrade sound quality and may even cause damage to human hearing and to loudspeaker components. When a sound signal is picked up by a microphone and then amplified and played back in the same acoustic environment, a closed signal loop is created, which may give rise to system instability. The existence of an acoustic feedback path limits a sound system's performance in two ways. First of all, there is an upper limit to the amount of amplification that can be applied if the system is required to remain stable, which is referred to as the maximum stable gain (MSG). Second, the sound quality is affected by occasional howling when the MSG is exceeded, or, even when the system is operating below the MSG, by ringing and excessive reverberation.

Many solutions to the acoustic feedback problem have been proposed, see [1] for an overview. Apart from manual feedback control [2], the two most promising solutions are notch-filter-based howling suppression (NHS) and adaptive feedback cancellation (AFC). Notch-filter-based solutions aim at detecting narrowband ringing or howling sounds in the microphone signal spectrum, and subsequently inserting suitably designed notch filters in the closed signal loop

to stabilize the system and remove the narrowband interferences. The ringing or howling detection is typically based on an energy comparison of the entire microphone signal spectrum and its narrowband components [3]-[5], whereas the notch filter design is usually performed using a biquadratic filter design method [6]-[8]. The main advantages of the notch-filter-based approach are its reasonable computational cost and its robustness when operating in an unstable sound system. A major shortcoming, however, is the reactive nature of NHS, i.e., feedback suppression is only achieved after the system has become (nearly) unstable. As a result, signal distortion can never be avoided. Moreover, the MSG when using an NHS is on average only 5 dB higher than the MSG without feedback control [1], which is too small in many applications. Two approaches have been suggested to obtain a so-called proactive feedback detection in NHS systems, i.e., for detecting frequencies where system instability could potentially occur even before ringing effects actually become noticeable. A first approach is based on detecting hidden resonances by cumulative harmonic analysis (CHA) [9], but has not yet been incorporated in an NHS system for performance evaluation. A second approach features a combined NHS-AFC system [10],[11], that achieves proactive notch filtering at the cost of an increased computational complexity.

Adaptive feedback cancellation (AFC) is a more recent approach to acoustic feedback control, based on the concept of acoustic echo cancellation (AEC). If the loudspeaker signal is fed to an adaptive filter, having the microphone signal as its desired output signal, then the feedback component in the microphone signal may be predicted by the adaptive filter. However, in contrast to the AEC situation, in AFC the loudspeaker signal is correlated with the near-end signal, which causes standard adaptive filtering algorithms to converge to a biased solution [12],[13]. A variety of decorrelation techniques has been suggested for bias removal, which can be divided into two categories [14], depending on whether the decorrelation is performed in the closed signal loop, or in the adaptive filtering circuit. Decorrelation in the closed signal loop is achieved by cascading the amplifier with a delay [12],[15],[16], with an all-pass filter [17], or with a nonlinear signal operation such as a frequency shifting [16],[18],[19] or a half-wave rectification [20],[21], or by adding a noise signal to the loudspeaker signal, in a continuous [21]-[23] or non-continuous [24],[25] way. Decorrelation in the adaptive filtering circuit can be obtained by cascading the adaptive filter with a delay [25]-[27], by performing an indirect closed-loop identification [28], or by prefiltering the adaptive filter's input and desired signal with an inverse model of the near-end signal, which is either fixed [13],[29], or adaptively estimated [11],[30]-[32]. In contrast to the NHS approach, an AFC algorithm performs proactive feedback control, i.e., if the acoustic feedback path characteristics can be effectively tracked by the adaptive filter, then the sound system is guaranteed to remain stable, and moreover ringing and reverberation effects are also avoided. In hearing aid applications, increases of up to 20 dB in MSG have been reported [15],[18],[31]. The main drawback of the AFC approach is

its high computational cost, especially in room acoustic applications where the adaptive filter order is typically very large.

Research efforts in acoustic feedback control so far have mainly dealt with speech applications. In this paper, we explicitly focus on feedback control in audio applications involving musical signals, e.g., public address (PA) systems in concert venues, or hearing aids (HA) operating in a musical environment. When dealing with audio instead of speech applications, two major issues should be taken into account. First of all, whereas in speech applications intelligibility is of prime interest, for audio applications sound quality becomes much more important. Second, audio signals typically exhibit a much higher degree of tonality than speech signals, whereas many feedback control techniques are not designed to work with tonal signals. In fact, none of the above-mentioned NHS or AFC approaches is capable of meeting these two requirements. From a sound quality point of view, the NHS approach is inappropriate since howling, ringing, and excessive reverberation cannot be avoided. Moreover, in the NHS feedback detection, discriminating between undesired feedback oscillations and desired tonal components in the microphone signal spectrum is a non-trivial task [3],[5]. Existing AFC techniques are generally also not appropriate for audio applications. Decorrelation in the closed signal loop will either lead to unacceptable signal distortion (in the case of frequency shifting, half-wave rectification, and noise injection), or will not be capable of providing sufficient decorrelation for tonal near-end signals (in the case of delay, all-pass filtering, and psychoacoustically masked noise injection [33]). When performing decorrelation in the adaptive filtering circuit, cascading the adaptive filter with a delay will also be insufficient for tonal signals, while indirect closed-loop identification requires the injection of a reference signal, which is again undesirable in terms of signal quality. AFC techniques that include a prefiltering of the adaptive filter's input and desired signal with an inverse model of the near-end signal have been designed particularly for near-end speech signals, where the near-end signal model is a low-order all-pole speech signal model. Finally, in a closed-loop scenario a tonal near-end signal generates a tonal loudspeaker signal, so that the adaptive filter input signal is also tonal, which may dramatically decrease its convergence speed [34, Ch. 9].

The aim of this paper is to develop a modification to existing prediction-error-method (PEM)-based AFC approaches [11],[31],[32], such that these become capable of dealing with tonal audio signals. The PEM-based AFC algorithms are based on the PEM for system identification [35, Ch. 3],[36, Ch. 7]. Decorrelation is performed by prefiltering the adaptive filter's input and desired output signal with a time-varying inverse model of the near-end signal, which is estimated by linear prediction (LP) of the feedback-compensated signal. The PEM-AF algorithm in [31] was derived for hearing aid applications, featuring a recursive LP of the feedback-compensated signal, and involving some common model approximations which are only relevant for short acoustic feedback

paths. In [32], the PEM-AFROW algorithm was proposed for room acoustic applications, featuring a batch (frame-based) LP, and inheriting its name from the fact that no model approximations are introduced such that the pre-filtering operation only performs row operations in the loudspeaker signal data matrix. In [37], the PEM-AF and PEM-AFROW algorithms were shown to be special cases of a more general recursive prediction error (RPE) identification algorithm. A common feature of the PEM-AF, PEM-AFROW, and RPE algorithms is the low-order all-pole structure that is used for modeling the near-end signal, which is indeed appropriate for speech signals. However, this conventional LP model is usually not well suited for tonal audio signals, which can be modeled more efficiently as a sum of sinusoids plus noise. It is well known that a signal consisting of sinusoids in noise admit a pole-zero rather than an all-pole representation [38],[39]. As a consequence, the existing PEM-based AFC algorithms can be applied to audio signals only if the all-pole near-end signal model order is chosen very large. This would however lead to a dramatic increase of the computational requirements for the PEM-AF, PEM-AFROW, and RPE algorithms and to a violation of the PEM-AF stationarity assumptions in time-varying acoustic environments. In [40], we have investigated several alternative LP models for audio signals: pole-zero models, pitch prediction all-pole models, frequency-warped all-pole models, and downsampled all-pole models. Some of these alternative models appear to be capable of generating a “whiter”, i.e., less correlated LP residual than the conventional low-order all-pole model, especially when cascaded with a conventional LP model. This observation is exploited in the current paper to derive a set of new AFC algorithms that can also handle tonal near-end signals. The proposed algorithms feature a cascade of two near-end signal models, a first one for predicting the tonal components, and a second one for predicting the “noise-like” components in the near-end signal. The noise components model is chosen to be a conventional low-order all-pole model, while the tonal components model can be any of the alternative LP models described in [40]. An additional advantage of the proposed algorithms is that, by prefiltering the adaptive filter’s input signal with the cascaded inverse near-end signal models, the tonal components in the input signal are also (partially) removed, and hence the adaptive filter’s convergence is further improved.

This paper is organized as follows. In Section 12.2, the acoustic feedback problem is described in a discrete-time signal processing context, and the AFC concept is explained. In Section 12.3, we introduce a prediction error minimization criterion that features a cascade of two near-end signal models, and outline the proposed AFC algorithm. Also, an overview is given of the possible model structures for the near-end tonal components. In Section 12.4, we rederive the identifiability conditions given in [31] for the PEM-AF algorithm, for the case of cascaded near-end signal models, resulting in the requirement of inserting processing delays at appropriate positions either in the closed signal loop or in the adaptive filtering circuit. Then in Section 12.5, algorithmic details of the

PEM-based AFC approach with cascaded near-end signal models are given for different near-end tonal components model structures. Section 12.6 deals with computational complexity and contains an overview of the model approximations that can be applied for decreasing the complexity. In Section 12.7, we illustrate the performance of the proposed algorithms by means of simulation results in both PA and HA scenarios. Finally, Section 12.8 concludes the paper.

## 12.2 Adaptive Feedback Cancellation

### 12.2.1 Problem Description

The acoustic feedback problem is depicted in Fig. 12.1(a) for a setup with one microphone and one loudspeaker. In this setup, we refer to the source signal  $v(t)$  as the near-end signal, and to the loudspeaker signal  $u(t)$  as the far-end signal (adopting terminology from acoustic echo cancellation). The acoustic feedback path  $F\{\cdot\}$  is defined as a function that maps the far-end signal  $u(t)$  to the feedback signal  $x(t)$ , and is typically assumed to be linear, (slowly) time-varying, and of finite order  $n_F$ , i.e.,

$$F(q, t) = f_0(t) + f_1(t)q^{-1} + \dots + f_{n_F}(t)q^{-n_F} \quad (12.1)$$

where  $t \in \mathbb{Z}$  denotes the discrete time variable after sampling at sampling frequency  $f_s = 1/T_s$ , and  $q$  denotes the time shift operator, i.e.,  $q^{-k}u(t) = u(t - k)$ . The electro-acoustic forward path  $G\{\cdot\}$  maps the microphone signal  $y(t) = v(t) + x(t)$  to the far-end signal  $u(t)$  and is defined as the cascade of the characteristics of the microphone, the A/D-converter, the amplifier, the D/A-converter, the loudspeaker, and any signal processing device that is inserted in the signal loop, such as an equalizer, a compressor, etc. The forward path mapping is typically nonlinear for large signal amplitudes, due to amplifier or loudspeaker saturation, or because of compression. In the closed-loop system analysis, however, it is usually assumed that the forward path mapping is linear and time-varying, i.e.,

$$G(q, t) = g_1(t)q^{-1} + \dots + g_{n_G}(t)q^{-n_G}, \quad (12.2)$$

and possibly of infinite order ( $n_G \rightarrow \infty$ ). Note that the forward path is assumed to contain (at least) one unit delay, i.e.,  $g_0(t) \equiv 0$ , to avoid an algebraic loop.

The far-end signal and the near-end signal are related by the so-called closed-loop transfer function as follows:

$$u(t) = \frac{G(q, t)}{1 - G(q, t)F(q, t)}v(t). \quad (12.3)$$

According to Nyquist's stability criterion [41], the closed-loop system becomes

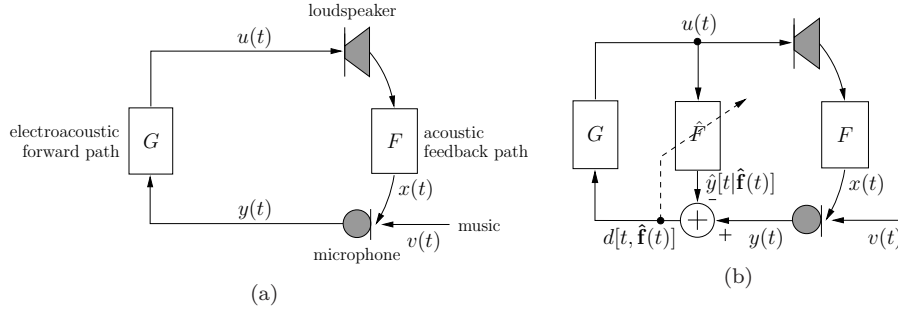


Figure 12.1: (a) Acoustic feedback problem in 1-microphone/1-loudspeaker setup, (b) Adaptive feedback cancellation concept.

unstable if there exists a radial frequency  $\omega$  for which

$$\begin{cases} |G(e^{j\omega}, t)F(e^{j\omega}, t)| \geq 1 \\ \angle G(e^{j\omega}, t)F(e^{j\omega}, t) = n2\pi, \quad n \in \mathbb{Z} \end{cases} \quad (12.4)$$

where the short-time frequency responses  $G(e^{j\omega}, t)$  and  $F(e^{j\omega}, t)$  of the forward and feedback path, respectively, are obtained using the short-time Fourier transform (STFT). Except for the phase-modulated feedback control methods (see [1] for an overview), most of the existing methods for acoustic feedback control attempt to avoid the magnitude condition in (12.4) from being met for any  $\omega \in [0, \pi]$ , disregarding the phase condition (12.5). The maximum stable gain (MSG) is defined as the electro-acoustic forward path gain value at which the point of instability of the closed-loop system is attained, and is usually determined in an experimental way, see, e.g., [25],[42]. If the amplifier's broadband gain factor  $K(t)$  is factored out from the forward path transfer function, i.e.,

$$G(q, t) = K(t)J(q, t) \quad (12.6)$$

and if  $\mathcal{P}$  denotes the set of frequencies at which the phase condition (12.5) is met, i.e.,

$$\mathcal{P} = \{\omega | \angle G(e^{j\omega}, t)F(e^{j\omega}, t) = n2\pi\} \quad (12.7)$$

then the maximum stable gain (MSG) can be formally defined as follows,

$$\text{MSG}(t) [\text{dB}] = -20 \log_{10} \left[ \max_{\omega \in \mathcal{P}} |J(e^{j\omega}, t)F(e^{j\omega}, t)| \right]. \quad (12.8)$$

### 12.2.2 Adaptive Feedback Cancellation

The AFC concept consists in placing an FIR adaptive filter  $\hat{F}(q, t)$  in parallel with the acoustic feedback path, having the far-end signal as its input and the microphone signal as its desired signal, see Fig. 12.1(b). The

feedback signal  $x(t)$  is then predicted by the adaptive filter output signal  $\hat{y}[t|\hat{\mathbf{f}}(t)] = \hat{F}(q, t)u(t)$ , which is subtracted from the microphone signal to deliver the feedback-compensated signal  $d[t, \hat{\mathbf{f}}(t)] = y(t) - \hat{y}[t|\hat{\mathbf{f}}(t)]$ , with

$$\hat{\mathbf{f}}(t) \triangleq [\hat{f}_0(t) \quad \dots \quad \hat{f}_{n_F}(t)]^T. \quad (12.9)$$

Throughout this paper, we will assume that the acoustic feedback path model order  $n_F$  is known and that the adaptive filter order is equal to  $n_F$ . Note that the PEM-based AFC approach introduced in Section 12.3 has been shown to reduce the undermodeling bias and variance that tend to occur in the insufficient order case ( $n_{\hat{F}} < n_F$ ) [43]. The closed-loop transfer function of the system with AFC is given by

$$u(t) = \frac{G(q, t)}{1 - G(q, t)[F(q, t) - \hat{F}(q, t)]}v(t) \quad (12.10)$$

such that the MSG can now be written as follows,

$$\text{MSG}(t) = -20 \log_{10} \left[ \max_{\omega} |J(e^{j\omega}, t)[F(e^{j\omega}, t) - \hat{F}(e^{j\omega}, t)]| \right] \quad (12.11)$$

and obviously increases when the mismatch between  $\hat{F}(q, t)$  and  $F(q, t)$  decreases. It is also expected that when  $\hat{F}(q, t)$  approaches  $F(q, t)$ , the feedback-compensated signal  $d[t, \hat{\mathbf{f}}(t)]$  will approach the near-end signal  $v(t)$ , which should lead to better sound quality [1].

## 12.3 PEM-Based AFC

### 12.3.1 Data Model

The estimation of the adaptive filter coefficients in  $\hat{\mathbf{f}}(t)$  should be approached from a closed-loop system identification point of view. It is well known that if the near-end signal  $v(t)$  is a correlated sequence, such as speech or music, then standard Wiener or least-squares (LS) estimation provides a biased solution [1],[12],[13],[44]. An unbiased feedback path estimate can be obtained with the so-called direct method [44] when a model of the near-end signal is taken into account in the identification (corresponding to the “noise model” in system identification theory). The data model can then be written as

$$y(t) = F(q, t)u(t) + \overbrace{H(q, t)e(t)}^{v(t)} \quad (12.12)$$

with  $e(t)$  an uncorrelated sequence such as Gaussian white noise or a Dirac impulse. However, because of the nonstationarity of speech and music signals, the near-end signal model  $H(q, t)$  is time-varying and so should be estimated

concurrently with the acoustic feedback path  $F(q, t)$ . This is possible by applying a prediction error system identification method [35, Ch. 3], [36, Ch. 7], as shown in [11], [31], [32], [37]. Here, the near-end signal model is assumed to be an all-pole model, which is a relevant assumption for speech applications.

If the near-end signal is a tonal audio signal, then an all-pole model is usually not appropriate, but instead a cascade of two linear models may be used for the near-end signal [40]. The data model can then be rewritten as

$$y(t) = F(q, t)u(t) + H_1(q, t)H_2(q, t)e(t). \quad (12.13)$$

In the near-end signal model cascade,  $H_1(q, t)$  is a model for the tonal components, while  $H_2(q, t)$  is a model for the “noise-like” components. The noise components model is again chosen to be an all-pole model, i.e.,

$$H_2(q, t) = \frac{1}{C(q, t)} = \frac{1}{1 + c_1(t)q^{-1} + \dots + c_{n_c}(t)q^{-n_c}} \quad (12.14)$$

which corresponds to the near-end speech model used in the estimation algorithms in [11], [31], [32], [37]. The tonal components model can be any of the LP models described in [40]: an all-pole (LP) model, a pole-zero (PZLP) model, a pitch prediction (PLP) model, a frequency-warped all-pole (WLP) model, or a selective all-pole (SLP) model. Table 12.1 lists these five models, together with the corresponding prediction error filter (PEF) transfer functions and parameter vectors. Note that the parameter vectors  $\alpha(t)$ , which contain the tonal components model parameters that have to be estimated in the PEM-based AFC algorithm, are not equivalent to the PEF impulse response vectors, which will be denoted as  $\mathbf{a}(t)$ . Also, the PEF order  $n_A$  is not necessarily equal to the number of elements in the parameter vector  $\alpha(t)$ , which will be denoted by  $n_\alpha$ . In the PZLP model, the numerator and denominator order are equal, and the poles and zeros are constrained to lie on the same radial lines in the  $z$ -plane, more specifically at angles  $\theta_i(t)$ ,  $i = 1, \dots, n_A/2$ . The fractional pitch lag  $K - l/D$  (with  $K \in \mathbb{Z}$  and  $l = 0, \dots, D - 1$ ) in the fractional 3-tap PLP model can be implemented by using a fractional interpolation filter  $I(q, l/D)$ . The WLP and SLP models both have an all-pole structure in which the unit delay element has been transformed: in the SLP model the transformation consists in a downsampling operation (anti-aliasing filtering followed by decimation) with a factor  $\Gamma$ , while in the WLP the unit delay  $q^{-1}$  is replaced by a bilinear all-pass filter

$$D(q, \lambda) = \frac{q^{-1} - \lambda}{1 - \lambda q^{-1}} \quad (12.15)$$

with warping parameter  $\lambda \in (-1, 1)$ . The WLP model moreover features an initial whitening filter

$$D_0^{-1}(q, \lambda) = \frac{1 - \lambda q^{-1}}{\sqrt{1 - \lambda^2}} \quad (12.16)$$

to increase the residual’s spectral flatness [45]. More details on these models and their properties can be found in [40].



Table 12.1: Overview of near-end tonal components models

Model	PEF transfer function	Parameter vector
LP	$A(q, t) = 1 + \sum_{i=1}^{n_A} a_i(t)q^{-i}$	$\boldsymbol{\alpha}(t) = [a_1(t), \dots, a_{n_A}(t)]^T$
PZLP	$\frac{A(q, t)}{B(q, t)} = \prod_{i=1}^{n_A/2} \frac{1 - 2\nu_i \cos \theta_i q^{-1} + \nu_i^2 q^{-2}}{1 - 2\rho_i \cos \theta_i q^{-1} + \rho_i^2 q^{-2}}$	$\boldsymbol{\alpha}(t) = [\theta_1(t), \dots, \theta_{n_A/2}(t)]^T$
PLP	$\begin{aligned} A(q, t) &= 1 - \sum_{i=-1}^1 \alpha_i(t)q^{-K-(l/D)-i} \\ &= 1 - \sum_{i=-1}^1 \alpha_i(t)I(q, l/D)q^{-K-i} \end{aligned}$	$\boldsymbol{\alpha}(t) = [K, l, \alpha_{-1}(t), \alpha_0(t), \alpha_1(t)]^T$
WLP	$A(q, t) = D_0^{-1}(q, \lambda)[1 + \sum_{i=1}^{n_A} \alpha_i(t)D^i(q, \lambda)]$	$\boldsymbol{\alpha}(t) = [\alpha_1(t), \dots, \alpha_{n_A}(t)]^T$
SLP	$A(q, t) = 1 + \sum_{i=1}^{n_A} \alpha_i(t)q^{-i\Gamma}$	$\boldsymbol{\alpha}(t) = [\alpha_1(t), \dots, \alpha_{n_A}(t)]^T$

### 12.3.2 Prediction Error Identification Algorithm

Using the data model in (12.13), the prediction error identification approach can be outlined as follows. The best one-step ahead predictor for  $y(t)$  can be calculated, following [35, Ch. 3], as

$$\hat{y}[t|\boldsymbol{\xi}(t)] = [1 - H_2^{-1}(q, t)H_1^{-1}(q, t)]y(t) + H_2^{-1}(q, t)H_1^{-1}(q, t)F(q, t)u(t) \quad (12.17)$$

with the parameter vector  $\boldsymbol{\xi}(t)$  defined as

$$\boldsymbol{\xi}(t) \triangleq [\mathbf{f}^T(t) \quad \boldsymbol{\gamma}^T(t) \quad \boldsymbol{\alpha}^T(t)]^T \quad (12.18)$$

and

$$\mathbf{f}(t) \triangleq [f_0(t) \quad \dots \quad f_{n_F}(t)]^T \quad (12.19)$$

$$\boldsymbol{\gamma}(t) \triangleq [c_1(t) \quad \dots \quad c_{n_C}(t)]^T \quad (12.20)$$

and with  $\boldsymbol{\alpha}(t)$  defined in Table 12.1. The prediction error, defined as

$$\varepsilon[t, \boldsymbol{\xi}(t)] \triangleq y(t) - \hat{y}[t|\boldsymbol{\xi}(t)] \quad (12.21)$$

can hence be calculated as

$$\varepsilon[t, \boldsymbol{\xi}(t)] = H_2^{-1}(q, t)H_1^{-1}(q, t)[y(t) - F(q, t)u(t)]. \quad (12.22)$$

The parameter vector  $\boldsymbol{\xi}(t)$  can be estimated by minimizing the sum of squared prediction errors,

$$\min_{\boldsymbol{\xi}(t)} \frac{1}{2N} \sum_{k=1}^t \zeta^{-1}(k, t) \varepsilon^2[k, \boldsymbol{\xi}(t)] \quad (12.23)$$

with  $\zeta^{-1}(k, t)$  a weighting factor for discounting old data and compensating for power variations in the near-end excitation signal  $e(t)$ , and  $N$  denoting the effective window length after data weighting.

In AFC, it is considered advantageous to decouple the identification of  $F(q, t)$ ,  $H_1(q, t)$ , and  $H_2(q, t)$ . This allows for using data windows of different length [31] and applying different estimation methods [32] for the identification of the acoustic feedback path and the near-end signal models. It has been shown that this approach results in an estimate  $\hat{\boldsymbol{\xi}}(t)$  that corresponds to a local minimum of the criterion in (12.23), but not necessarily to the global minimum [32],[37]. It was found in [37] that a smaller near-end signal model order increases the probability of finding the global solution, which is yet another motivation for using a cascade of two low-order near-end signal models rather than a single high-order all-pole model. The identification of  $F(q, t)$ ,  $H_1(q, t)$ , and  $H_2(q, t)$  can be decoupled by performing the minimization of (12.23) in three stages:

1. Estimation of  $H_1(q, t)$ : using (12.14), we can rewrite (12.22) as

$$H_1(q, t)\varepsilon[t, \boldsymbol{\xi}(t)] = C(q, t)[y(t) - F(q, t)u(t)] \quad (12.24)$$

$$\triangleq w[t, \boldsymbol{\gamma}(t), \mathbf{f}(t)]. \quad (12.25)$$

The near-end tonal components model  $H_1(q, t)$  can then be estimated using an appropriate LP method for predicting  $w[t, \boldsymbol{\gamma}(t), \mathbf{f}(t)]$ , and replacing the parameter vectors  $\boldsymbol{\gamma}(t)$  and  $\mathbf{f}(t)$  by recently obtained estimates, see Section 12.5 for a detailed treatment. Note that the prefiltering operation with  $C(q, t)$  in (12.24) is expected to whiten the near-end noise components in the feedback-compensated signal  $y(t) - F(q, t)u(t)$ , which facilitates the estimation of the near-end tonal components model  $H_1(q, t)$ .

2. Estimation of  $H_2(q, t)$ : rewriting (12.22) with (12.14) as

$$C^{-1}(q, t)\varepsilon[t, \boldsymbol{\xi}(t)] = H_1^{-1}(q, t)[y(t) - F(q, t)u(t)] \quad (12.26)$$

$$\triangleq r[t, \boldsymbol{\alpha}(t), \mathbf{f}(t)]. \quad (12.27)$$

reveals that the near-end noise components model  $H_2(q, t) = C^{-1}(q, t)$  can be estimated by LP of  $r[t, \boldsymbol{\alpha}(t), \mathbf{f}(t)]$ , with  $\boldsymbol{\alpha}(t)$  and  $\mathbf{f}(t)$  replaced by recent estimates, see Section 12.5. Since the near-end tonal components in the feedback-compensated signal  $y(t) - F(q, t)u(t)$  are cancelled by the prefiltering with  $H_1^{-1}(q, t)$ , these do not disturb the near-end noise components model estimation.

3. Estimation of  $F(q, t)$ : if we define the following prefiltered far-end and microphone signals,

$$\tilde{u}[t, \boldsymbol{\alpha}(t), \boldsymbol{\gamma}(t)] \triangleq C(q, t)H_1^{-1}(q, t)u(t) \quad (12.28)$$

$$\tilde{y}[t, \boldsymbol{\alpha}(t), \boldsymbol{\gamma}(t)] \triangleq C(q, t)H_1^{-1}(q, t)y(t) \quad (12.29)$$

then the minimization of the sum of squared prediction errors in (12.23) w.r.t.  $\boldsymbol{\xi}(t)$  can be rewritten as a standard LS minimization w.r.t.  $\mathbf{f}(t)$ ,

$$\min_{\mathbf{f}(t)} \frac{1}{2N} \sum_{k=1}^t \zeta^{-1}(k, t) \{ \tilde{y}[t, \boldsymbol{\alpha}(t), \boldsymbol{\gamma}(t)] - F(q, t)\tilde{u}[t, \boldsymbol{\alpha}(t), \boldsymbol{\gamma}(t)] \}^2 \quad (12.30)$$

in which the parameter vectors  $\boldsymbol{\alpha}(t)$  and  $\boldsymbol{\gamma}(t)$  may be replaced by recently obtained estimates, see Section 12.5. In the LS problem defined in (12.30), the near-end signal component in the microphone signal has been whitened by prefiltering with  $C(q, t)H_1^{-1}(q, t)$  such that an unbiased estimate of the acoustic feedback path can be obtained. A beneficial side effect of this approach is that the tonal components in the far-end signal, whose frequencies can be assumed to be equal to the near-end tonal component frequencies since the electro-acoustic forward path is modeled as a linear system  $G(q, t)$ , are (partially) cancelled by prefiltering with  $H_1^{-1}(q, t)$ , which improves the conditioning of the LS problem in (12.30).

## 12.4 Identifiability Conditions

Before presenting the details of the PEM-based AFC algorithm with cascaded near-end signal models, it is necessary to derive the conditions under which the models  $F(q, t)$ ,  $H_1(q, t)$ , and  $H_2(q, t)$  are jointly identifiable from the LS criterion in (12.22)-(12.23). This derivation differs depending on which tonal components model is used.

### 12.4.1 LP, PLP, and SLP Near-End Tonal Components Models

When the inverse near-end tonal components model  $H_1^{-1}(q, t)$  is chosen to have a finite-order all-zero parametrization (i.e., referring to Table 12.1, in case of the LP, PLP, and SLP model structures), the inverse cascaded near-end signal models  $H_1^{-1}(q, t) = A(q, t)$  and  $H_2^{-1}(q, t) = C(q, t)$  form a single all-zero model  $D(q, t) \triangleq C(q, t)A(q, t)$  of order  $n_D = n_A + n_C$ , and the identifiability conditions derived in [31] can be applied. In this case,  $F(q, t)$  and  $D(q, t)$  are jointly identifiable if all of the following conditions are satisfied [31]:

1. the near-end signal admits an autoregressive (AR) representation of order  $n_D$  or less,
2. processing delays of  $d_1$  and  $d_2$  samples are inserted in the electro-acoustic forward path  $G(q, t)$  and in the adaptive filtering circuit, respectively, with  $d_1 + d_2 \geq n_D + 1$ ,
3. the acoustic feedback path has an initial delay of at least  $d_2 T_s$  s due to the time needed for the sound to travel in a direct path from the loudspeaker to the microphone.

Note that these conditions do not guarantee the unique identification of  $C(q, t)$  and  $A(q, t)$ , since all the zeros of these polynomials are identified together in the cascade model  $D(q, t)$ . However, this should not be a problem since the identification of  $C(q, t)$  and  $A(q, t)$  is not of primary interest, but merely serves as an auxiliary procedure for consistently identifying  $F(q, t)$ .

### 12.4.2 WLP Near-End Tonal Components Model

The WLP PEF can either be implemented as an IIR filter, or as a warped FIR filter [45]. In the latter case, the derivation of the identifiability conditions is similar to the derivation in [31], resulting in the requirements that:

1. the near-end signal admits a mixed conventional/frequency-warped AR representation of orders  $n_C$  and  $n_A$  or less, respectively,
2. processing delays  $d_1$  and  $d_2$  are inserted with  $d_1 + d_2 \geq n_A + n_C + 1$ ,
3. the acoustic feedback path has an initial delay of at least  $d_2 T_s$  s.



Using (12.10) and  $y(t) = F(q, t)u(t) + v(t)$ , we can rewrite (12.32) as

$$\varepsilon[t, \boldsymbol{\xi}(t)] = \frac{D(q, t) + G(q, t)[L(q, t) + \hat{F}(q, t)D(q, t)]}{1 - G(q, t)[F(q, t) - \hat{F}(q, t)]}v(t) - q^{-1}\bar{B}(q, t)\varepsilon[t, \boldsymbol{\xi}(t)]. \quad (12.33)$$

Let us again assume that the forward path and the adaptive filtering circuit contain processing delays of  $d_1$  and  $d_2$  samples respectively, and that the acoustic feedback path has an initial delay of at least  $d_2T_s$  s. Under these assumptions, the following equalities hold:

$$G(q, t) = q^{-d_1}\bar{G}(q, t) \text{ with } \bar{G}(q, t) \triangleq g_{d_1} + g_{d_1+1}q^{-1} + \dots + g_{n_G}q^{-n_G+d_1} \quad (12.34)$$

$$F(q, t) = q^{-d_2}\bar{F}(q, t) \text{ with } \bar{F}(q, t) \triangleq f_{d_2} + f_{d_2+1}q^{-1} + \dots + f_{n_F}q^{-n_F+d_2} \quad (12.35)$$

$$\hat{F}(q, t) = q^{-d_2}\hat{\hat{F}}(q, t) \text{ with } \hat{\hat{F}}(q, t) \triangleq \hat{f}_{d_2} + \hat{f}_{d_2+1}q^{-1} + \dots + \hat{f}_{n_F}q^{-n_F+d_2} \quad (12.36)$$

$$L(q, t) = q^{-d_2}\bar{L}(q, t) \text{ with } \bar{L}(q, t) \triangleq l_{d_2} + l_{d_2+1}q^{-1} + \dots + l_{n_L}q^{-n_L+d_2} \quad (12.37)$$

with  $n_L = n_A + n_C + n_F$ , and hence (12.33) can be rewritten as follows:

$$\begin{aligned} \varepsilon[t, \boldsymbol{\xi}(t)] &= \frac{D(q, t) + q^{-(d_1+d_2)}\bar{G}(q, t)[\bar{L}(q, t) + \hat{\hat{F}}(q, t)D(q, t)]}{1 - q^{-(d_1+d_2)}\bar{G}(q, t)[\bar{F}(q, t) - \hat{\hat{F}}(q, t)]}v(t) - q^{-1}\bar{B}(q, t)\varepsilon[t, \boldsymbol{\xi}(t)] \\ &= \{D(q, t) + q^{-(d_1+d_2)}\bar{G}(q, t)[\bar{L}(q, t) + \hat{\hat{F}}(q, t)D(q, t)]\}v(t) \\ &\quad - \{q^{-1}\bar{B}(q, t) - q^{-(d_1+d_2)}\bar{G}(q, t)[\bar{F}(q, t) - \hat{\hat{F}}(q, t)]B(q, t)\}\varepsilon[t, \boldsymbol{\xi}(t)]. \end{aligned} \quad (12.38)$$

If the near-end signal admits an ARMA representation  $D_0(q, t)/B_0(q, t)$  with the AR and MA orders less than or equal to  $n_D$  and  $n_B$ , respectively, then the solution to the LS problem (12.23) with (12.38) is equal to the desired solution if

$$d_1 + d_2 \geq \max\{n_D, n_B\} + 1 \quad (12.39)$$

$$\geq n_A + n_C + 1 \quad (12.40)$$

where the latter inequality follows from the fact that we have constrained the PZLP model denominator and numerator order to be equal, see Section 12.3.1. Indeed, it can be verified that in this case the solution to (12.23) and (12.38) corresponds to

$$\left\{ \begin{array}{l} D(q, t) = D_0(q, t) \end{array} \right. \quad (12.41)$$

$$\left\{ \begin{array}{l} \bar{G}(q, t)[\bar{L}(q, t) + \hat{\hat{F}}(q, t)D(q, t)] \equiv 0 \Leftrightarrow \bar{L}(q, t) \equiv -\hat{\hat{F}}(q, t)D(q, t) \end{array} \right. \quad (12.42)$$

$$\left\{ \begin{array}{l} B(q, t) = B_0(q, t) \end{array} \right. \quad (12.43)$$

$$\left\{ \begin{array}{l} \bar{G}(q, t)[\bar{F}(q, t) - \hat{\hat{F}}(q, t)]B(q, t) \equiv 0 \Leftrightarrow \bar{F}(q, t) \equiv \hat{\hat{F}}(q, t). \end{array} \right. \quad (12.44)$$

Note that, as was the case for the LP, PLP, and SLP models, an unavoidable ambiguity exists between the zeros of the PZLP near-end tonal components

model PEF  $A(q, t)/B(q, t)$  and the noise components model PEF  $C(q, t)$ , which are combined in the cascade model  $D(q, t)$ .

Finally, also note that an example of a signal admitting an ARMA( $n_D, n_B$ ) representation is a signal consisting of a sum of sinusoids in AR noise, i.e.,

$$v(t) = \sum_{n=1}^N \beta_n \cos(\omega_n t + \phi_n) + \frac{1}{C(q, t)} e(t). \quad (12.45)$$

As shown in [39], the linear prediction property of a sum of  $N$  sinusoidal signals leads to an ARMA( $2N, 2N$ ) representation in white noise, which can be extended to an ARMA( $2N + n_C, 2N$ ) representation in AR( $n_C$ ) noise.

## 12.5 Algorithm Details

In the existing PEM-AF [31] and RPE [37] algorithms, the near-end signal model  $H(q, t)$  is identified recursively, while the PEM-AFROW [32] algorithm features a batch near-end signal model identification. It has been found that the latter approach is more robust, since a recursive near-end signal model identification may result in numerical problems due to a scaling ambiguity that is inherent in the PEM-based approach [46]. Moreover, efficient batch estimation methods for identifying the near-end tonal components models in Table 12.1 are readily available in the literature, see [40] for an overview. For these reasons, we will only consider batch estimation of the near-end tonal and noise components models  $H_1(q, t)$  and  $H_2(q, t)$ . Moreover, we will assume that  $H_1(q, t)$  and  $H_2(q, t)$  are piecewise stationary on similar time scales, such that both models can be identified on data windows of the same size. More specifically, we will use data windows that have a length of  $M$  samples and a hop size of  $P$  samples. Moreover, the data window is positioned in time such that it contains  $P - 1$  future samples and  $M - P$  past samples. The choice of  $M$  and  $P$  are crucial for the AFC algorithm performance:  $M$  should be chosen large enough to obtain low-variance estimates of the parameters of  $H_1(q, t)$  and  $H_2(q, t)$ , but not too large such that the models themselves can be assumed stationary in the entire data window. For LP of audio signals, data windows of 40–60 ms appear to be well suited [40]. The hop size  $P$  could theoretically be chosen nearly as large as the data window length  $M$  (a minimal difference of  $M - P = n_C$  will appear to be necessary, as shown below), however, it should be taken into account that a processing delay of  $P - 1$  samples has to be inserted in the forward path  $G(q, t)$  to preserve causality in the AFC algorithm. We will typically choose  $P = M/2$ , such that successive LP data windows have a 50 % overlap. This choice implies that the forward path contains a delay corresponding to 20–30 ms. From a perceptual point of view, a forward path delay of 20–30 ms should be acceptable in PA applications since the typical

distance values between the loudspeakers and the audience introduce similar delay values. In HA applications, inserting a forward path delay introduces a time offset between the so-called “bone-conducted” sound signal and the “aid-conducted” sound signal. Delays of 20–30 ms (or higher for severely hearing-impaired subjects) were found to be acceptable in terms of speech quality [47], however, no results for audio signals have been reported.

The PEM-based AFC algorithms with cascaded near-end signal models presented here, are recursive algorithms in which each recursion consists of a sequence of nine operations:

```

for  $t$ 
  if  $j = t \bmod P = 0$ 
    1) calculation of a priori feedback-compensated signal  $d[t, \hat{\mathbf{f}}(t-1)]$ 
       for the entire LP data window
    2) calculation of prefiltered data vector  $\mathbf{w}[t, \hat{\gamma}(t-P), \hat{\mathbf{f}}(t-1)]$ 
    3) batch estimation of  $\hat{\boldsymbol{\alpha}}(t)$  using  $\mathbf{w}[t, \hat{\gamma}(t-P), \hat{\mathbf{f}}(t-1)]$ 
    4) calculation of prefiltered data vector  $\mathbf{r}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)]$ 
    5) batch estimation of  $\hat{\gamma}(t)$  using  $\mathbf{r}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)]$ 
  end if
  6) calculation of prediction error  $\varepsilon[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\gamma}(t-j), \hat{\mathbf{f}}(t-1)]$ 
     and prefiltered data vector  $\tilde{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\gamma}(t-j)]$ 
  7) recursive estimation of prediction error power  $\sigma^2(t)$ 
  8) recursive estimation of  $\hat{\mathbf{f}}(t)$  using  $\varepsilon[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\gamma}(t-j), \hat{\mathbf{f}}(t-1)]$ 
     and  $\tilde{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\gamma}(t-j)]$ 
  9) calculation of a posteriori feedback-compensated signal  $d[t, \hat{\mathbf{f}}(t)]$ 
end for

```

The prefiltering and LP estimation details are different depending on the near-end tonal components model used, and will be described for the different cases.

### 12.5.1 LP, PLP, and SLP Near-End Tonal Components Models

If the near-end tonal components model has an all-zero PEF, i.e., for the LP, PLP, and SLP models, the above nine operations can be described as shown in Table 12.2. The impulse response coefficients of the PEFs  $\hat{A}(q, t)$  and  $\hat{C}(q, t)$  are collected in the vectors  $\hat{\mathbf{a}}(t)$  and  $\hat{\mathbf{c}}(t)$ , respectively, which are different from—but related to—parameter vectors  $\hat{\boldsymbol{\alpha}}(t)$  and  $\hat{\gamma}(t)$  [see (12.20) and Table 12.1]. Note that for the calculation of  $\varepsilon[t, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t), \hat{\mathbf{f}}(t-1)]$  from  $r[k, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)]$ ,  $k \in [t - n_C, t]$  in step 6a), it is required that  $P \leq M - n_C$ .

The recursive estimation of the acoustic feedback path parameter vector  $\hat{\mathbf{f}}(t)$  in step 8 of the PEM-based AFC algorithm is carried out using an NLMS-like update equation, using the prefiltered far-end signal vector  $\tilde{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t)]$



instead of the original far-end signal vector (as would be used in a standard NLMS-based AFC algorithm). Apart from the normalization factor  $\tilde{\mathbf{u}}^T \tilde{\mathbf{u}}$ , the estimated prediction error power  $\sigma^2(t)$  and the regularization parameter  $\delta$  also appear in the denominator of the update term. Three estimates of the prediction error power [ $\sigma_A^2(t)$ ,  $\sigma_C^2(t)$ , and  $\sigma_\varepsilon^2(t)$ ] are available in the algorithm, and these are averaged to obtain the prediction error power estimate  $\sigma^2(t)$  that is used in the update equation for  $\hat{\mathbf{f}}(t)$ .

In Table 12.2, we have omitted the actual algorithms for estimating the LP, PLP, and SLP model coefficients. The estimation of LP model coefficients is a well-known problem, which is readily solved by estimating a set of autocorrelation coefficients and subsequently solving a linear system of equations, see, e.g., [48]. When estimating the autocorrelation coefficients, either the autocorrelation method or the covariance method can be used. However, the covariance method appears to be superior for predicting the near-end signal tonal components since it does not disturb the (quasi-)periodicity of the near-end signal by zero padding. The SLP model coefficients are estimated in a similar way, be it that before estimating the autocorrelation coefficients, the signal is downsampled by a factor  $\Gamma$ . The resulting SLP model coefficient vector is upsampled by the same factor  $\Gamma$ . The downsampling factor should be chosen such that the most important harmonics are maintained in the downsampled signal spectrum, which implies that this choice generally depends on the frequency range and the relative importance of the harmonics of the musical instrument that produces the near-end signal. Estimating the coefficients of the fractional 3-tap PLP model coefficients can be done by applying a two-step pitch prediction algorithm. First the pitch lag  $K$  and fractional phase  $l$  are estimated by performing an exhaustive search for the minimal fractional one-tap PLP residual power in the two-dimensional grid defined by  $K \in \{[K_{\min}, K_{\max}] \cap \mathbb{Z}\}$  and  $l \in \{[0, D-1] \cap \mathbb{Z}\}$  [49],[50]. The fractional 3-tap PLP model coefficients are then estimated by calculating the autocorrelation coefficients for lags around the previously estimated fractional pitch lag value  $K + l/D$ , and subsequently solving a linear system of equations. This system of equations can be forced to be Toeplitz or diagonal to speed up the estimation [51].

### 12.5.2 WLP Near-End Tonal Components Model

Since the WLP PEF  $A(q, t)$  has an infinite impulse response, the algorithm in Table 12.2 cannot be used when the tonal components model has the WLP model structure. It was shown in [52] that an efficient recursive AFC algorithm can be obtained in this case by performing the prefiltering operations involving  $A(q, t)$  directly in the warped domain. This is possible because an IIR WLP PEF can be implemented as a warped FIR filter [45], which has a finite number of filter states. The approach in [52] can be extended with a cascaded near-end noise components model, resulting in the algorithm shown in Table 12.3. The main difference with the algorithm in Table 12.2 is found in step 4, where the

Table 12.2: PEM-based AFC algorithm: LP, PLP, and SLP near-end tonal components models

---

**for**  $t$

**if**  $j = t \bmod P = 0$

- 1)  $d[k, \hat{\mathbf{f}}(t-1)] = y(k) - \begin{bmatrix} u(k) & \dots & u(k-n_F) \end{bmatrix} \hat{\mathbf{f}}(t-1), \quad k \in [t+P-M-\max(n_A, n_C), t+P-1]$
- 2)  $\mathbf{w}[t, \hat{\boldsymbol{\gamma}}(t-P), \hat{\mathbf{f}}(t-1)] = \begin{bmatrix} d[t+P-M, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-M-n_C, \hat{\mathbf{f}}(t-1)] \\ \vdots & \ddots & \vdots \\ d[t+P-1, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-1-n_C, \hat{\mathbf{f}}(t-1)] \end{bmatrix} \hat{\mathbf{c}}(t-P)$
- 3)  $\{\hat{\boldsymbol{\alpha}}(t), \sigma_A^2(t)\} = \text{lp/plp/slp}\{\mathbf{w}[t, \hat{\boldsymbol{\gamma}}(t-P), \hat{\mathbf{f}}(t-1)]\}$
- 4)  $\mathbf{r}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] = \begin{bmatrix} d[t+P-M, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-M-n_A, \hat{\mathbf{f}}(t-1)] \\ \vdots & \ddots & \vdots \\ d[t+P-1, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-1-n_A, \hat{\mathbf{f}}(t-1)] \end{bmatrix} \hat{\mathbf{a}}(t)$
- 5)  $\{\hat{\boldsymbol{\gamma}}(t), \sigma_C^2(t)\} = \text{lp}\{\mathbf{r}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)]\}$
- 6a)  $\begin{cases} \varepsilon[t, \hat{\boldsymbol{\alpha}}(t), \hat{\boldsymbol{\gamma}}(t), \hat{\mathbf{f}}(t-1)] = \begin{bmatrix} r[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] & \dots & r[t-n_C, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] \end{bmatrix} \hat{\mathbf{c}}(t) \\ \check{u}[k, \hat{\boldsymbol{\alpha}}(t)] = \begin{bmatrix} u(k) & \dots & u(k-n_A) \end{bmatrix} \hat{\mathbf{a}}(t), \quad k \in [t-n_F-n_C, t+P-1] \\ \check{y}[k, \hat{\boldsymbol{\alpha}}(t)] = \begin{bmatrix} y(k) & \dots & y(k-n_A) \end{bmatrix} \hat{\mathbf{a}}(t), \quad k \in [t-n_C+1, t+P-1] \\ \check{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\boldsymbol{\gamma}}(t)] = \begin{bmatrix} \check{u}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\boldsymbol{\gamma}}(t)] \\ \vdots \\ \check{u}[t-n_F, \hat{\boldsymbol{\alpha}}(t), \hat{\boldsymbol{\gamma}}(t)] \end{bmatrix} = \begin{bmatrix} \check{u}[t, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{u}[t-n_C, \hat{\boldsymbol{\alpha}}(t)] \\ \vdots & \ddots & \vdots \\ \check{u}[t-n_F, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{u}[t-n_F-n_C, \hat{\boldsymbol{\alpha}}(t)] \end{bmatrix} \hat{\mathbf{c}}(t) \end{cases}$

**else**

- 6b)  $\begin{cases} \check{u}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] = \begin{bmatrix} \check{u}[t, \hat{\boldsymbol{\alpha}}(t-j)] & \dots & \check{u}[t-n_C, \hat{\boldsymbol{\alpha}}(t-j)] \end{bmatrix} \hat{\mathbf{c}}(t-j) \\ \check{y}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] = \begin{bmatrix} \check{y}[t, \hat{\boldsymbol{\alpha}}(t-j)] & \dots & \check{y}[t-n_C, \hat{\boldsymbol{\alpha}}(t-j)] \end{bmatrix} \hat{\mathbf{c}}(t-j) \\ \check{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] = \begin{bmatrix} \check{u}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] & \dots & \check{u}[t-n_F, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] \end{bmatrix}^T \\ \varepsilon[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j), \hat{\mathbf{f}}(t-1)] = \check{y}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] - \check{\mathbf{u}}^T[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] \hat{\mathbf{f}}(t-1) \end{cases}$

**end if**

- 7)  $\begin{cases} \sigma_\varepsilon^2(t) = \lambda_\varepsilon \sigma_\varepsilon^2(t-1) + (1-\lambda_\varepsilon) \varepsilon^2[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j), \hat{\mathbf{f}}(t-1)] \\ \sigma^2(t) = [\sigma_A^2(t-j) + \sigma_C^2(t-j) + \sigma_\varepsilon^2(t)]/3 \end{cases}$
- 8)  $\hat{\mathbf{f}}(t) = \hat{\mathbf{f}}(t-1) + \mu \frac{\check{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] \varepsilon[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j), \hat{\mathbf{f}}(t-1)]}{\check{\mathbf{u}}^T[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] \check{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t-j), \hat{\boldsymbol{\gamma}}(t-j)] + \sigma^2(t) + \delta}$
- 9)  $d[t, \hat{\mathbf{f}}(t)] = y(t) - \begin{bmatrix} u(t) & \dots & u(t-n_F) \end{bmatrix} \hat{\mathbf{f}}(t)$

**end for**

---

signals  $\check{u}[k, \hat{\alpha}(t)]$  and  $\check{y}[k, \hat{\alpha}(t)]$  are computed as an intermediate step before calculating the prefiltered data vectors  $\mathbf{r}[t, \hat{\alpha}(t), \hat{\mathbf{f}}(t-1)]$  and  $\tilde{\mathbf{u}}[t, \hat{\alpha}(t), \hat{\gamma}(t)]$ . The far-end and microphone signals  $u(k)$  and  $y(k)$  are transformed to the two-dimensional frequency-warped signals  $\bar{u}(k, \kappa)$  and  $\bar{y}(k, \kappa)$ , before being filtered by the warped PEF  $\hat{A}(q, t)$  to obtain  $\check{u}[k, \hat{\alpha}(t)]$  and  $\check{y}[k, \hat{\alpha}(t)]$ . By organizing the calculations in this way, none of the filtering operations involve an infinite number of filter states. An efficient algorithm for estimating the WLP model coefficients in  $\alpha(t)$  can be found in [45]: first the warped autocorrelation coefficients are calculated, which are then fed to a Levinson-Durbin recursion to find the model coefficient estimates.

### 12.5.3 PZLP Near-End Tonal Components Model

The PZLP PEF  $\hat{A}(q, t)/\hat{B}(q, t)$  also has an infinite impulse response but, in contrast with the WLP PEF, an exact recursive computation is not possible in the PZLP case. Therefore, in all prefiltering operations involving the PZLP PEF, the initial denominator filter states are approximated by signal values that are prefiltered with an earlier estimate of the PZLP PEF denominator  $B(q, t)$ . The resulting algorithm is shown in Table 12.4. The PZLP approximations appear in step 4 and step 6a of Table 12.4, more specifically in the data matrices multiplying the PZLP PEF denominator coefficient vector  $\hat{\mathbf{b}}(t) \triangleq [\hat{b}_1(t), \dots, \hat{b}_{n_A}(t)]$  (which has been truncated such that the leading coefficient  $\hat{b}_0(t) \equiv 1$  is lacking). The signal values in the upper triangular part (above and including the diagonal) of these matrices are prefiltered using the previously estimated PZLP PEF  $\hat{A}(q, t-P)/\hat{B}(q, t-P)$  instead of using the current estimate. We should also remark that the matrix equations involving prefiltering with the PZLP PEF in step 4) and step 6a) of Table 12.4 should be evaluated in a row-by-row fashion, since some of the output signal values needed in the right-hand side of the equation are only available in the precedings rows on the left hand side.

The PZLP model coefficients can be estimated using the so-called constrained pole-zero linear prediction (CPZLP) method [53],[54]. This method is similar to the adaptive notch filtering (ANF) method [55]-[57], however, it operates iteratively on a batch of data instead of recursively updating the estimates of the PZLP model parameters. The main advantage of the batch estimation lies in the fact that the gradient estimates are recalculated using the entire data window in each iteration, which makes the algorithm less sensitive to the choice of the initial conditions as compared to the ANF algorithms [54].

Table 12.3: PEM-based AFC algorithm: WLP near-end tonal components model

---

**for**  $t$

**if**  $j = t \bmod P = 0$

1)  $d[k, \hat{\mathbf{f}}(t-1)] = y(k) - \begin{bmatrix} u(k) & \dots & u(k-n_F) \end{bmatrix} \hat{\mathbf{f}}(t-1), \quad k \in [t+P-M-n_C, t+P-1]$

2)  $\mathbf{w}[t, \hat{\gamma}(t-P), \hat{\mathbf{f}}(t-1)] = \begin{bmatrix} d[t+P-M, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-M-n_C, \hat{\mathbf{f}}(t-1)] \\ \vdots & \ddots & \vdots \\ d[t+P-1, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-1-n_C, \hat{\mathbf{f}}(t-1)] \end{bmatrix} \hat{\mathbf{c}}(t-P)$

3)  $\{\hat{\boldsymbol{\alpha}}(t), \sigma_A^2(t)\} = \text{wlp}\{\mathbf{w}[t, \hat{\gamma}(t-P), \hat{\mathbf{f}}(t-1)]\}$

4)  $\left\{ \begin{array}{l} \bar{u}(k, \kappa) = D_0^{-1}(q, \lambda) D^\kappa(q, \lambda) u(k), \quad k \in [t, t+P-1], \quad \kappa \in [0, n_\alpha] \\ \bar{y}(k, \kappa) = D_0^{-1}(q, \lambda) D^\kappa(q, \lambda) y(k), \quad k \in [t, t+P-1], \quad \kappa \in [0, n_\alpha] \\ \check{u}[k, \hat{\boldsymbol{\alpha}}(t)] = \bar{u}(k, 0) + \begin{bmatrix} \bar{u}(k, 1) & \dots & \bar{u}(k, n_\alpha) \end{bmatrix} \hat{\boldsymbol{\alpha}}(t), \quad k \in [t+P-M-n_F, t+P-1] \\ \check{y}[k, \hat{\boldsymbol{\alpha}}(t)] = \bar{y}(k, 0) + \begin{bmatrix} \bar{y}(k, 1) & \dots & \bar{y}(k, n_\alpha) \end{bmatrix} \hat{\boldsymbol{\alpha}}(t), \quad k \in [t+P-M, t+P-1] \end{array} \right.$

5)  $\{\hat{\gamma}(t), \sigma_C^2(t)\} = \text{lp}\{\mathbf{r}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)]\}$

6a)  $\left\{ \begin{array}{l} \varepsilon[t, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t), \hat{\mathbf{f}}(t-1)] = \begin{bmatrix} r[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] & \dots & r[t-n_C, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] \end{bmatrix} \hat{\mathbf{c}}(t) \\ \tilde{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t)] = \begin{bmatrix} \check{u}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t)] \\ \vdots \\ \check{u}[t-n_F, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t)] \end{bmatrix} \\ = \begin{bmatrix} \check{u}[t, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{u}[t-n_C, \hat{\boldsymbol{\alpha}}(t)] \\ \vdots & \ddots & \vdots \\ \check{u}[t-n_F, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{u}[t-n_F-n_C, \hat{\boldsymbol{\alpha}}(t)] \end{bmatrix} \hat{\mathbf{c}}(t) \end{array} \right.$

**else**

6b) as in Table 12.2

**end if**

7)–9) as in Table 12.2

**end for**

---

Table 12.4: PEM-based AFC algorithm: PZLP near-end tonal components model

---

for  $t$

if  $j = t \bmod P = 0$

- 1)  $d[k, \hat{\mathbf{f}}(t-1)] = y(k) - [u(k) \dots u(k-n_F)] \hat{\mathbf{f}}(t-1), \quad k \in [t+P-M-\max(n_A, n_C), t+P-1]$
- 2)  $\mathbf{w}[t, \hat{\gamma}(t-P), \hat{\mathbf{f}}(t-1)] = \begin{bmatrix} d[t+P-M, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-M-n_C, \hat{\mathbf{f}}(t-1)] \\ \vdots & & \vdots \\ d[t+P-1, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-1-n_C, \hat{\mathbf{f}}(t-1)] \end{bmatrix} \hat{\mathbf{c}}(t-P)$
- 3)  $\{\hat{\boldsymbol{\alpha}}(t), \sigma_A^2(t)\} = \text{pzlp}\{\mathbf{w}[t, \hat{\gamma}(t-P), \hat{\mathbf{f}}(t-1)]\}$
- 4)  $\mathbf{r}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] = \begin{bmatrix} d[t+P-M, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-M-n_A, \hat{\mathbf{f}}(t-1)] \\ \vdots & & \vdots \\ d[t+P-1, \hat{\mathbf{f}}(t-1)] & \dots & d[t+P-1-n_A, \hat{\mathbf{f}}(t-1)] \end{bmatrix} \hat{\mathbf{a}}(t)$   

$$- \begin{bmatrix} r[t+P-M-1, \hat{\boldsymbol{\alpha}}(t-P), \hat{\mathbf{f}}(t-1)] & \dots & r[t+P-M-n_A, \hat{\boldsymbol{\alpha}}(t-P), \hat{\mathbf{f}}(t-1)] \\ r[t+P-M, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] & \dots & r[t+P-M-n_A+1, \hat{\boldsymbol{\alpha}}(t-P), \hat{\mathbf{f}}(t-1)] \\ \vdots & & \vdots \\ r[t+P-2, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] & \dots & r[t+P-1-n_A, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] \end{bmatrix} \hat{\mathbf{b}}(t)$$
- 5)  $\{\hat{\gamma}(t), \sigma_C^2(t)\} = \text{lp}\{\mathbf{r}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)]\}$
- 6a) 
$$\left\{ \begin{aligned} \varepsilon[t, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t), \hat{\mathbf{f}}(t-1)] &= [r[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] \dots r[t-n_C, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t-1)] \hat{\mathbf{c}}(t) \\ \begin{bmatrix} \check{u}[t-n_F-n_C, \hat{\boldsymbol{\alpha}}(t)] \\ \vdots \\ \check{u}[t+P-1, \hat{\boldsymbol{\alpha}}(t)] \end{bmatrix} &= \begin{bmatrix} u(t-n_F-n_C) & \dots & u(t-n_F-n_C-n_A) \\ \vdots & & \vdots \\ u(t+P-1) & \dots & u(t+P-1-n_A) \end{bmatrix} \hat{\mathbf{a}}(t) \\ \begin{bmatrix} \check{u}[t-n_F-n_C-1, \hat{\boldsymbol{\alpha}}(t-P)] & \dots & \check{u}[t-n_F-n_C-n_A, \hat{\boldsymbol{\alpha}}(t-P)] \\ \check{u}[t-n_F-n_C, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{u}[t-n_F-n_C-n_A+1, \hat{\boldsymbol{\alpha}}(t-P)] \\ \vdots & & \vdots \\ \check{u}[t+P-2, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{u}[t+P-1-n_A, \hat{\boldsymbol{\alpha}}(t)] \end{bmatrix} & \hat{\mathbf{b}}(t) \\ \begin{bmatrix} \check{y}[t-n_C+1, \hat{\boldsymbol{\alpha}}(t)] \\ \vdots \\ \check{y}[t+P-1, \hat{\boldsymbol{\alpha}}(t)] \end{bmatrix} &= \begin{bmatrix} y(t-n_C+1) & \dots & u(t-n_C+1-n_A) \\ \vdots & & \vdots \\ y(t+P-1) & \dots & y(t+P-1-n_A) \end{bmatrix} \hat{\mathbf{a}}(t) \\ \begin{bmatrix} \check{y}[t-n_C, \hat{\boldsymbol{\alpha}}(t-P)] & \dots & \check{y}[t-n_C+1-n_A, \hat{\boldsymbol{\alpha}}(t-P)] \\ \check{y}[t-n_C+1, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{y}[t-n_C+2-n_A, \hat{\boldsymbol{\alpha}}(t-P)] \\ \vdots & & \vdots \\ \check{y}[t+P-2, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{y}[t+P-1-n_A, \hat{\boldsymbol{\alpha}}(t)] \end{bmatrix} & \hat{\mathbf{b}}(t) \\ \check{\mathbf{u}}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t)] &= \begin{bmatrix} \check{u}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t)] \\ \vdots \\ \check{u}[t-n_F, \hat{\boldsymbol{\alpha}}(t), \hat{\gamma}(t)] \end{bmatrix} = \begin{bmatrix} \check{u}[t, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{u}[t-n_C, \hat{\boldsymbol{\alpha}}(t)] \\ \vdots & & \vdots \\ \check{u}[t-n_F, \hat{\boldsymbol{\alpha}}(t)] & \dots & \check{u}[t-n_F-n_C, \hat{\boldsymbol{\alpha}}(t)] \end{bmatrix} \hat{\mathbf{c}}(t) \end{aligned} \right.$$

else

6b) as in Table 12.2

end if

7)-9) as in Table 12.2

end for

---

## 12.6 Computational Complexity and Model Approximations

### 12.6.1 Computational Complexity

The computational complexity of the PEM-based AFC algorithms with cascaded near-end signal models can be quantified in terms of the average number of multiplications that have to be performed in each recursion. This complexity measure is shown in Table 12.5 for the five different near-end tonal components models, and also for the existing PEM-AFROW [32] and NLMS [34, Ch. 9] algorithms. The complexity measure has been calculated individually for each of the nine steps in the algorithm, such that the expressions in Table 12.5 can be easily compared with the corresponding descriptions given in Tables 12.2-12.4.

Before interpreting the expressions in Table 12.5, we should define the variables that have not appeared earlier: the PZLP model coefficients are estimated using the CPZLP line search optimization algorithm, which requires on average  $\beta$  backtracking steps per iteration and  $\bar{k}$  iterations per parameter  $\theta_i(t)$  in  $\alpha(t)$  [53]. The fractional 3-tap pitch prediction method for estimating the PLP model coefficients requires the specification of limits  $K_{\min}$  and  $K_{\max}$  for the pitch lag  $K$ , and its computational complexity depends on the related quantities  $\Delta_K \triangleq K_{\max} - K_{\min}$  and  $\Sigma_K \triangleq K_{\max} + K_{\min}$ , as well as on the order  $n_I$  of the fractional interpolation filter  $I(q, l/D)$ . The SLP model coefficients are estimated after downsampling the analyzed signal, which requires the inclusion of an anti-aliasing filter of order  $n_S$ .

The relative complexity of the different steps in the algorithm depends on the application area. In room acoustic applications, the required adaptive filter order  $n_F$  is typically much larger (i.e., several orders of magnitude) than the near-end signal model orders  $n_A$ ,  $n_I$ ,  $n_\alpha$ , and  $n_C$ , and usually a few times larger than the data window length  $M$  and hop size  $P$ . As a consequence, the main extra complexity of the PEM-based algorithms in room acoustic applications is in steps 1 and 6, when compared to the NLMS complexity. Moreover, since the data window hop size  $P$  is often significantly larger than the near-end signal model orders, the complexity of step 6 comes close to the NLMS complexity of  $n_F + 1$  multiplications, hence the overall increase in complexity can almost completely be attributed to step 1 and approximately equals  $2(n_F + 1)$  multiplications (since we have suggested to choose  $M = 2P$ ), which is 50 % of the overall NLMS complexity. Note that when the WLP near-end tonal components model is used, step 4 approximately involves another  $2(n_F + 1)$  multiplications such that the overall complexity is about twice the NLMS complexity. In HA applications,  $n_F$  is usually also larger than the near-end signal model orders  $n_A$ ,  $n_I$ ,  $n_\alpha$ , and  $n_C$ , but similar to the squared near-end signal model orders  $n_A^2$ ,  $n_I^2$ ,  $n_\alpha^2$ , and  $n_C^2$  and the multiplied orders  $n_A n_C$ ,  $n_I n_C$ , and  $n_\alpha n_C$ . Consequently, steps 3, 5, and 6 contribute more significantly to the

Table 12.5: Complexity comparison: average number of multiplications per recursion

	1)	2)	3)			
NLMS	0	0	0			
PEM-AFROW	$\frac{M}{P}(n_F+1)$	0	0			
$H_1 = \text{LP}$	$\frac{M+\max(n_A, n_C)}{P}(n_F+1)$	$\frac{M}{P}n_C$	$\frac{1}{P}n_A^2 + \frac{M+4}{P}n_A + \frac{M}{P}$			
$H_1 = \text{PZLP}$	$\frac{M+\max(n_A, n_C)}{P}(n_F+1)$	$\frac{M}{P}n_C$	$\frac{\bar{\kappa}[(13+3\bar{\beta})M+(17+5\bar{\beta})]n_A}{2P}$			
$H_1 = \text{PLP}$	$\frac{M+\max(n_A, n_C)}{P}(n_F+1)$	$\frac{M}{P}n_C$	$-\frac{2}{P}n_I^2 + \frac{4M\Delta_K - 5\Sigma_K + 6M}{2P}n_I + \frac{2(M+1)\Delta_K - 5\Sigma_K + 6M + 38}{2P}$			
$H_1 = \text{WLP}$	$\frac{M+n_C}{P}(n_F+1)$	$\frac{M}{P}n_C$	$\frac{1}{P}n_\alpha^2 + \frac{2M+4}{P}n_\alpha + \frac{M}{P}$			
$H_1 = \text{SLP}$	$\frac{M+\max(\Gamma n_\alpha, n_C)}{P}(n_F+1)$	$\frac{M}{P}n_C$	$\frac{1}{P}n_\alpha^2 + \frac{M/\Gamma+4}{P}n_\alpha + \frac{M}{P}(\frac{1}{\Gamma}+n_S+1)$			
	4)	5)				
NLMS	0	0				
PEM-AFROW	0	$\frac{1}{P}n_C^2 + \frac{M+4}{P}n_C + \frac{M}{P}$				
$H_1 = \text{LP}$	$\frac{M}{P}n_A$	$\frac{1}{P}n_C^2 + \frac{M+4}{P}n_C + \frac{M}{P}$				
$H_1 = \text{PZLP}$	$\frac{2M}{P}n_A$	$\frac{1}{P}n_C^2 + \frac{M+4}{P}n_C + \frac{M}{P}$				
$H_1 = \text{PLP}$	$\frac{M}{P}(n_I+3)$	$\frac{1}{P}n_C^2 + \frac{M+4}{P}n_C + \frac{M}{P}$				
$H_1 = \text{WLP}$	$\frac{M+n_\alpha}{P}(n_F+1) + \frac{2(M+P)-1}{P}n_\alpha + 4$	$\frac{1}{P}n_C^2 + \frac{M+4}{P}n_C + \frac{M}{P}$				
$H_1 = \text{SLP}$	$\frac{M}{P}n_\alpha$	$\frac{1}{P}n_C^2 + \frac{M+4}{P}n_C + \frac{M}{P}$				
	6)		7)	8)	9)	
NLMS	$n_F+1$		0	$2(n_F+2)$	$n_F+1$	
PEM-AFROW	$\frac{P+n_C}{P}(n_F+1) + \frac{2P-1}{P}n_C$		4	$2(n_F+2)$	$n_F+1$	
$H_1 = \text{LP}$	$\frac{P+n_A+n_C-1}{P}(n_F+1) + \frac{2}{P}n_A n_C + \frac{2(P-1)}{P}(n_A+n_C)$		4	$2(n_F+2)$	$n_F+1$	
$H_1 = \text{PZLP}$	$\frac{P+2n_A+n_C-1}{P}(n_F+1) + \frac{4}{P}n_A n_C + \frac{2(P-1)}{P}(2n_A+n_C)$		4	$2(n_F+2)$	$n_F+1$	
$H_1 = \text{PLP}$	$\frac{P+n_I+n_C+2}{P}(n_F+1) + \frac{2}{P}(n_I+3)n_C + \frac{2(P-1)}{P}(n_I+n_C+3)$		4	$2(n_F+2)$	$n_F+1$	
$H_1 = \text{WLP}$	$\frac{P+n_C-1}{P}(n_F+1) + 2n_C$		4	$2(n_F+2)$	$n_F+1$	
$H_1 = \text{SLP}$	$\frac{P+n_\alpha+n_C-1}{P}(n_F+1) + \frac{2}{P}n_\alpha n_C + \frac{2(P-1)}{P}(n_\alpha+n_C)$		4	$2(n_F+2)$	$n_F+1$	

overall complexity than in the room acoustic case. However, this contribution is negligible for  $P \geq \{n_A, n_I, n_\alpha, n_C\}$ . Finally, an important feature of the PEM-based algorithms is that no additional complexity is introduced in the adaptive filtering part of the algorithm (i.e., steps 7–9), so when using a more demanding adaptive filtering algorithm like the recursive least squares (RLS) or affine projection algorithm (APA), the extra complexity of the PEM-based algorithms does not increase accordingly.

### 12.6.2 Model Approximations

In the PEM-based AFC algorithms, the data vectors that are needed for the identification of  $\mathbf{f}(t)$ ,  $\boldsymbol{\alpha}(t)$ , and  $\boldsymbol{\gamma}(t)$  are recalculated entirely once in every  $P$  recursions, see steps 1, 2, 4, and 6a in the algorithms given in Tables 12.2-12.4. These prefiltering operations may contribute significantly to the overall computational complexity, as can be seen from Table 12.5. However, by applying certain model approximations, the number of prefiltering operations can be reduced significantly without sacrificing too much of the AFC performance.

These model approximations are related to the stationarity of the acoustic feedback path  $F(q, t)$  and the near-end signal models  $H_1(q, t)$  and  $H_2(q, t)$ . If these models are assumed to be piecewise stationary with time scales of  $Q_F + 1$ ,  $Q_{H_1} + 1$ , and  $Q_{H_2} + 1$  samples, respectively, then the corresponding model estimates  $\hat{F}(q, t)$ ,  $\hat{H}_1(q, t)$ , and  $\hat{H}_2(q, t)$  can be assumed equal on similar time scales, i.e.,

$$F(q, t - Q_F) = \dots = F(q, t) \Rightarrow \hat{F}(q, t - Q_F) = \dots = \hat{F}(q, t) \quad (12.46)$$

$$H_1(q, t - Q_{H_1}) = \dots = H_1(q, t) \Rightarrow \hat{H}_1(q, t - Q_{H_1}) = \dots = \hat{H}_1(q, t) \quad (12.47)$$

$$H_2(q, t - Q_{H_2}) = \dots = H_2(q, t) \Rightarrow \hat{H}_2(q, t - Q_{H_2}) = \dots = \hat{H}_2(q, t). \quad (12.48)$$

Obviously, the above approximations are only exact if the time index  $t$  corresponds to the final time index of a stationarity time interval for  $F(q, t)$ ,  $H_1(q, t)$ , and  $H_2(q, t)$ , and if the model estimates have zero variance. Nevertheless, we will apply (12.46)-(12.48) without explicitly assuming that these two conditions are fulfilled.

When applying the approximations in (12.46)-(12.48) to the algorithms given in Tables 12.2-12.4, we can apply the following simplifications:

- In step 1, we can approximate  $\hat{\mathbf{f}}(t-1)$  by  $\hat{\mathbf{f}}(k)$  for  $k = t - Q_F - 1, \dots, t - 2$  such that  $d[k, \hat{\mathbf{f}}(t-1)]$  is approximated by  $d[k, \hat{\mathbf{f}}(k)]$ , which is the a posteriori feedback-compensated signal that has been calculated in step 9 of the  $k$ th recursion. This simplification leads to an average computational saving of  $\frac{n_F+1}{P} \min[Q_F, M - P + \max(n_A, n_C)]$  multiplications per recursion (or  $\frac{n_F+1}{P} \min(Q_F, M - P + n_C)$  multiplications in the WLP case).



- In step 2, we may replace  $\hat{\mathbf{c}}(t - P)$  by  $\hat{\mathbf{c}}(t - lP)$ ,  $l = 2, \dots, \lfloor \frac{Q_{H_2}}{P} \rfloor + 1$ , such that only the first  $M - (\lfloor \frac{Q_{H_2}}{P} \rfloor + 1)P$  and the last  $P$  elements of the prefiltered data vector  $\mathbf{w}[t, \hat{\boldsymbol{\gamma}}(t - P), \hat{\mathbf{f}}(t - 1)]$  have to be recalculated using  $\hat{\mathbf{c}}(t - P)$ , while the other elements are copied and shifted from the previous data vector  $\mathbf{w}[t - P, \hat{\boldsymbol{\gamma}}(t - 2P), \hat{\mathbf{f}}(t - P - 1)]$ . In this way, an average number of  $n_C \min(\lfloor \frac{Q_{H_2}}{P} \rfloor, \frac{M-P}{P})$  multiplications per recursion can be saved.
- In step 4, a similar approximation  $\hat{\mathbf{a}}(t) = \hat{\mathbf{a}}(t - lP)$ ,  $l = 1, \dots, \lfloor \frac{Q_{H_1}}{P} \rfloor$  may again lead to a computational saving: on average  $n_A \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{M-P}{P})$  multiplications per recursion in the LP case,  $2n_A \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{M-P}{P})$  in the PZLP case,  $(n_I + 3) \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{M-P}{P})$  in the PLP case, and  $n_\alpha \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{M-P}{P})$  in the SLP case. In the WLP case, the calculation of  $\check{u}[k, \hat{\boldsymbol{\alpha}}(t)]$  and  $\check{y}[k, \hat{\boldsymbol{\alpha}}(t)]$  can be simplified in a similar way, saving on average  $n_\alpha \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{M-P+n_F}{P})$  and  $n_\alpha \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{M-P}{P})$  multiplications per recursion, respectively, while the computation of  $\mathbf{r}[t, \hat{\boldsymbol{\alpha}}(t), \hat{\mathbf{f}}(t - 1)]$  can be simplified using (12.46) to save on average  $\frac{n_F+1}{P} \min(Q_F, M - P)$  multiplications per recursion.
- In step 6a, approximating  $\hat{\mathbf{a}}(t) = \hat{\mathbf{a}}(t - lP)$ ,  $l = 1, \dots, \lfloor \frac{Q_{H_1}}{P} \rfloor$  may save  $n_A [\min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{n_F+n_C}{P}) + \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{n_C-1}{P})]$  multiplications per recursion in the LP case,  $2n_A [\min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{n_F+n_C}{P}) + \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{n_C-1}{P})]$  in the PZLP case,  $(n_I + 3) [\min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{n_F+n_C}{P}) + \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{n_C-1}{P})]$  in the PLP case, and  $n_\alpha [\min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{n_F+n_C}{P}) + \min(\lfloor \frac{Q_{H_1}}{P} \rfloor, \frac{n_C-1}{P})]$  in the SLP case. Approximating  $\hat{\mathbf{c}}(t) = \hat{\mathbf{c}}(t - lP)$ ,  $l = 1, \dots, \lfloor \frac{Q_{H_2}}{P} \rfloor$  further leads to a saving of  $n_C \min(\lfloor \frac{Q_{H_2}}{P} \rfloor, \frac{n_F}{P})$  multiplications per recursion for all cases.

Since the data window size  $M$  should be chosen as large as possible without violating the assumption that the near-end signal models are stationary in the entire data window, we typically have  $M \approx Q_{H_1} \approx Q_{H_2}$ . The stationarity time scale of the acoustic feedback path depends heavily on the nature of the changes in the acoustic environment. In PA applications, variations in room acoustics are mainly due to microphone/loudspeaker movements, people moving around the room, and temperature variations. The time scale of room acoustic variations due to moving people (hence also due to objects being moved by people) has been estimated to be around 10 ms for wideband audio applications, while temperature variations are considerably slower [58]. In HA applications, the largest feedback path variations have been found to result from external effects (e.g., by using a telephone set or due to changes in the enclosing room acoustics) [59], hence the variability time scale may be assumed similar to that found in PA applications. When using 50 % overlapping data windows of 40–60 ms, e.g.,  $M = 2P = 2048$  at  $f_s = 44.1$  kHz, the main computational overhead

of approximately  $2(n_F + 1)$  multiplications (due to step 1) can be reduced to  $1.6(n_F + 1)$  multiplications in fast changing environments ( $Q_F = 10 \text{ ms} \times 44.1 \text{ kHz} = 441$ ), or  $n_F + 1$  multiplications in slowly changing environments ( $Q \geq M - P + n_C \approx 1058 = 24 \text{ ms} \times 44.1 \text{ kHz}$ ). In this way, the additional complexity of the proposed PEM-based AFC algorithm compared to NLMS reduces to 25–40 % of the overall NLMS complexity.

## 12.7 Simulation Results

We will evaluate the performance of the proposed PEM-based AFC algorithms with cascaded near-end signal models by means of simulation results obtained in two substantially different scenarios. The first scenario is a typical PA scenario (at  $f_s = 44.1 \text{ kHz}$ ), in which the sound of a single musical instrument is picked up by a microphone, amplified, and fed back from the loudspeaker to the microphone through a room acoustic feedback path. The second scenario is related to HA applications, by simulating a HA that processes an incoming classical music signal at  $f_s = 16 \text{ kHz}$ . We should emphasize that, except for the adaptive filter length  $n_F$ , identical values of all the algorithm parameters are used in both simulation scenarios. It can hence be understood that the algorithm parameters are not particularly optimized to provide a good AFC performance in one specific simulation scenario, but instead are chosen such as to be generally applicable.

The algorithm parameters are chosen as follows: at  $f_s = 44.1 \text{ kHz}$ , the data window length  $M = 2048$  and the hop size  $P = M/2 = 1024$ , while at  $f_s = 16 \text{ kHz}$ ,  $M = 1024$  and  $P = M/2 = 512$ . The near-end tonal components model order is chosen such as to be able to model 15 tonal components in each data window, i.e.,  $n_A = 30$  for the LP and PZLP models, whereas  $n_\alpha = 30$  for the WLP and SLP models. The near-end noise components model order is also set to  $n_C = 30$ . A processing delay of  $d_1 = P - 1$  samples is inserted in the electro-acoustic forward path to allow  $P - 1$  future data samples to be included in the LP data window. In this way, the identifiability condition in (12.40) is fulfilled without the need for inserting an additional processing delay  $d_2$  in the adaptive filtering circuit. Moreover, the electro-acoustic forward path contains a hard clipping saturation function to avoid numerical overflow in case of system instability. In the identification of the WLP near-end tonal components model, the warping parameter is chosen such that the warping map approximates the Bark scale as suggested in [60]:  $\lambda_{\text{Bark}}(f_s) = 1.0674 \sqrt{\frac{2}{\pi} \arctan(0.06583 f_s)} - 0.1916$ . The SLP model is identified using a downsampling factor  $\Gamma = 3$  and an anti-aliasing filter of order  $n_S = 20\Gamma = 60$ . The PZLP model is identified using the CPZLP algorithm parameters suggested in [53] and with an initial estimate of  $\hat{\theta}_i^{(0)}(t) = (2\pi 440)/f_s$  for all the PZLP model angles. The PLP

model identification features a pitch lag range between  $K_{\min} = \lfloor f_s/1000 \rfloor$  and  $K_{\max} = \lfloor f_s/100 \rfloor$  corresponding to fundamental frequencies in the range 100–1000 Hz. The interpolation ratio for estimating fractional pitch lag values  $K + l/D$  is set to  $D = 8$ , and the fractional interpolation filter order is chosen as  $n_I = 31$ . The prediction error power  $\sigma_\varepsilon^2(t)$  is estimated using an effective data window length of  $M$  samples by setting the forgetting factor  $\lambda_\varepsilon = 1 - 1/M$ . Finally, the stochastic gradient algorithm for updating the feedback path estimate features a step size  $\mu = 0.005$  and a regularization parameter  $\delta = 10^{-6}$ . Unless mentioned otherwise, no model approximations are applied in the simulations, i.e.,  $Q_F = Q_{H_1} = Q_{H_2} = 0$ .

Both simulation scenarios have a temporal layout made up of four phases of equal duration. During the first phase of the simulation, the electro-acoustic forward path broadband gain factor  $K(t)$  is fixed to a value that would result in a 3 dB gain margin if no AFC algorithm were applied. In the second phase, the gain  $20 \log_{10} K(t)$  is increased linearly with time, until a value is attained that is 10 dB above the gain applied in the first phase. This simulated gain increase resembles the way an AFC algorithm is applied in practice, i.e., PA operators and HA users are expected to turn on the AFC algorithm at a relatively low gain value and subsequently raise the gain to benefit from the MSG increase provided by the AFC algorithm. Moreover, this gain increase leads to an improved AFC convergence, since the ratio of the feedback signal power to the near-end signal power is increased. In the third phase, the gain is fixed to the final gain value in the second phase, while the fourth phase features a simulated acoustic feedback path change.

In the PA simulation scenario, the near-end signal is a 60 s excerpt from the Partita No. 2 in D minor (Allemande) for solo violin by J. S. Bach. The motivation for using a violin piece, is that the violin appears to be a problematic instrument in terms of sound amplification in PA applications, which is probably due to its highly frequency-dependent directivity [61]. The acoustic feedback path impulse response has a length of 100 ms (corresponding to  $n_F = 4410$  samples) and was measured in a medium-sized room. The AFC performance is quantified by evaluating the misadjustment, defined as

$$\text{MA}_F(\text{dB}) = 20 \log_{10} \frac{\|\hat{\mathbf{f}}(t) - \mathbf{f}\|}{\|\mathbf{f}\|} \quad (12.49)$$

and the MSG defined in (12.11) as a function of time. The results when only a near-end tonal components model  $H_1(q, t)$  is used (with  $H_2(q, t) \equiv 1$ ) are shown in Figs. 12.3(a),(c), while the results with cascaded near-end models are displayed in Figs. 12.3(b),(d). In both cases, the NLMS [34, Ch. 6] and PEM-AFROW (with  $n_C = 30$ ) [32] algorithm performance is also included for reference. It can be observed that by only including a near-end tonal components model, the AFC misadjustment as compared to the PEM-AFROW algorithm can be improved when using the PZLP, PLP, and WLP models.

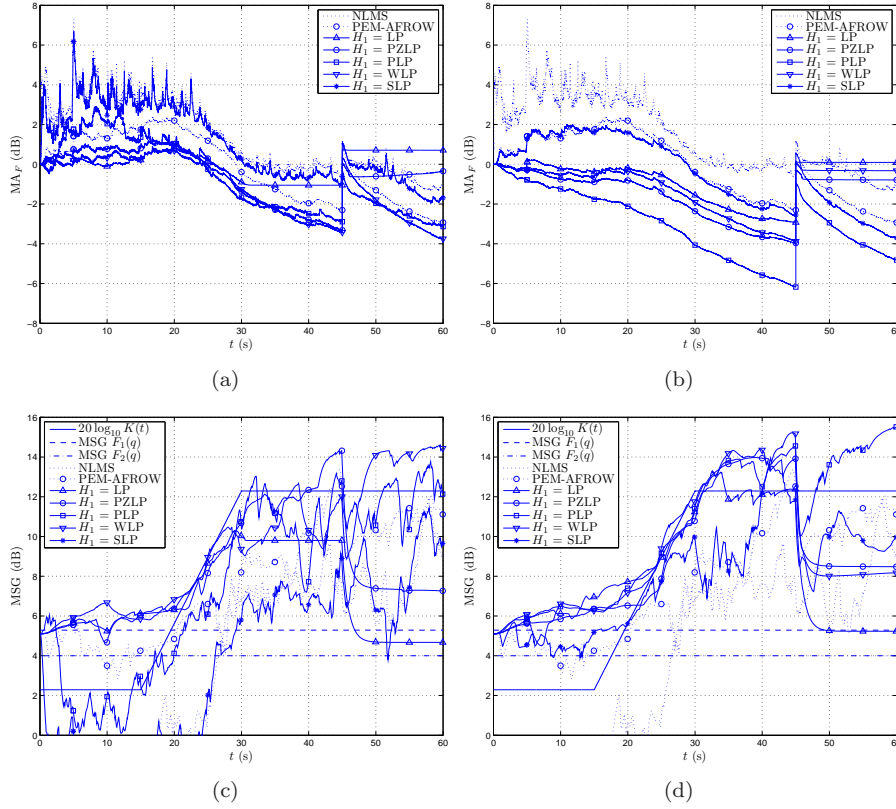


Figure 12.3: Comparison of PEM-based AFC algorithm with NLMS and PEM-AFROW in a PA application: (a) Misadjustment if only a near-end tonal components model  $H_1(q, t)$  is used, (b) Misadjustment if cascaded near-end models  $H_1(q, t)$  and  $H_2(q, t)$  are used, (c) MSG if only a near-end tonal components model  $H_1(q, t)$  is used, (d) MSG if cascaded near-end models  $H_1(q, t)$  and  $H_2(q, t)$  are used.

However, when using the cascaded near-end signal models, a much more significant performance improvement can be obtained, particularly when a PLP near-end tonal components model is applied. Some of the algorithms appear not to be able to cope with an acoustic feedback path change when operating at a high gain value, hence closed-loop instability results (apparent from the horizontal misadjustment curves in Figs. 12.3(a)-(b)). In terms of the MSG, the AFC algorithm should converge fast enough such that its MSG increases at least as fast as the gain factor  $20 \log_{10} K(t)$ , otherwise ringing and howling effects will occur. The best MSG performance is obtained with the LP, PZLP, PLP, and WLP near-end tonal components models cascaded with a noise components model. In Figs. 12.3(c)-(d), the instantaneous gain value  $20 \log_{10} K(t)$ , as well as the MSG values without AFC are also shown (with “MSG  $F_1(q)$ ” and “MSG  $F_2(q)$ ” denoting the MSG before and after the acoustic feedback path change). An MSG increase of more than 11 dB w.r.t. the case when no AFC is applied, is obtained for the cascaded structure with a PLP tonal components model (compared to an 8 dB MSG increase with the PEM-AFROW algorithm).

The HA simulation reflects a scenario in which a HA user is listening to a musical recording or performance. In this simulation, the near-end signal is a 16 s excerpt from the first part (Kyrie) of the Mass in C minor (“Grosse Messe”, K427) by W. A. Mozart, which features a soprano, chorus, and orchestra. The acoustic feedback path is a 12.5 ms measured HA feedback path impulse response, i.e.,  $n_F + 1 = 200$ . The misadjustment and MSG curves are given in Figs. 12.4(a),(c) for a near-end tonal components model only and in Figs. 12.4(b),(d) for cascaded near-end signal models. In contrast to the PA scenario, the HA simulation results indicate that the existing PEM-based AFC algorithms such as PEM-AFROW may work fine in audio applications too, as was also observed in [31],[42]. This can be explained by the fact that the conventional LP model (which is used in these existing PEM-based AFC algorithms) is better suited for modeling audio signals at lower sampling frequencies [40]. However, the AFC performance may be further improved by using cascaded near-end signal models, particularly with a PZLP tonal components model, which clearly results in the fastest AFC convergence and MSG increase. However, the algorithm with the PZLP tonal components model appears to be non-robust to acoustic feedback path changes, hence the PLP tonal components model may be a better choice. The largest MSG increase compared to the MSG without AFC equals more than 8 dB, which is approximately 3 dB larger than the MSG increase obtained with the PEM-AFROW algorithm.

Finally, we evaluate the applicability of the model approximations introduced in Section 12.6.2. In Figs. 12.5(a) and 12.5(b), we show the misadjustment convergence curves for the PEM-based AFC algorithm with cascaded near-end signal models for five different combinations of values for  $Q_F$ ,  $Q_{H_1}$ , and  $Q_{H_2}$ . Each of these variables is either set to zero, or to the value which delivers

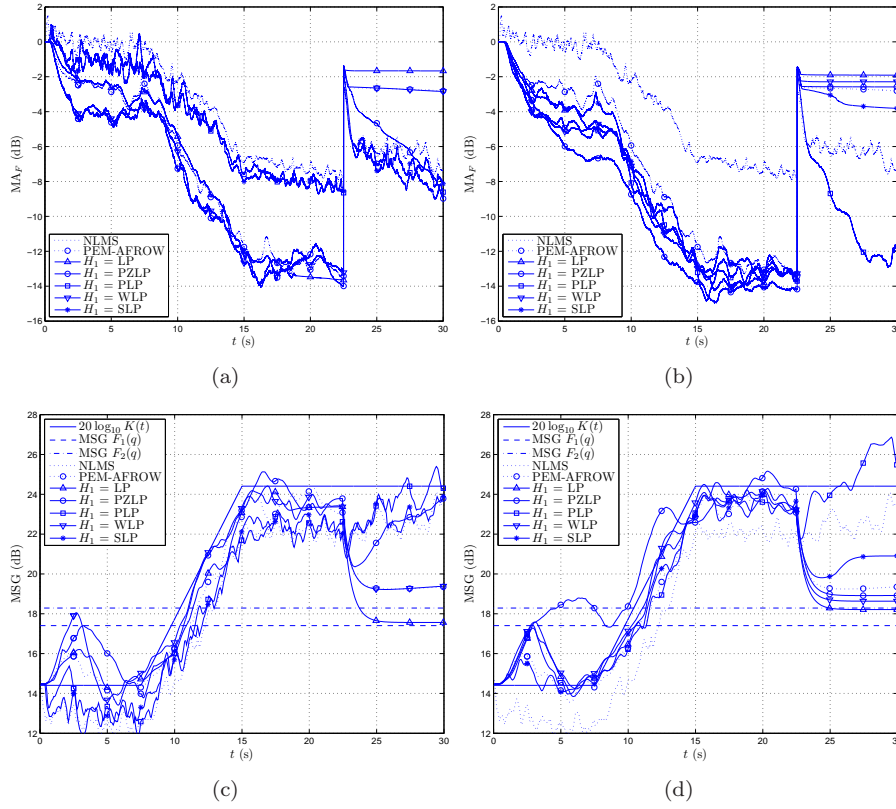


Figure 12.4: Comparison of PEM-based AFC algorithm with NLMS and PEM-AFROW in a HA application: (a) Misadjustment if only a near-end tonal components model  $H_1(q, t)$  is used, (b) Misadjustment if cascaded near-end models  $H_1(q, t)$  and  $H_2(q, t)$  are used, (c)  $MSG$  if only a near-end tonal components model  $H_1(q, t)$  is used, (d)  $MSG$  if cascaded near-end models  $H_1(q, t)$  and  $H_2(q, t)$  are used.

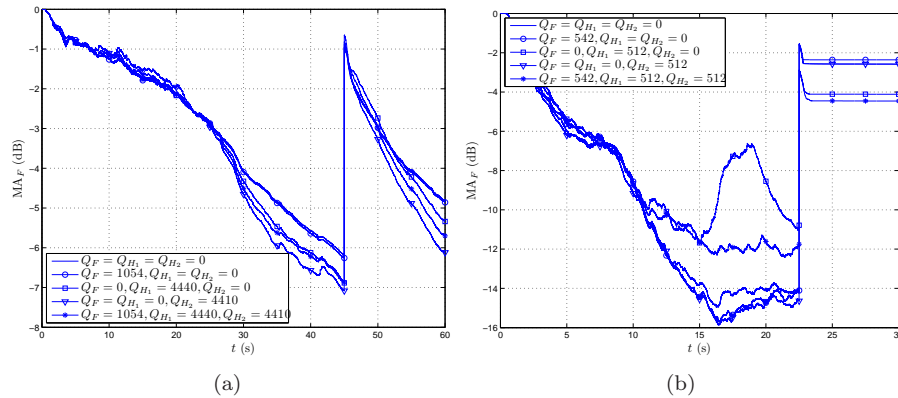


Figure 12.5: Evaluation of model approximations in PEM-based AFC algorithm with cascaded near-end signal models: Misadjustment for different stationarity time scales  $Q_F$ ,  $Q_{H_1}$ , and  $Q_{H_2}$ , (a) PA application (with PLP tonal components model), (b) HA application (with PZLP tonal components model).

the maximum achievable computational saving. It can be seen that in the PA case (with a PLP tonal components model), a rather unexpected performance improvement occurs when  $Q_{H_1}$  and/or  $Q_{H_2}$  are increased. In the HA case (with a PZLP tonal components model), increasing  $Q_{H_2}$  leads to a slightly better performance, while increasing  $Q_{H_2}$  may lead to a severe performance decrease.

## 12.8 Conclusion

In this paper, we have proposed a modification to the PEM-based AFC approach, which consists in replacing the all-pole near-end signal model by a cascade of two linear models. In this way, the PEM-based AFC approach, which was originally designed for speech applications only, can also be applied to audio applications involving musical sound signals. Since the cascaded near-end signal models are able to whiten the near-end signal component in the microphone signal more effectively, a significant AFC performance improvement is obtained. Five different algorithms have been presented, in which the organization of the prefiltering operations is specifically tailored to the underlying near-end tonal components model structure. The computational requirements of these algorithms were analyzed in detail, leading to the conclusion that a complexity increase of 50 % of the overall NLMS complexity can be expected. However, if model approximations related to the stationarity of the feedback path and the near-end signal are applied, the computational overhead can further be reduced to 25 % of the NLMS complexity without significantly

influencing the AFC performance. Both in PA and HA simulation scenarios, a considerable improvement in the acoustic feedback path misadjustment and the MSG has been observed. In particular, the PZLP and PLP near-end tonal components models, which both feature a PEF that behaves as a cascade of notch filters, have been found to offer the best performance.

## Bibliography

- [1] T. van Waterschoot and M. Moonen, “50 years of acoustic feedback control: state of the art and future challenges,” *Proc. IEEE*, submitted for publication, Feb. 2009, ESAT-SISTA Technical Report TR 08-13, Katholieke Universiteit Leuven, Belgium. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/08-13.html>
- [2] C. P. Boner, “A procedure for controlling room-ring modes and feedback modes in sound systems with narrow-band filters,” *J. Audio Eng. Soc.*, vol. 13, no. 4, pp. 297–299, 1965.
- [3] M. P. Lewis, T. J. Tucker, and D. M. Oster, “Method and apparatus for adaptive audio resonant frequency filtering,” U.S. Patent 5,245,665, Sep., 1993.
- [4] M. Börsch, “Method for constraining electroacoustic feedback,” European Patent Application EP1684543 A1, Jul., 2006.
- [5] N. Osmanovic, V. E. Clarke, and E. Velandia, “An in-flight low latency acoustic feedback cancellation algorithm,” in *Preprints AES 123rd Convention*, New York, NY, USA, Oct. 2007, AES Preprint 7266.
- [6] J. A. Moorer, “The manifold joys of conformal mapping: Applications of digital filtering in the studio,” *J. Audio Eng. Soc.*, vol. 31, no. 11, pp. 826–841, Nov. 1983.
- [7] P. A. Regalia and S. K. Mitra, “Tunable digital frequency response equalization filters,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 1, pp. 118–120, Jan. 1987.
- [8] T. van Waterschoot and M. Moonen, “A pole-zero placement technique for designing second-order IIR parametric equalizer filters,” *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 8, pp. 2561–2565, Nov. 2007.
- [9] Y. Takahashi, M. Tohyama, and Y. Yamasaki, “Cumulative spectral analysis for transient decaying signals in a transmission system including a feedback loop,” *J. Audio Eng. Soc.*, vol. 54, no. 7/8, pp. 620–629, July/Aug. 2006.



- [10] G. Rombouts, T. van Waterschoot, and M. Moonen, "Proactive notch filtering for acoustic feedback cancellation," in *Proc. 2nd Annual IEEE Benelux/DSP Valley Signal Process. Symp. (SPS-DARTS '06)*, Antwerp, Belgium, Mar. 2006, pp. 169–172. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/06-81.html>
- [11] —, "Robust and efficient implementation of the PEM-AFROW algorithm for acoustic feedback cancellation," *J. Audio Eng. Soc.*, vol. 55, no. 11, pp. 955–966, Nov. 2007.
- [12] M. G. Siqueira and A. Alwan, "Steady-state analysis of continuous adaptation in acoustic feedback reduction systems for hearing-aids," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 4, pp. 443–453, Jul. 2000.
- [13] J. Hellgren and U. Forssell, "Bias of feedback cancellation algorithms in hearing aids based on direct closed loop identification," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 7, pp. 906–913, Nov. 2001.
- [14] T. van Waterschoot, G. Rombouts, and M. Moonen, "On the performance of decorrelation by prefiltering for adaptive feedback cancellation in public address systems," in *Proc. 4th IEEE Benelux Signal Process. Symp. (SPS '04)*, Hilvarenbeek, The Netherlands, Apr. 2004, pp. 167–170. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/04-24.html>
- [15] P. Estermann and A. Kaelin, "Feedback cancellation in hearing aids: results from using frequency-domain adaptive filters," in *Proc. 1994 IEEE Int. Symp. Circuits Syst. (ISCAS '94)*, vol. 2, London, UK, May/June 1994, pp. 257–260.
- [16] C. P. Janse and P. A. A. Timmermans, "Signal amplifier system with improved echo cancellation," U.S. Patent 5,748,751, May, 1998.
- [17] C. Boukis, D. P. Mandic, and A. G. Constantinides, "Towards bias minimization in acoustic feedback cancellation systems," *J. Acoust. Soc. Amer.*, vol. 121, no. 3, pp. 1529–1537, Mar. 2007.
- [18] H. A. L. Josen, F. Asano, Y. Suzuki, and T. Sone, "Adaptive feedback cancellation with frequency compression for hearing aids," *J. Acoust. Soc. Amer.*, vol. 94, no. 6, pp. 3248–3254, Dec. 1993.
- [19] S. Kamerling, K. Janse, and F. van der Meulen, "A new way of acoustic feedback suppression," in *Preprints AES 104th Convention*, Amsterdam, The Netherlands, May 1998, AES Preprint 4735.
- [20] T. van Waterschoot, K. Eneman, and M. Moonen, "Instrumental variable methods for acoustic feedback cancellation," Katholieke

- Universiteit Leuven, Belgium, Tech. Rep. ESAT-SISTA TR 05-14, Oct. 2004. [Online]. Available: <ftp.esat.kuleuven.be/pub/sista/vanwaterschoot/abstracts/05-14.html>
- [21] G. Schmidt and T. Haulick, "Signal processing for in-car communication systems," *Signal Processing*, vol. 86, no. 6, pp. 1307–1326, Jun. 2006, special Issue on Applied Speech and Audio Processing.
- [22] A. M. Engebretson and M. F. George, "Properties of an adaptive feedback equalization algorithm," *J. Rehab. Res. Dev.*, vol. 30, no. 1, pp. 8–16, 1993.
- [23] A. Goertz, "An adaptive subtraction filter for feedback cancellation in public address sound systems," in *Proc. 15th Int. Congr. Acoust. (ICA '95)*, Trondheim, Norway, Jun. 1995, pp. 69–72.
- [24] J. M. Kates, "Feedback cancellation in hearing aids: Results from a computer simulation," *IEEE Trans. Signal Process.*, vol. 39, no. 3, pp. 553–562, Mar. 1991.
- [25] J. A. Maxwell and P. M. Zurek, "Reducing acoustic feedback in hearing aids," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 4, pp. 304–313, Jul. 1995.
- [26] D. K. Bustamante, T. L. Worrall, and M. J. Williamson, "Measurement and adaptive suppression of acoustic feedback in hearing aids," in *Proc. 1989 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '89)*, vol. 3, Glasgow, UK, 1989, pp. 2017–2020.
- [27] A. Ortega, E. Lleida, and E. Masgrau, "Speech reinforcement system for car cabin communications," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 917–929, Sep. 2005.
- [28] N. A. Shusina and B. Rafaely, "Unbiased adaptive feedback cancellation in hearing aids by closed-loop identification," *IEEE Trans. Audio Speech Lang. Process.*, vol. 14, no. 2, pp. 658–665, Mar. 2006.
- [29] J. Hellgren, "Analysis of feedback cancellation in hearing aids with Filtered-X LMS and the direct method of closed-loop identification," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 2, pp. 119–131, Feb. 2002.
- [30] A. Ortega, E. Lleida, E. Masgrau, L. Buera, and A. Miguel, "Acoustic feedback cancellation in speech reinforcement systems for vehicles," in *Proc. Interspeech 2005*, Lisbon, Portugal, Sep. 2005, pp. 2061–2064.
- [31] A. Spriet, I. Proudler, M. Moonen, and J. Wouters, "Adaptive feedback cancellation in hearing aids with linear prediction of the desired signal," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3749–3763, Oct. 2005.

- [32] G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, "Acoustic feedback suppression for long acoustic paths using a nonstationary source model," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sep. 2006.
- [33] T. van Waterschoot, "Akoestische feedbackonderdrukker," Master's thesis, K.U.Leuven, Departement Elektrotechniek (ESAT), Leuven, Belgium, Jun. 2001, (in Dutch).
- [34] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1996.
- [35] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. Cambridge, Massachusetts: MIT Press, 1986.
- [36] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
- [37] T. van Waterschoot, G. Rombouts, P. Verhoeve, and M. Moonen, "Double-talk-robust prediction error identification algorithms for acoustic echo cancellation," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 846–858, Mar. 2007.
- [38] S. M. Kay, "The effects of noise on the autoregressive spectral estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 5, pp. 478–485, Oct. 1979.
- [39] Y. T. Chan, J. M. M. Lavoie, and J. B. Plant, "A parameter estimation approach to estimation of frequencies of sinusoids," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 2, pp. 214–219, Apr. 1981.
- [40] T. van Waterschoot and M. Moonen, "Comparison of linear prediction models for audio signals," *EURASIP J. Audio, Speech, Music Process.*, vol. 2008, Article ID 706935, 24 pages, 2008, doi:10.1155/2008/706935.
- [41] H. Nyquist, "Regeneration theory," *Bell Syst. Tech. J.*, vol. 11, pp. 126–147, 1932.
- [42] A. Spriet, K. Eneman, M. Moonen, and J. Wouters, "Objective measures for real-time evaluation of adaptive feedback cancellation algorithms in hearing aids," in *Proc. 16th European Signal Process. Conf. (EUSIPCO '08)*, Lausanne, Switzerland, Aug. 2008.
- [43] G. Rombouts, T. van Waterschoot, K. Struyve, P. Verhoeve, and M. Moonen, "Identification of undermodelled room impulse responses," in *Proc. 2005 Int. Workshop Acoustic Echo Noise Control (IWAENC '05)*, Eindhoven, The Netherlands, Sep. 2005, pp. 153–156.
- [44] U. Forssell and L. Ljung, "Closed-loop identification revisited," *Automatica*, vol. 35, no. 7, pp. 1215–1241, Jul. 1999.

- [45] A. Härmä and U. K. Laine, “A comparison of warped and conventional linear predictive coding,” *IEEE Trans. Speech Audio Process.*, vol. 9, no. 5, pp. 579–588, Jul. 2001.
- [46] T. van Waterschoot, G. Rombouts, and M. Moonen, “Dually regularized recursive prediction error identification for acoustic feedback and echo cancellation,” in *Proc. 15th European Signal Process. Conf. (EUSIPCO '07)*, Poznań, Poland, Sep. 2007, pp. 1610–1614.
- [47] M. A. Stone and B. C. J. Moore, “Tolerable hearing aid delays. I. Estimation of limits imposed by the auditory path alone using simulated hearing losses,” *Ear & Hearing*, vol. 20, no. 3, pp. 182–191, Jun. 1999.
- [48] J. Makhoul, “Linear prediction: A tutorial review,” *Proc. IEEE*, vol. 63, no. 4, pp. 561–578, Apr. 1975.
- [49] P. Kroon and B. S. Atal, “Pitch predictors with high temporal resolution,” in *Proc. 1990 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '90)*, vol. 2, Albuquerque, NM, USA, Apr. 1990, pp. 661–664.
- [50] Y. Qian, G. Chahine, and P. Kabal, “Pseudo-multi-tap pitch filters in a low bit-rate CELP speech coder,” *Speech Commun.*, vol. 14, no. 4, pp. 339–358, Sep. 1994.
- [51] R. P. Ramachandran and P. Kabal, “Pitch prediction filters in speech coding,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-37, no. 4, pp. 467–478, Apr. 1989.
- [52] T. van Waterschoot and M. Moonen, “Adaptive feedback cancellation for audio signals using a warped all-pole near-end signal model,” in *Proc. 2008 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '08)*, Las Vegas, NV, USA, Apr. 2008, pp. 269–272.
- [53] —, “Constrained pole-zero linear prediction: an efficient and near-optimal method for multi-tone frequency estimation,” in *Proc. 16th European Signal Process. Conf. (EUSIPCO '08)*, Lausanne, Switzerland, Aug. 2008.
- [54] T. van Waterschoot, M. Diehl, and M. Moonen, “Constrained pole-zero linear prediction: Optimization of cascaded biquadratic notch filters for multi-tone and multi-pitch estimation,” to be submitted for publication, Mar. 2009, ESAT-SISTA Technical Report TR 07-115, Katholieke Universiteit Leuven, Belgium, Feb. 2008.
- [55] A. Nehorai, “A minimal parameter adaptive notch filter with constrained poles and zeros,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 4, pp. 983–996, Aug. 1985.

- [56] M. V. Dragošević and S. S. Stanković, “An adaptive notch filter with improved tracking properties,” *IEEE Trans. Signal Process.*, vol. 43, no. 9, pp. 2068–2078, Sep. 1995.
- [57] G. Li, “A stable and efficient adaptive notch filter for direct frequency estimation,” *IEEE Trans. Signal Process.*, vol. 45, no. 8, pp. 2001–2009, Aug. 1997.
- [58] L. M. van de Kerkhof and W. J. W. Kitzen, “Tracking of a time-varying acoustic impulse response by an adaptive filter,” *IEEE Trans. Signal Process.*, vol. 40, no. 6, pp. 1285–1294, Jun. 1992.
- [59] J. Hellgren, T. Lunner, and S. Arlinger, “Variations in the feedback of hearing aids,” *J. Acoust. Soc. Amer.*, vol. 106, no. 5, pp. 2821–2833, Nov. 1999.
- [60] J. O. Smith and J. S. Abel, “Bark and ERB bilinear transforms,” *IEEE Trans. Speech Audio Process.*, vol. 7, no. 6, pp. 697–708, Nov. 1999.
- [61] H. J. Vos, O. Warusfel, N. Misdariis, and D. de Vries, “Analysis and reproduction of the frequency spectrum and directivity of a violin,” *Journal of the Acoustical Society of the Netherlands*, no. 167, Sep. 2003.



## Chapter 13

# Conclusions and Suggestions for Future Research

### 13.1 Summary and Conclusions

In this thesis, we have designed and evaluated a number of novel room acoustic signal enhancement algorithms for acoustic echo cancellation and acoustic feedback control. We have designed these algorithms with the following requirements in mind: a robust performance, a high sound quality, and a reasonable complexity increase as compared to state-of-the-art algorithms. These goals have been achieved by departing from the traditional ad hoc approach for dealing with problems such as double-talk, poor excitation, and decorrelation, but instead approaching these problems from a parameter estimation and system identification point of view, hence developing theoretically well-founded solutions.

A major part of the thesis has been devoted to the development of so-called PEM-based adaptive filtering algorithms, i.e., algorithms that are based on the prediction error method for system identification [1, Ch. 3],[2, Ch. 7]. These algorithms allow the concurrent modeling and identification of the room acoustics and the near-end signal characteristics in acoustic echo and feedback cancellation. In this way, several open problems in these two applications have been approached from a new perspective and promising solutions have been proposed. A first problem that was tackled with the PEM-based approach is the double-talk problem in AEC. It was shown in **Chapter 3** that the appli-

cation of a PEM-based adaptive filtering algorithm in AEC alleviates the use of a DTD. The corresponding convergence improvement is significant both in a continuous and in a bursting double-talk situation, particularly for underdetermined RLS algorithms such as the NLMS algorithm, while the complexity increase is limited to 35–65 % of the NLMS complexity. A second problem in which the PEM-based approach has proven valuable is the undermodeling problem that occurs when the AEC or AFC adaptive filter order is chosen insufficiently large. In **Chapter 4**, we have verified that the variance resulting from undermodeling can be reduced with 20–35 dB in a scenario in which the adaptive filter order is only 30 % of the true system order, while the bias can be concentrated in a few adaptive filter taps which can afterwards be discarded. A third application of the PEM-based adaptive filtering approach was inspired by hearing aid research [3] and consists in reducing the correlation between the loudspeaker and near-end signals for obtaining an unbiased acoustic feedback path estimate in AFC. The PEM-based hearing aid AFC approach was extended to room acoustic applications in **Chapter 9**, by alleviating several modeling approximations that appear to be invalid in room acoustic speech applications and including a pitch predictor in the decorrelation. The resulting PEM-AFROW algorithm was shown to deliver a 7 dB misadjustment improvement compared to the PEM-based hearing aid AFC algorithm at a complexity increase of only 18 %, while the convergence improvement compared to the NLMS algorithm was found to be 12 dB at the cost of a 50 % complexity increase. In **Chapter 12**, the PEM-based approach was applied to AFC in audio applications by using two cascaded near-end signal models. When the near-end signal tonal components are modeled with a pole-zero or pitch prediction model, an additional 4 dB misadjustment improvement can be obtained compared to the PEM-AFROW algorithm. Moreover, due to the fact that the average stationarity period of audio signals is somewhat longer than for speech signals, the modeling approximations used in the PEM-based hearing aid AFC approach do make sense in room acoustic audio applications, hence the complexity increase compared to the NLMS algorithm can be reduced to 25 %. Finally, in **Chapter 2**, the PEM-based AFC algorithms proposed in this thesis were compared to the state-of-the-art AFC algorithms as well as to other acoustic feedback control methods. The PEM-based AFC approach was found to deliver the best performance in terms of the achievable amplification and the resulting sound quality, while its reliability is slightly worse compared to the state-of-the-art algorithms. The latter observation has motivated us to improve the robustness of the PEM-AFROW algorithm using several enhancements reported in **Chapter 10**. By also increasing the PEM-AFROW efficiency through subband and frequency domain implementations, a real-time AFC system for speech applications was obtained and tested in real-life experiments, thereby confirming the added stable gain values of around 13 dB that were reported in **Chapter 2**.

A second achievement of the thesis is related to the use of regularization in lin-



ear adaptive filtering, and more particularly in room acoustic signal enhancement. In **Chapter 5**, a novel framework for regularization in adaptive filtering was conceived by approaching the associated parameter estimation problem from a Bayesian minimum mean square error point of view. As a consequence, the existing scaled-identity-matrix-based Tikhonov and Levenberg-Marquardt regularization methods could be related to each other, in the sense that these methods were shown to correspond to different assumptions on the mean value of the parameter vector's prior distribution. Moreover, the use of a non-identity regularization matrix was found to be MSE optimal in case prior knowledge on the echo/feedback path as well as on the near-end signal could be measured or estimated. In the underdetermined RLS case, the MSE optimal approach to regularization has been related to the concept of proportionate adaptation, thereby providing a new interpretation of the PAPA and PNLMS algorithms. Two methods for limiting the complexity increase of the regularized RLS algorithms were reviewed, while the complexity increase of the regularized APA and NLMS algorithms was shown to be around 25 % if a diagonal regularization matrix is used. The proposed approach to regularization was found to deliver an increased performance both in AEC and AFC applications, in stationary as well as nonstationary acoustic environments. In **Chapter 6**, "dual" regularization of the near-end signal model estimation was introduced to alleviate numerical problems encountered in some of the PEM-based adaptive filtering algorithms. Again, the use of prior knowledge (through a model of long-term speech characteristics) was shown to have a beneficial effect on the AEC and AFC performance.

A third major contribution of the thesis consists in providing a comprehensive treatment of the state-of-the-art in acoustic feedback control, including results of a comparative evaluation of the state-of-the-art methods as well as the methods proposed in this thesis. In **Chapter 2**, a literature review is presented that spans five decades of research in acoustic feedback control, while the three most widely used acoustic feedback control methods are treated in more detail, thereby discussing conceptual as well as implementation issues. These methods are then compared in terms of the achievable amplification, the sound quality, and the reliability, both in speech and audio sound reinforcement applications.

Finally, we have treated three other research issues in this thesis, which on one hand have aided us for arriving at the novel AEC and AFC algorithms discussed above, but on the other hand may also be regarded as self-contained contributions to signal processing. The first of these contributions is the pole-zero placement design procedure outlined in **Chapter 7** for designing biquadratic parametric equalizer filters in an intuitive yet accurate way. This procedure was successfully applied in **Chapter 2** for the design of notch filters in the two-stage NHS algorithms. A second self-contained contribution of the thesis is the development of a novel parametric frequency estimation method in

**Chapter 8.** The CPZLP method has a few attractive features compared to the state-of-the-art parametric frequency estimation methods and can hence be considered a valuable tool in spectral estimation, particularly due to its low computational complexity and its ability to operate in colored noise. Finally, a third contribution is related to audio signal modeling using linear prediction models. In **Chapter 11**, we have revealed why conventional all-pole models appear to be unsuited for audio signal modeling, and we have reviewed and evaluated five alternative linear prediction models for audio signals.

## 13.2 Suggestions for Future Research

In terms of future research, we want to stress the importance of reducing the AEC and AFC computational complexity, which we consider the greatest challenge in room acoustic signal enhancement. We believe that the fundamental problem concerning computational complexity, lies in the fact that a room acoustic path is traditionally modeled using its impulse response. A large number of coefficients is then required, particularly when operating at a high sampling frequency (e.g., in audio applications), since then the impulse response is more densely sampled. A challenging approach for reducing the model complexity consists in using models different from the impulse response. Since a room acoustic path typically exhibits sharp peaks in its magnitude response, IIR (or pole-zero) models seem to be an appropriate alternative. The use of such models in room acoustics has both been recommended [4]-[6] and discouraged [7],[8]. However, the appeal of these models for room acoustic applications is mainly due to the conjecture that the IIR model denominator coefficients can in fact be assumed time-invariant in a certain acoustic environment, regardless of the loudspeaker and microphone positions [4]. A related model, which also exploits the assumption of time-invariant room acoustic resonance frequencies, is based on the use of orthogonal basis functions such as the discrete-time Laguerre or Kautz functions, which have been evaluated in an AEC context in [9],[10]. This concept could probably be further improved by considering generalized orthonormal basis functions rather than the classical Laguerre and Kautz functions [11],[12].

A related challenge in room acoustic signal enhancement, is to extrapolate the methods proposed in a single-channel context to multi-channel systems. Since the number of acoustic paths to be identified in a multi-channel system equals the number of loudspeakers times the number of microphones, the computational complexity of most single-channel algorithms can be understood to explode in a multi-channel context. Again, the use of IIR models or models based on (generalized) orthogonal basis functions may bring some relief, since, following the arguments in [4],[10], these models would then share a common denominator. Another problem arising in multi-channel room acoustic signal

enhancement is related to the identifiability of the room acoustic path models in case the loudspeaker signals are correlated. This problem has been studied in the context of multi-channel AEC, and is typically tackled using ad hoc solutions such as decorrelating the loudspeaker signals by introducing nonlinear signal distortions [13],[14]. A theoretically more attractive way of reducing the correlation between the loudspeaker signals would be to use adaptive decorrelation filters, similarly to the approach taken in this thesis for reducing the loudspeaker and near-end signal correlation in AFC.

## Bibliography

- [1] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. Cambridge, Massachusetts: MIT Press, 1986.
- [2] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
- [3] A. Spriet, I. Proudler, M. Moonen, and J. Wouters, “Adaptive feedback cancellation in hearing aids with linear prediction of the desired signal,” *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3749–3763, Oct. 2005.
- [4] Y. Haneda, S. Makino, and Y. Kaneda, “Common acoustical pole and zero modeling of room transfer functions,” *IEEE Trans. Speech Audio Process.*, vol. 2, no. 2, pp. 320–328, 1994.
- [5] J. Pongsiri, P. Amin, and C. Thompson, “Modeling the acoustic transfer function of a room,” in *Proc. 12th Int. Conf. Mathematical Comput. Modelling Scientific Comput. (ICMCMSC '99)*, Chicago, Illinois, USA, Aug. 1999.
- [6] T. Gustafsson, J. Vance, H. R. Pota, B. D. Rao, and M. M. Trivedi, “Estimation of acoustical room transfer functions,” in *Proc. 39th IEEE Conf. Decision Control (CDC '00)*, Sydney, Australia, Dec. 2000, pp. 5184–5189.
- [7] S. Gudvangen and S. J. Flockton, “Comparison of pole-zero and all-zero modelling of acoustic transfer functions,” *IEE Electronics Lett.*, vol. 28, no. 21, pp. 1976–1978, 1992.
- [8] A. P. Liavas and P. A. Regalia, “Acoustic echo cancellation: Do IIR models offer better modeling capabilities than their FIR counterparts,” *IEEE Trans. Signal Process.*, vol. 46, no. 9, pp. 2499–2504, Sep. 1998.
- [9] G. W. Davidson and D. D. Falconer, “Reduced complexity echo cancellation using orthonormal functions,” *IEEE Trans. Circuits Syst.*, vol. 38, no. 1, pp. 20–28, Jan. 1991.

- [10] L. S. H. Ngia, "Recursive identification of acoustic echo systems using orthonormal basis functions," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 3, pp. 278–293, May 2003.
- [11] P. S. C. Heuberger, P. M. J. V. den Hof, and O. H. Bosgra, "A generalized orthonormal basis for linear dynamical systems," *IEEE Trans. Autom. Control*, vol. 40, no. 3, pp. 451–465, Mar. 1995.
- [12] B. Ninness, H. Hjalmarsson, and F. Gustafsson, "The fundamental role of general orthonormal bases in system identification," *IEEE Trans. Autom. Control*, vol. 44, no. 7, pp. 1384–1406, Jul. 1999.
- [13] M. M. Sondhi, D. R. Morgan, and J. L. Hall, "Stereophonic acoustic echo cancellation – an overview of the fundamental problem," *IEEE Signal Process. Lett.*, vol. 2, no. 8, pp. 148–151, Aug. 1995.
- [14] J. Benesty, D. R. Morgan, and M. M. Sondhi, "A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 2, pp. 156–165, Mar. 1998.

# Publication List<sup>1</sup>

## International Journal Papers

1. G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, “Acoustic feedback suppression for long acoustic paths using a nonstationary source model,” *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3426–3434, Sept. 2006.
2. T. van Waterschoot, G. Rombouts, P. Verhoeve, and M. Moonen, “Double-talk-robust prediction error identification algorithms for acoustic echo cancellation,” *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 846–858, Mar. 2007.
3. G. Rombouts, T. van Waterschoot, and M. Moonen, “Robust and efficient implementation of the PEM-AFROW algorithm for acoustic feedback cancellation,” *J. Audio Eng. Soc.*, vol. 55, no. 11, pp. 955–966, Nov. 2007.
4. T. van Waterschoot and M. Moonen, “A pole-zero placement technique for designing second-order IIR parametric equalizer filters,” *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 8, pp. 2561–2565, Nov. 2007.
5. T. van Waterschoot, G. Rombouts, and M. Moonen, “Optimally regularized adaptive filtering algorithms for room acoustic signal enhancement,” *Signal Processing*, vol. 88, no. 3, pp. 594–611, Mar. 2008.
6. T. van Waterschoot and M. Moonen, “Comparison of linear prediction models for audio signals,” *EURASIP J. Audio, Speech, Music Process.*, vol. 2008, Article ID 706935, 24 pages, 2008. doi:10.1155/2008/706935
7. T. van Waterschoot and M. Moonen, “Adaptive feedback cancellation for audio applications,” *Signal Processing*, conditionally accepted for publication, Mar. 2009.
8. V. Le Nir, T. van Waterschoot, M. Moonen, and J. Duplicy, “Blind ZP-OFDM parameter estimation in frequency selective channels,”

---

<sup>1</sup>An up-to-date publication list can be accessed online via <http://homes.esat.kuleuven.be/~tvanwate>, where the full-text of most publications, as well as additional material (such as presentation slides, sound fragments, and MATLAB software code) is available for download.

- EURASIP J. Wireless Commun.*, submitted for publication, Nov. 2008.
9. T. van Waterschoot and M. Moonen, “50 years of acoustic feedback control: state of the art and future challenges,” *Proc. IEEE*, submitted for publication, Feb. 2009.
  10. L. Ausiello, T. van Waterschoot, and M. Moonen, “Design and evaluation of frequency-warped sigma delta modulators,” to be submitted for publication, Mar. 2009.
  11. T. van Waterschoot, M. Diehl, and M. Moonen, “Constrained pole-zero linear prediction: optimization of cascaded biquadratic notch filters for multi-tone and multi-pitch estimation,” to be submitted for publication, Mar. 2009.

## International Conference Papers

1. T. van Waterschoot and M. Moonen, “Double-talk robust acoustic echo cancellation with continuous near-end activity,” in *Proc. 13th European Signal Process. Conf. (EUSIPCO '05)*, Antalya, Turkey, Sept. 2005.
2. G. Rombouts, T. van Waterschoot, K. Struyve, and M. Moonen, “Acoustic feedback suppression for long acoustic paths using a nonstationary source model,” in *Proc. 13th European Signal Process. Conf. (EUSIPCO '05)*, Antalya, Turkey, Sept. 2005.
3. T. van Waterschoot, G. Rombouts, and M. Moonen, “Towards optimal regularization by incorporating prior knowledge in an acoustic echo canceller,” in *Proc. 2005 Int. Workshop Acoustic Echo Noise Control (IWAENC '05)*, Eindhoven, The Netherlands, Sept. 2005, pp. 157–160.
4. G. Rombouts, T. van Waterschoot, K. Struyve, P. Verhoeve, and M. Moonen, “Identification of undermodelled room impulse responses,” in *Proc. 2005 Int. Workshop Acoustic Echo Noise Control (IWAENC '05)*, Eindhoven, The Netherlands, Sept. 2005, pp. 153–156.
5. T. van Waterschoot, G. Rombouts, and M. Moonen, “MSE optimal regularization of APA and NLMS algorithms in room acoustic applications,” in *Proc. 2006 Int. Workshop Acoustic Echo Noise Control (IWAENC '06)*, Paris, France, Sept. 2006.
6. T. van Waterschoot and M. Moonen, “Linear prediction of audio signals,” in *Proc. Interspeech 2007*, Antwerp, Belgium, Aug. 2007, pp. 518–521.
7. T. van Waterschoot, G. Rombouts, and M. Moonen, “Dually regularized recursive prediction error identification for acoustic feedback and echo cancellation,” in *Proc. 15th European Signal Process. Conf. (EUSIPCO '07)*, Poznań, Poland, Sept. 2007, pp. 1610–1614.
8. T. van Waterschoot and M. Moonen, “Adaptive feedback cancellation for audio signals using a warped all-pole near-end signal model,” in *Proc.*

- 2008 *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '08)*, Las Vegas, NV, USA, Apr. 2008, pp. 269–272.
9. L. Ausiello, T. van Waterschoot, and M. Moonen, “A first-order frequency-warped sigma delta modulator with improved signal-to-noise ratio,” in *Proc. 16th European Signal Process. Conf. (EUSIPCO '08)*, Lausanne, Switzerland, Aug. 2008.
  10. T. van Waterschoot and M. Moonen, “Constrained pole-zero linear prediction: an efficient and near-optimal method for multi-tone frequency estimation,” in *Proc. 16th European Signal Process. Conf. (EUSIPCO '08)*, Lausanne, Switzerland, Aug. 2008.
  11. T. van Waterschoot and M. Moonen, “Comparative evaluation of howling detection criteria in notch-filter-based howling suppression,” in *Preprints AES 126th Convention*, Munich, Germany, May 2009, to appear.
  12. T. van Waterschoot and M. Moonen, “Assessing the acoustic feedback control performance of adaptive feedback cancellation in sound reinforcement systems,” submitted for publication in *Proc. 17th European Signal Process. Conf. (EUSIPCO '09)*, Glasgow, Scotland, Aug. 2009.
  13. V. Le Nir, T. van Waterschoot, M. Moonen, and J. Duplicy, “Spectral monitoring and parameter estimation for ZP-OFDM signals,” submitted for publication in *Proc. 17th European Signal Process. Conf. (EUSIPCO '09)*, Glasgow, Scotland, Aug. 2009.
  14. P. Gil-Cacho, T. van Waterschoot, M. Moonen, and S. H. Jensen, “Regularized adaptive notch filters for acoustic howling suppression,” submitted for publication in *Proc. 17th European Signal Process. Conf. (EUSIPCO '09)*, Glasgow, Scotland, Aug. 2009.

## National Conference Papers

1. T. van Waterschoot, G. Rombouts, and M. Moonen, “On the performance of decorrelation by prefiltering for adaptive feedback cancellation in public address systems,” in *Proc. 4th IEEE Benelux Signal Process. Symp. (SPS '04)*, Hilvarenbeek, The Netherlands, Apr. 2004, pp. 167–170.
2. T. van Waterschoot, G. Rombouts, K. Struyve, and M. Moonen, “Acoustic echo cancellation in the presence of continuous double-talk,” in *Proc. 1st Annual IEEE Benelux/DSP Valley Signal Process. Symp. (SPS-DARTS '05)*, Antwerp, Belgium, Apr. 2005, pp. 53–57.
3. T. van Waterschoot, G. Rombouts, and M. Moonen, “Optimally regularized recursive least squares for acoustic echo cancellation,” in *Proc. 2nd Annual IEEE Benelux/DSP Valley Signal Process. Symp. (SPS-DARTS '06)*, Antwerp, Belgium, Mar. 2006, pp. 31–34.
4. G. Rombouts, T. van Waterschoot, and M. Moonen, “Proactive notch filtering for acoustic feedback cancellation,” in *Proc. 2nd Annual IEEE*

*Benelux/DSP Valley Signal Process. Symp. (SPS-DARTS '06)*, Antwerp, Belgium, Mar. 2006, pp. 169–172.

## International Conference Abstracts

1. T. van Waterschoot and M. Moonen, “Acoustic feedback cancellation in a closed-loop system identification framework,” in *Abstracts ERNSI Workshop System Identification 2003*, Noordwijkerhout, The Netherlands, Oct. 2003, p. 6.

## National Conference Abstracts

1. T. van Waterschoot and M. Moonen, “Design and evaluation of DSP algorithms for feedback cancellation in public address systems,” in *Abstracts IAP V/22 Study Day*, Louvain-la-Neuve, Belgium, Nov. 2002, p. 10.
2. T. van Waterschoot and M. Moonen, “Adaptive feedback cancellation for audio signals: an application of prediction error identification with cascaded noise models,” in *Abstracts IAP VI/4-DYSCO Study Day*, Brussels, Belgium, Nov. 2008, p. 10.

## Patent Applications

1. G. Rombouts, P. Verhoeve, K. Struyve, T. van Waterschoot, and M. Moonen, “Circuit and method for estimating a room impulse response,” European Patent Application EP1675374 A1, June, 2006.

## Technical Reports

1. T. van Waterschoot, K. Eneman, and M. Moonen, “Instrumental variable methods for acoustic feedback cancellation,” Katholieke Universiteit Leuven, Belgium, Tech. Rep. ESAT-SISTA TR 05-14, Oct. 2004.
2. T. van Waterschoot, G. Rombouts, and M. Moonen, “Software for double-talk-robust acoustic echo cancellation,” Katholieke Universiteit Leuven, Belgium, Tech. Rep. ESAT-SISTA TR 05-162, Oct. 2005.
3. T. van Waterschoot, M. Diehl, and M. Moonen, “Pole optimization in discrete-time Kautz and Laguerre filters,” Katholieke Universiteit Leuven, Belgium, Tech. Rep. ESAT-SISTA TR 07-61, May 2007.



# Curriculum Vitae



Toon van Waterschoot was born in Lier, Belgium, on June 11, 1979. He received the Master's degree in Electrical Engineering from Katholieke Universiteit Leuven (K.U.Leuven), Belgium, in 2001. In 2002, he spent a year as a Teaching Assistant with the Antwerp Maritime Academy (Hogere Zeevaartschool Antwerpen), Belgium, after which he returned to K.U.Leuven to start a Ph.D. research project on acoustic feedback and echo cancellation, under the supervision of Prof. Marc Moonen.

From 2004 to 2007, he was a Research Assistant with the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT), while from 2002 to 2003, and since 2008, he has been a Research Assistant with K.U.Leuven. Since 2005, he has been a Visiting Teaching Assistant at the Advanced Learning and Research Institute of the University of Lugano (Università della Svizzera italiana), Switzerland. From 2003 to 2006, he was involved in the IWT research projects "SMS4PA" (Sound Management System for Public Address) and "SMS4PA-II", in cooperation with Televic, and since 2008, he has been participating in the European FP7 research project "UCCELLS" (Ultra-Wide Band Real-Time Interference Monitoring and Cellular Management Strategies). His research interests are in adaptive signal processing and parameter estimation, with application to room acoustic signal enhancement, speech and audio processing, and spectral estimation.