

KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT INGENIEURSWETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK
Kasteelpark Arenberg 10, B-3001 Heverlee, Belgium

FACULTEIT BIO-INGENIEURSWETENSCHAPPEN
DEPARTEMENT BIOSYST
Kasteelpark Arenberg 30, B-3001 Heverlee, Belgium

MODEL-BASED PREDICTIVE CONTROL TO INCREASE THE CAPACITY OF HARVESTING MACHINERY IN UNCERTAIN CONDITIONS

Promotoren:
Prof. dr. ir. J. De Baerdemaeker
Prof. dr. ir. B. De Moor

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de ingenieurswetenschappen

door

Tom COEN

March 2009

Fall down seven times,
get up eight.

-Japanese proverb-

Dedicated to Luc Coen
who made it all possible

$^{o}1952 - \dagger 2007$



KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT INGENIEURSWETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK
Kasteelpark Arenberg 10, B-3001 Heverlee, Belgium

FACULTEIT BIO-INGENIEURSWETENSCHAPPEN
DEPARTEMENT BIOSYST
Kasteelpark Arenberg 30, B-3001 Heverlee, Belgium

MODEL-BASED PREDICTIVE CONTROL TO INCREASE THE CAPACITY OF HARVESTING MACHINERY IN UNCERTAIN CONDITIONS

Jury:
Prof. dr. ir. H. Hens, voorzitter
Prof. dr. ir. J. De Baerdemaeker, promotor
Prof. dr. ir. B. De Moor, promotor
Prof. dr. ir. H. Ramon
Prof. dr. ir. J. Vandewalle
Prof. dr. ir. G. Deconinck
Prof. dr. ir. M. Diehl
Dr. ir. B. Missotten, CNH Belgium NV
Dr. ir. P. Goulart, Imperial College, London

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de ingenieurswetenschappen

door

Tom COEN

UDC 631.3 March 2009

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

Alle rechten voorbehouden. Niets uit deze uitgave mag worden 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/2009/7515/21
ISBN 978-94-6018-038–5

Voorwoord

Het leven is als een bos,
rechtdoor kan nooit.
-Oleg Popov-

Afstuderen aan de universiteit is een belangrijk keuzemoment. Je staat als het ware in een treinstation, en je moet kiezen welke trein je gaat nemen zonder dat je al weet waar je heen wil. Je keuze op dat moment zegt waarschijnlijk meer over wat je van het leven verwacht dan over waar je eigenlijk heen wil. Sommige mensen willen gewoon verder, en ze nemen de eerste trein die het station aandoet. Voor een aantal blijkt dit de juiste keuze te zijn. Soms valt het ook tegen, en verlaat men de trein bij één van de volgende stations. De manier waarop men omgaat met de vaststelling dat het de verkeerde keuze was, zegt veel over iemands karakter. De ene persoon twijfelt lang, de andere beslist heel snel. Waar de ene zich eerst goed informeert over de andere verbindingen in elk station, zal de andere er gewoon uitstappen en zich dan pas afvragen hoe het verder moet.

Ook al maak je een beredeneerde keuze, toch is het vaak nog steeds een gok. Kies je voor de stoptrein die je de mogelijkheid geeft nog wat rond te kijken, of kies je voor een HST die je heel snel op je bestemming brengt, waar dat ook mogen wezen. Wil je wel snel op je bestemming zijn, als je niet zeker weet dat het de juiste bestemming is?

Doctoraatsstudenten maken in dit station een bijzondere keuze. Ze zijn ervan overtuigd dat achter het bos naast het station een internationaal vliegveld ligt, van waar ze veel meer bestemmingen kunnen bereiken, in kortere tijd. Al wat ze daarvoor moeten doen, is zich een weg zoeken door het bos. Ze nemen een groot risico. Ze verlaten de begane (spoor)wegen, en wagen een sprong in het onbekende. Ze doen dit niet alleen omdat ze menen dat de tocht door het bos naar het vliegveld leidt, maar ook omdat ze ervan overtuigd zijn dat deze tocht op zich een waardevolle en verrijkende ervaring is, die hen zal helpen hun latere bestemming te kiezen.

De moeilijkheidsgraad van deze expeditie is voor iedereen verschillend. De ene sluit zich aan bij een grote groep, onder begeleiding van een ervaren gids, de andere kiest voor een solotocht. Niet iedereen haalt de andere zijde van het bos. Op je reis doorheen het bos kom je immers regelmatig wegen tegen, en dan is de verleiding telkens groot om toch voor deze verharde, gemakkelijkere weg te kiezen. Sommigen draaien ook rondjes in het bos. Een aantal zakt de moed in de schoenen, en keren terug naar het station.

Ik wil dan ook vooreerst mijn ouders, en niet in het minst mijn vader danken, om me steeds aan te moedigen om vol te houden, en de tocht naar het vliegveld verder te zetten. Het is zeker de moeite waard geweest. Ik wil ook mijn moeder bedanken om er voor me te zijn als ik het moeilijk had.

Zonder professor De Baerdemaeker had ik nooit geloofd dat ik de tocht kon volbrengen. Hij heeft me gedurende de hele reis met raad en daad bijgestaan, en me een duwtje in de rug gegeven wanneer ik het nodig had. Ik ben ook het IWT zeer dankbaar om mijn expeditie te financieren. Mijn dank gaat ook naar professor De Moor, die mijn onderzoek mede heeft mogelijk gemaakt.

De vele gesprekken met mijn collega's van de maaidorsergroep hebben bijgedragen tot dit werk, en hebben de vele veldtesten ook aangenamer gemaakt. Carmen en Geert, bedankt daarvoor! De veldtesten zouden niet mogelijk geweest zijn zonder de medewerking van vele landbouwers in het Leuvense, en de onaflatende inzet van Dirk Leroy en Bart Missotten. Nogmaals bedankt Dirk om al die uren met mij (en de machine) rond te rijden. Jan Anthonis heeft me wegwijs gemaakt in het boeiende veld van de controletheorie. Wouter Saeys ben ik uitermate dankbaar voor het advies tijdens mijn doctoraat en het vele naleeswerk achteraf. Bij professor Darius kon ik steeds terecht met statistische vraagstukken. Ik wil ook Christel, Ann en Wim bedanken voor de logistieke ondersteuning. Zij hebben mij verlost van heel wat administratieve rompslomp.

Onderzoek op landbouwmachines is niet mogelijk zonder de medewerking van een constructeur. Zonder de steun van Case New Holland had dit doctoraat nooit tot stand kunnen komen. In het bijzonder wil ik Bart Missotten bedanken, die er steeds voor zorgde dat het nodige materiaal ter beschikking stond. Ook de ondersteuning van Nicolas Monteyne, Olivier Vanhercke, Didier Verhaeghe en het veldtestteam van de balenpers verdienen een vermelding.

I also want to gratefully acknowledge the valuable feedback of Paul Goulart (Imperial College, London, UK), Eric Kerrigan (Imperial College, London, UK) and professor Jan Maciejowski (University of Cambridge, Cambridge, UK). They were willing to free up some time to discuss the concepts presented in this thesis.

Ik wens ook de leden van mijn jury te bedanken voor hun waardevolle inbreng, professoren Ramon, Vandewalle, Deconinck en Diehl, en Bart Missotten en Paul Goulart. Ook mijn welgemeende dank aan professor Hens voor het waarnemen van het voorzitterschap.

Ik wil ook Noël Jans bedanken. Hij heeft me als een echte vriend steeds aangemoedigd tijdens deze tocht, en me op de hoogte gehouden van het reilen en zeilen op de luchthaven.

Dan last but not least wil ik ook Steffi bedanken. Zij heeft me geholpen bij het drukklaar krijgen van deze tekst, en me aangemoedigd om het af te maken. Schatje, bedankt!

Tom Coen
2 maart 2009

Korte Inhoud

Als we wisten wat we deden, heette het geen onderzoek
-Albert Einstein-

De laatste jaren is het belang van automatisatie in de wereld van landbouwmachines enorm toegenomen. Verdere capaciteitsverhogingen en het verbeteren van het comfort van de operatoren is immers enkel nog mogelijk door een verregaande automatisatie van de machines. De belangrijkste uitdaging bij de controle van landbouwmachines in het algemeen, en een maaidorser in het bijzonder, is de interactie tussen de machine en het inkomende gewas. Dit gewas wordt immers gekenmerkt door een hoge mate van variabiliteit. Bovendien bevatten de processen in landbouwmachines vaak aanzienlijke niet-lineariteiten. Deze beide effecten maken het systeem tijdsvariabel. Dit betekent dat de modelonzekerheid bijna even belangrijk is als het model zelf.

Tijdsvariabele, onzekere systemen vormen een belangrijke modelklasse. Er is al veel onderzoek gebeurd om regelsystemen robuust te maken tegen modelonzekerheid, maar meestal gaat men uit van een deterministische visie. In dit werk wordt gefocust op de mogelijkheden van een Stochastische aanpak. Het onderzoek start vanuit het capaciteitscontrole-probleem op de maaidorser, en past de meest geschikte controletechniek toe die beschikbaar is in de academische wereld.

Gezien het belang van beperkingen en het voorspellen van verstoringen, is Modelgebaseerde Predictieve Controle (MPC) een geschikte methodologie voor deze applicatie. In deze thesis wordt een uitbreiding van MPC afgeleid om te kunnen omgaan met stochastische informatie. Dit is een versie van zogenoemde Stochastische MPC. Dit werk bevat implementaties van zowel traditionele als Stochastische MPC, die extensief getest zijn op verschillende real-life controleproblemen op gebied van landbouwautomatisatie.

Deze tekst kan gezien worden als een handleiding om MPC te implementeren op een real-life, onzeker systeem. Tuning is één van de grote uitdagingen van controleapplicaties in de praktijk. Het is mijn persoonlijke overtuiging dat Stochastische MPC de tuning van een regelaar aanzienlijk kan vereenvoudigen. Stochastische MPC maakt het mogelijk om de trade-off tussen controlefout en ingangseffort automatisch bij te stellen afhankelijk van de toestand van het proces. Dit betekent dat de regelaar sneller zal reageren indien de modelonzekerheid klein is, en de regelaar trager zal reageren bij een minder nauwkeurig model om zo stabiliteit te garanderen.

Abstract

What is research, but a blind date with knowledge?
-Will Henry-

In the last decade automation has become a hot topic on agricultural machinery, and is considered the key to further capacity increases and enhanced operator comfort. One of the main challenges of controlling agricultural machinery, and more specifically a combine harvester, is the close interaction between the system (the combine harvester process itself) and the biological variation of the field which influences the process input. Moreover, a machine such as a combine harvester contains nonlinearities which cause any linear model to depend on the current working point. Due to this time-varying nature of the system, the model uncertainty is just as important as the model itself.

Time-varying, uncertain systems form a very important model class. A lot of work has been done to make control systems robust to model uncertainties, but most of these approaches started from a deterministic point of view. This research focuses on the possibilities of a stochastic approach, and is unique for its combination of theory and practice. The research starts from the capacity control problem on the combine harvester, and selects and modifies the most suited technique available in academia.

In view of the importance of constraints and disturbance prediction, Model-based Predictive Control (MPC) is a well-suited control methodology for this application. In the course of this research an extension to MPC to deal with stochastic information is derived, which is in fact a form of so-called Stochastic MPC. Classic MPC as well as Stochastic MPC are also implemented and extensively tested on several real-life control problems in the area of agricultural automation.

This work can be seen as a manual on how to implement MPC on a real-life, uncertain system. It is my personal conviction that Stochastic MPC will greatly simplify one of the major challenges of control theory in practice, namely tuning. The methodology enables an automatic correction of the trade-off between control error and input effort depending on the actual plant condition. This allows swift, aggressive control if the model uncertainty is small, without compromising on stability in situations where the model is less accurate.

Modelgebaseerde Predictieve Controle ter verhoging van de capaciteit van oogstmachines in onzekere omstandigheden

Goed opgevoede mensen spreken anderen tegen
Wijze mensen spreken zichzelf tegen
-Oscar Wilde-

I Inleiding

De hoofddoelstelling van dit werk kan geformuleerd worden als volgt:

> **Doelstelling.** *Ontwikkelen van een praktisch capaciteitscontrolesysteem dat kan omgaan met* ***modelonzekerheid*** *op een* ***optimale*** *manier, gebruik makende van het* ***'Stochastische Modelgebaseerde Predictieve Controle framework'*** *(Stochastische MPC). Het moet mogelijk zijn om de bekomen regelaar te implementeren op de standaard processing* ***hardware*** *die beschikbaar is op de huidige maaidorsers met een aanvaardbare engineering effort. De methodologie dient eveneens* ***gevalidateerd*** *te worden op een andere applicatie.*

II Geschiedenis en ontwikkeling van MPC

De doelstellingen van een controlesysteem zijn divers en afhankelijk van de situatie. Enkele voorbeelden hiervan zijn:

verhogen van de productiviteit: Dit is ongetwijfeld het meest geassocieerd met controletheorie. Processen worden geautomatiseerd om de productiesnelheid te verhogen en de personeelsvereiste te verlagen.

bewaken en verbeteren van de kwaliteit: Voor vele productieprocessen wordt de winst bepaald door de hoeveelheid off-spec productie. Door de opkomst van nieuwe

sensortechnieken kunnen heel wat processen die voorheen enkel door mensen bestuurd konden worden, nu automatisch geregeld worden.

verlagen van de operator belasting: Soms is de doelstelling om de taak van de operator te vereenvoudigen.

Definitie MPC is een controlemethodologie die een optimale ingangssequentie berekend gegeven een doelfunctie, een procesmodel en beperkingen over een bepaalde tijdshorizon. De meeste versies van MPC zijn gebaseerd op tijdsdomein-modellen, hetgeen een eenvoudige interpretatie toelaat.

De doelfunctie hangt af van de toestanden, uitgangen en ingangen. De optimale ingangssequentie is zo bepaald dat de doelfunctie wordt geminimaliseerd. In theorie kan deze functie gelijk welke structuur bezitten, maar in de praktijk (en zo ook in dit onderzoek) is de doelfunctie kwadratisch. De vorm van de doelfunctie bepaalt de oplossingsmethode, en dus ook de vereiste oplossingstijd. De algemene vorm van de doelfunctie is:

$$V(\mathbf{y}_1, \ldots, \mathbf{y}_{N_P}, \mathbf{x}_1, \ldots, \mathbf{x}_{N_P}, \mathbf{u}_0, \ldots, \mathbf{u}_{N_C-1}) =$$

$$\begin{aligned} &\sum_{k=1}^{N_P} (\mathbf{y}_k - \mathbf{y}_{ref,k})^T \mathbf{S}_k (\mathbf{y}_k - \mathbf{y}_{ref,k}) \\ &+ \sum_{j=1}^{N_P} (\mathbf{x}_j - \mathbf{x}_{ref,j})^T \mathbf{Q}_j (\mathbf{x}_j - \mathbf{x}_{ref,j}) \\ &+ \sum_{t=0}^{N_C-1} \mathbf{u}_t^T \mathbf{R}_t \mathbf{u}_t \end{aligned} \tag{1}$$

met $\mathbf{S}_k$, $\mathbf{Q}_j$ en $\mathbf{R}_t$ gewichtsmatrices op respectievelijk tijdsstap k, j en t. $\mathbf{y}_k$ is de procesuitgang op tijdsstap k, $\mathbf{x}_j$ is de toestand op tijdsstap j en $\mathbf{u}_t$ is de procesingang op tijdsstap t. $\mathbf{y}_{ref,k}$ is de referentie-uitgang op tijdsstap k en $\mathbf{x}_{ref,j}$ is de referentietoestand op tijdsstap j.

De predictiehorizon N_P duidt aan voor hoeveel tijdsstappen in de toekomst de procesuitgangen en -toestanden worden berekend en opgenomen in de doelfunctie. De controlehorizon N_C duidt aan voor hoeveel tijdsstappen de ingang als variabel wordt beschouwd. Het aantal optimisatievariabelen is dus evenredig met N_C.

Het procesmodel definieert de relaties tussen de variabelen (ingangen, toestanden en uitgangen) in de doelfunctie. De beperkingen kunnen verschillende vormen aannemen. Het kan gaan om beperkingen op de procesingang, -uitgang en/of -toestand. Deze beperkingen zijn essentieel voor de meeste praktische controleproblemen.

Voordelen MPC biedt heel wat voordelen voor dit type applicatie, zoals:

flexibel omgaan met beperkingen: vaak liggen de economisch meest interessante werkingstoestanden op één of meerdere beperkingen. Een controletheorie die deze beperkingen expliciet mee in rekening kan brengen, biedt dus interessante perspectieven.

eenvoudige tuning: het algoritme kan in eenvoudige termen worden uitgelegd, wat het zeer geschikt maakt voor industriële applicaties. Daarnaast zijn de meeste parameters gewichten in de doelfunctie, wat de interpretatie sterk vereenvoudigt.

tijdsvariabel model: aangezien de regelactie online bepaald wordt, kan er gebruik gemaakt worden van een tijdsvariabel model.

tijdsvariabel controle-objectief: net als het model, kan ook de doelfunctie online worden aangepast. Dit maakt het mogelijk om de doelfunctie aan te passen aan de omstandigheden.

modelgebaseerd: aangezien een procesmodel aan de basis ligt van de regelaar, kunnen gekende vertragingen en dynamica in rekening worden gebracht.

multivariabel: MPC kan ook controleproblemen aan waar er meerdere in- en/of uitgangen aanwezig zijn.

Robuuste MPC Robuuste MPC is de verzamelnaam voor alle MPC-controletechnieken die de modelonzekerheid in rekening brengen om het effect van deze onzekerheid op de regelperformantie of stabiliteit zo klein mogelijk te maken. Al deze technieken maken gebruik van een onzekerheidsbeschrijving. Normbegrensde onzekerheid en polytoop-onzekerheid zijn de meest bekende vormen. De meeste vormen van robuuste MPC gaan uit van een deterministische aanpak. Er wordt verondersteld dat het systeem zich steeds bevindt in een bepaald gebied, en er wordt geen rekening gehouden met de waarschijnlijkheidsverdeling van het model over dat gebied.

Stochastische MPC gaat uit van een stochastische visie op de onzekerheid. Dit betekent dat in plaats van te stellen dat een bepaalde parameter α als volgt begrensd is:

$$\alpha \in [\alpha_{min}, \alpha_{max}], \tag{2}$$

we stellen dat de parameter α een bepaalde waarschijnlijkheidsverdeling heeft, bijvoorbeeld:

$$\alpha \propto \mathcal{N}\left(\bar{\alpha}, \sigma_\alpha^2\right), \tag{3}$$

met $\bar{\alpha}$ de verwachte waarde van α en σ_α de standaarddeviatie.

III Landbouwmachines

Dit hoofdstuk bespreekt de motivatie om landbouwmachines te automatiseren, en geeft ook de nodige achtergrondinformatie over de werking van een maaidorser. Tot slot wordt ook de huidige state of the art van maaidorsers besproken.

III.1 Motivatie

Historische context In de 19e eeuw was de landbouwsector één van de eerste sectoren waar mechanisatie doorbrak. Dit heeft geleid tot een enorme productiviteitsverhoging van de landbouw. Voor de mechanisatie was tijdens de oogstperiodes bijna

iedereen in het veld nodig. Intussen is in België slechts 1% van de bevolking betrokken bij landbouw, en dit terwijl België zelfs een netto uitvoerder is van landbouwproducten. Men kan dus stellen dat mechanisatie van de landbouw aan de basis lag van de ontwikkeling van onze huidige dienstenmaatschappij.

Milieu De laatste jaren worden er door de Europese Commissie alsmaar strengere milieuwetgeving opgelegd aan grote machines zoals maaidorsers. Het gaat dan voornamelijk om het beperken van uitstoot en lawaai.

Daarnaast zijn er allerlei stimuli om het gebruik van pesticides en meststoffen zoveel mogelijk te reduceren. De meest voor de hand liggende manier om dit te doen is plaatsafhankelijk doseren van de stoffen. Dit vereist een verregaande automatisatie van onder andere meststofsproeiers en mestinjecteurs.

Sociale context Omwille van een hele serie incidenten waarbij er schadelijke stoffen in de voedselketen terecht kwamen, is er grote vraag naar traceerbaarheid van landbouwproducten. Ook dit stelt heel wat technologische uitdagingen.

Economische context De vraag naar een verdere automatisatie van landbouwmachines wordt ook in grote mate bepaald door economische motieven, zoals:

schaalgrootte van de klant: er komen meer en meer loonwerkers die zich specialiseren in een aantal gewassen. Deze bedrijven hebben uiteraard grotere budgetten voor machines.

schaalgrootte van de producent: door de aantrekkende markt voor landbouwmachines produceren de grote constructeurs hogere volumes. Dit maakt het mogelijk om meer budget vrij te maken voor innovatie.

vraag naar landbouwproducten: de vraag naar landbouwproducten is de laatste jaren enorm toegenomen, vooral omwille van het stijgend aantal consumenten, de groeiende vleesconsumptie en de vraag naar biobrandstoffen.

prijs van landbouwproducten: door de stijgende prijzen van landbouwproducten is er meer vraag naar machines die aan hogere efficiëntie (lees, met minder verliezen) werken.

Commerciële context Eén van dè verkoopsargumenten voor geavanceerde landbouwmachines is echter de ontlasting van de operator. Door taken over te nemen van de operator zal de performantie immers min of meer constant blijven gedurende de dag. Daarnaast maken meer geavanceerde machines het ook mogelijk dat de machine efficiënt bestuurd wordt door operatoren met weinig ervaring.

III.2 Maaidorsers

Werkingsprincipes Het gewas wordt verzameld (en meestal tegelijkertijd ook gemaaid) door de header. Er zijn verschillende types van header naargelang gewas en oogstprincipe. Het graan wordt naar het dorsmechanisme gevoerd via een vijzel en een transportband. Het dorsmechanisme dorst dan het graan, zodat de graankorrels los

komen te zitten. Vervolgens gebeurt er een afscheiding (separatie) tussen het gedorste graan en het stro. Het stro verlaat achteraan de machine en het gedorste graan komt op de reiniging terecht. Deze reining gaat de korrels (het eigenlijke graan) scheiden van het kaf en ander vuil (zoals gebroken stro). De korrels worden dan naar de graantank gevoerd. Het stro en het MOG verlaat de machine achteraan.

Sensoren Op deze machine zijn er heel wat sensoren beschikbaar. De belangrijkste hiervan zijn de feed rate sensor, de graanmassdebiet sensor en de verliessensor. Deze sensoren zullen later gebruikt worden om de machine te controleren.

Road map De uiteindelijke doelstelling van de producenten van landbouwmachines is een volledig autonome machine te bouwen. Vooral in landen met heel grote velden, zoals Frankrijk en Duitsland, is dit interessant. Het zou het immers mogelijk maken om zonder operatoren meerdere machines een veld te laten bewerken op de meest efficiënte manier, dus ook rekening houdend met het afvoeren van het graan. De volgende stap op deze road map is een capaciteitscontrolesysteem. Dit is dan ook de doelstelling van de applicatie in dit doctoraat.

IV Onderzoeksdoelstellingen

De hoofddoelstelling van dit werk kan opgesplitst worden in de volgende onderzoeksdoelstellingen:

Onderzoeksdoelstelling IV.1. *Ontwerp een tijdsvariabel procesmodel dat variaties in het proces zodanig kan volgen dat de parameterfout normaal verdeeld is.*

Onderzoeksdoelstelling IV.2. *Ontwikkel een capaciteitscontrolesysteem met behulp van MPC dat feed rate (gewasinstroom), motorbelasting en graanmassadebiet kan regelen naar de gevraagde instelwaarde, zonder hierbij de beperkingen op ingangen of toestanden te overschrijden.*

Onderzoeksdoelstelling IV.3. *Voorspel online de onzekerheid op de toestanden die veroorzaakt wordt door veranderingen in de procesingang. Neem deze onzekerheid op in de regelaar, gebruik makende van het Stochastische MPC framework zodat deze regelaar altijd even performant reageert als de modelkwaliteit toelaat.*

Onderzoeksdoelstelling IV.4. *Implementeer Stochastische MPC op een real-life machine, en houd hierbij rekening met de bestaande hardware- en software-beperkingen. Het moet mogelijk zijn deze implementatie eenvoudig over te dragen naar de CCM's (processing hardware van de maaidorsers).*

Onderzoeksdoelstelling IV.5. *Toon aan dat de methodologie ontwikkeld in deze research, en dan vooral de Stochastische MPC-aanpak, ook werkt voor andere applicaties dan capaciteitscontrole op de maaidorser.*

V Experimentele set-up van de maaidorser

Voor de experimenten in dit doctoraat wordt er gewerkt met een New Holland CR960 maaidorser (W, New Holland Benelux (2008)). Dit is één van de meest krachtige machines die op dit moment op de markt zijn.

Om de experimenten uit te voeren wordt er gewerkt met een industriële PC van National Instruments (W, National Instruments (2008)), die connecteert met de CAN-bus van de machine. De producent heeft ook enkele wijzigingen aangebracht aan de machine-software om het mogelijk te maken de snelheid te controleren via een externe computer.

VI Systeemmodelering

Relaties Op basis van de fysische kennis van het systeem is duidelijk welke relaties er aanwezig zijn tussen de verschillende procesvariabelen. Feed rate hangt zo bijvoorbeeld af van de rijsnelheid. De statische versterkingsfactor tussen feed rate en rijsnelheid zal afhankelijk zijn van de gewasdichtheid.

Systeemidentificatie Elke relatie wordt onderzocht op dynamica. Indien er significante dynamica aanwezig blijkt te zijn, wordt deze gemodeleerd, gebruik makende van tijdsdomein-systeemidentificatie. Er is zo bijvoorbeeld dynamica aanwezig tussen feed rate en graanmassdebiet.

Variatie van de vertraging Tijdens het opstellen van de dynamische modellen is de optimale tijdsvertraging bepaald voor die dataset en die modelstructuur. Het is echter ook interessant om te onderzoeken hoe sterk deze tijdsvertragingen variëren, en of ze afhangen van het gewas. Er is bijvoorbeeld een groot verschil tussen tarwe en maïs voor de tijdsvertraging tussen de snelheid en de feed rate. Deze vertraging hangt immers af van de header, en de maïs-header lijkt in niets op die van tarwe.

Invloeden van verstoringen De invloed van een aantal niet-gemodeleerde variabelen, zoals de helling van het veld wordt eveneens onderzocht. Het is namelijk belangrijk om te weten hoe betrouwbaar het model is, en hoe significant de niet-gemodeleerde effecten zijn. Zo blijkt bijvoorbeeld dat de verliezen veel hoger zijn als de machine bergop rijdt dan wanneer ze bergaf rijdt, hetgeen al verwacht werd op basis van fysische overwegingen.

Deze niet-gemodeleerde effecten zijn dan ook meteen de motivatie om het model tijdsvariabel te maken. Op deze manier kunnen immers heel wat van deze effecten gecompenseerd worden.

Tijdsvariabel model Een belangrijk aspect van het tijdsvariabel model is de distributie van de parameterfout. Deze wordt immers gebruikt in Stochastische MPC, en is bij voorkeur normaal verdeeld. Twee types schatter worden vergeleken, namelijk Sliding Mode en Recursive Least Squares. Met een kleine aanpassing om de snelheid van de schatter aan te passen aan de modelfout, blijkt Recursive Least Squares (of

eigenlijk Fuzzy Recursive Least Squares) een normaal verdeelde modelfout te kunnen leveren. Deze techniek wordt dan ook gebruikt in het uiteindelijke model.

VII Modelgebaseerde Predictieve Controle

Doelstelling De snelheid dient zo geregeld te worden dat de capaciteitsbenutting van de machine constant is, en gelijk aan het gewenste niveau. De capaciteitsbenutting kan op vele manieren worden uitgedrukt, zoals feed rate of motorbelasting. De regelaar maakt een trade-off tussen de controlefout en de regeleffort. Aangezien er een operator op de machine zit, moet er immers rekening gehouden worden met het comfort van die operator, en mag de machine dus niet te agressief accelereren of afremmen. Daarnaast dient de regelaar een aantal beperkingen te respecteren. Sommige van deze beperkingen zijn eveneens gemotiveerd vanuit operatorcomfort, andere dienen om de specificaties van de machine te bewaken, zoals de machinebelasting.

Configuratie In dit werk worden de invloeden van de configuratieparameters van MPC verklaard en geïllustreerd. De belangrijkste parameters zijn de predictie- en controlehorizon, en de gewichtsfactoren in de doelfunctie. Het effect van het adaptieve model wordt eveneens bestudeerd. Een fout op de modelparameters kan geïnterpreteerd worden als een vorm van regularisatie.

Beperkingen Veldexperimenten duiden aan dat de regelaar onstabiel wordt in de buurt van de beperkingen, tenzij de regelaarrespons zeer traag wordt gemaakt. Dit is een gevolg van de verregaande (en tijdsvariabele) modelonzekerheid. Dit probleem zal verholpen worden door gebruik te maken van Stochastische MPC.

Benchmark De feed rate regelaar wordt vergeleken met een eerder ontwikkelde PID en fuzzy regelaar. De MPC-regelaar scoort op alle performantiecriteria beter dan de andere regelaars. De regeleffort, hetgeen een maat is voor het comfortgevoel van de operator, blijft ook zeer laag.

VIII Stochastische MPC

Stochastische MPC is een vorm van robuuste MPC die procesvariabelen beschouwt als stochastische variabelen. In dit werk gaan we er van uit dat de onzekerheid op de toestanden wordt veroorzaakt door de onzekerheid op de modelparameters. Deze parameters zijn immers tijdsvariabele en worden online geschat. Daarnaast worden ook nog de volgende veronderstellingen gemaakt:

- de onzekerheid op de toestanden wordt enkel veroorzaakt door ingangswijzigingen. Indien de ingang constant blijft is de onzekerheid dus 0.
- de parameterfout wordt verondersteld normaal verdeeld te zijn.
- de parameterfout van de verschillende parameters is statistisch onafhankelijk.

- de onzekere parameters zijn enkel aanwezig in een voorwaarts pad. Hiermee wordt bedoeld dat het niet voorkomt dat een toestand vermenigvuldigd wordt met een onzekere parameter die de toestand (mede) bepaalt.

- de toestandsvergelijkingen zijn lineair in de onzekere parameters.

Stochastische MPC betekent dat de doelfunctie de verwachte waarde is van een functie van toestanden en ingangen, en dat de beperkingen enkel met een gegeven waarschijnlijkheid dienen voldaan te worden. Uitgaande van de bovenstaande veronderstellingen wordt de doelfunctie herwerkt zodanig dat de verwachte waarde-operator verwijderd wordt. Deze bemoeilijkt immers de optimisatie.

In eerste instantie wordt de situatie bekeken waar de onzekere parameters enkel voorkomen in de ingangsmatrix. Voor deze situatie kan de doelfunctie herschreven worden in functie van de standaarddeviatie van de onzekere modelparameters:

$$V\left(\bar{\mathbf{x}}_1, \ldots, \bar{\mathbf{x}}_N, \mathbf{u}_1, \ldots, \mathbf{u}_{N-1}\right) =$$

$$\sum_{k=1}^{N} \bar{\mathbf{x}}_k^T \bar{\mathbf{x}}_k + \sum_{j=1}^{r} \sigma_{K_j}^2 \sum_{k=1}^{N} \left(\sum_{i=0}^{k-1} \mathbf{A}^{k-i-1} \mathbf{B}_j \mathbf{u}_i \right)^T \left(\sum_{i=0}^{k-1} \mathbf{A}^{k-i-1} \mathbf{B}_j \mathbf{u}_i \right) \quad (4)$$

Dit betekent dat de doelfunctie nu bestaat uit een zogenaamde bias- en variantie term. De bias term is de controlefout, en de variantieterm is de toestandsonzekerheid. Doordat de toestandsonzekerheid verondersteld wordt veroorzaakt te zijn door ingangswijzigingen, is de variantie term functie van de ingangsvector. Dit betekent dat deze term kan gezien worden als een vorm van regularisatie. Bovendien hangt deze term af van de onzekerheid op de modelparameters, die tijdsvariabel is. Stochastische MPC is met een andere woorden een methode om de doelfunctie van een MPC-regelaar online bij te sturen op basis van de modelkwaliteit.

De situatie waar er ook onzekere parameters voorkomen in de toestandsmatrix is geen eenvoudige uitbreiding van het bovenstaande geval. In dit werk wordt er gekozen voor een pragmatische aanpak die resulteert in een benaderende oplossing.

Door de beperkingen om te vormen tot beperkingen die slechts met een bepaalde waarschijnlijkheid dienen voldaan te zijn, wordt de regelaar stabieler in de buurt van de beperkingen zonder dat de regelaar conservatief moet ingesteld worden voor het hele werkingsgebied. Deze aanpak maakte het mogelijk om beperkingen toe te voegen aan de capaciteitscontrole op de maaidorser. Met een standaard implementatie van de beperkingen was het niet mogelijk te een snelle respons te hebben en toch stabiel te blijven in de buurt van beperkingen.

IX MPC implementatie

Kwadratisch programmeren In dit werk zijn alle MPC-problemen van het type QP, en kunnen dus met technieken van kwadratisch programmeren worden aangepakt. Voor de implementatie is het belangrijk te kunnen beschikken over een methode die computationeel performant is, maar toch geen al te hoge implementatiecomplexiteit heeft. De methode moet immers met beperkte inspanning kunnen geïmplementeerd

worden op verschillende systemen. Aan het begin van dit onderzoek was er voor het gebruikte implementatieplatform enkel een IPM (Interior-Point Methode) beschikbaar, die bovendien te traag was. Een logisch startpunt om zelf een methode te implementeren, zijn de bestaande technieken waarvan ASM's (Active Set Methodes) en IPM's de belangrijkste zijn.

Implementatie Active Set Methode Eén van de voordelen van ASM's is dat ze steeds een oplossing opleveren. Aangezien er in elke iteratie een tussenoplossing wordt gegenereerd, kan er steeds een resultaat gegeven worden als de beschikbare tijd opgebruikt is. De optimisatiemethode gebruikt in dit werk is dan ook een ASM. Er zijn wel een aantal aanpassingen gebeurt om de implementatie te vereenvoudigen. Het voornaamste verschil met een standaard ASM is dat deze ASM vertrekt vanuit een grafische interpretatie van het probleem. Dat maakt het mogelijk om het algoritme met beperkte inspanning te implementeren in een omgeving waar er geen gespecialiseerde bibliotheken beschikbaar zijn.

Benchmark De methode wordt vergeleken met een IPM die commercieel beschikbaar is voor het implementatieplatform. De vergelijking gebeurd voor een eenvoudig probleem, en voor een situatie die voorkomt in de applicatie van Hoofdstuk 10. De eigen ontwikkelde ASM blijkt significant sneller te zijn dan de commercieel beschikbare methode.

X Applicatie: cruisecontrol

Doelstelling Zoals eerder vemeld, worden er steeds strengere normen opgelegd door de Europese Commissie, onder andere voor de uitstoot en geluidsproductie van zware machines. Het aandrijfsysteem van de gebruikte maaidorser is een hydraulisch systeem (met een variabele instelling van de pomp) dat is aangedreven door een dieselmotor. Bij het rijden op de weg kan het toerental van deze dieselmotor eveneens worden aangepast. Er zijn met andere woorden twee ingangsvariabelen om de snelheid te sturen.

Om de uitstoot en het geluidsniveau van de machine te beperken, is het interessant dat het toerental van de dieselmotor zo laag mogelijk wordt gehouden. Enkel tijdens acceleratie mag het toerental verhoogd worden om de acceleratie-performantie van de machine op hetzelfde niveau te houden.

De doelstelling is dan ook een regelaar te ontwikkelen die het toerental zo laag mogelijk houdt, zonder de performantie op vlak van acceleratie te schaden.

Niet-lineaire MPC Na een uitgebreide meetcampagne, wordt er een statisch en een dynamisch model van het aandrijfsysteem opgesteld. Dit model is essentieel om een MPC-regelaar te ontwikkelen. Aangezien het systeem niet-lineair is, dient het model in elke iteratie opnieuw gelineariseerd te worden.

De resultaten van deze regelaar zijn zeer bevredigend. Het gemiddelde toerental van de motor wordt verlaagd, zonder dat de operator het gevoel heeft dat de machine trager accelereert.

XI Applicatie: balendensiteitsregeling

Doelstelling De densiteit van balen hangt af van de druk op de perskamer van de balenpers, en van de lokale gewasomstandigheden. Op een balenpers kan de druk op het achterste deel van de zijwand van de pers ingesteld worden. De kracht op de plunjer tijdens het persen wordt verondersteld evenredig te zijn met de densiteit van de baal. Dit is uiteraard geen absolute densiteitsmeting, maar als relatieve meting (meer of minder densiteit) werkt dit vrij behoorlijk. De doelstelling is om de druk op de perskamer zodanig te regelen dat de kracht op de plunjer gelijk blijft aan de ingestelde waarde. Eén van de uitdagingen bij deze applicatie is het grote verschil in dynamica. De kracht op de plunjer varieert zeer traag, en kan enkel gemeten worden bij een actieve plunjerslag. De stroom naar de hydraulische pomp varieert uiteraard zeer snel, en wordt gesampled aan 25 Hz. De druk ligt qua reactiesnelheid daar tussenin, en wordt gesampled aan 5 Hz.

Stochastische MPC Dit is een zeer interessante applicatie voor Stochastische MPC. Het is een compleet andere machine dan de maaidorser, maar omdat de biologische variabiliteit een belangrijke rol speelt, is het ook hier interessant om de modelonzekerheid in rekening te brengen. Door de weersomstandigheden was het niet mogelijk dit algoritme uitgebreid te testen. De eerste resultaten zijn echter veelbelovend.

XII Besluit

XII.1 Conclusies

Stochastische MPC De doelstellingen opgelijst in sectie IV zijn allen behaald. Vooreerst is er een tijdsvariabel model opgesteld op basis waarvan een capaciteitsregeling voor de maaidorser is ontwikkeld met MPC.

Er is eveneens een model opgesteld voor de modelonzekerheid. Aangezien de onzekerheid veroorzaakt bij ingangswijzigingen ten gevolge van parameterfouten de belangrijkste is, wordt enkel met dit type van onzekerheid rekening gehouden. Stochastische MPC maakt het mogelijk om de snelheid van de regelaar online af te stemmen op de instantane modelkwaliteit. Dit is vooral belangrijk voor stabiele operatie in de buurt van de beperkingen.

Om de regelaar vlot te kunnen implementeren is er een grafische interpretatie opgesteld van een Active Set Methode. Dit maakt het mogelijk om de optimisatie zeer snel uit te voeren, en dit met een beperkte implementatiecomplexiteit.

Tenslotte zijn er aantal applicaties opgenomen ter validatie van de besproken methodologie. Vooral voor Stochastische MPC is het belangrijk aan te tonen dat de methologie ook op andere applicaties kan toegepast worden.

Applicaties Dit onderzoek heeft geresulteerd in aardig wat applicaties. De hoofdapplicatie is uiteraard capaciteitsregeling. Deze regelaar heeft aangetoond zeer goed te werken in verschillende veldomstandigheden, en in staat te zijn om te gaan met de variabele modelonzekerheid. Stochastische MPC bleek noodzakelijk om een performante regelaar te ontwikkelen die toch stabiel blijft in de buurt van beperkingen.

Daarnaast is 'IntelliCruise', de commerciële capaciteitsregeling van CNH, gebaseerd op dit onderzoek.

MPC is ook toegepast op andere applicaties, zoals bijvoorbeeld cruisecontrol voor de maaidorser. Dit is een niet-lineaire applicatie waarbij de snelheid van de machine geregeld wordt door de instelling van de hydraulische pomp en het toerental van de dieselmotor te variëren. Experimenten hebben aangetoond dat deze regelaar het toerental kan verlagen tot een minimale waarde, zonder verlies van acceleratieperformantie.

Balendensiteitsregeling is tenslotte een validatie van de Stochastische MPC-methodologie. Deze applicatie is ook sterk onderhevig aan de veranderende eigenschappen van het inkomende gewas. Veldexperimenten tonen ook voor deze applicatie zeer belovende resultaten.

XII.2 Toekomstig onderzoek

Enkele interessante pistes voor toekomstig onderzoek zijn:

- een meer algemene aanpak van de modelonzekerheid.
- incorporatie van het feedback effect. Hoewel MPC open loop optimisatie is, is er toch sprake van een feedback doordat de procedure elke tijdsstap herhaald wordt.
- het kan interessant zijn de verschillende tijdsvertragingen online te bepalen.
- om objectief de voordelen van capaciteitscontrole vast te stellen is er een uitgebreide benchmark nodig op grote velden.
- de balendensiteitsregeling dient nog verder verfijnd te worden.

Glossary

Notation

$\mathbf{A}$	denotes a matrix
$\mathbf{a}$	denotes a vector
a	denotes a scalar
$\hat{x}$	the estimated value of x
$\bar{x}$	the expected value of x
$\tilde{x}$	the error term of x: $x = \bar{x} + \tilde{x}$
σ_K	standard deviation of parameter K
$\Gamma(\cdot)$	gamma distribution function
$\mathcal{N}\left(\mu, \sigma^2\right)$	normally distributed with mean μ and standard deviation σ
$\mathfrak{N}(\cdot)$	the cumulative normal distribution function: $\Pr\{Z \leq z\} = \mathfrak{N}(z)$ for $Z \propto \mathcal{N}(0,1)$
$\Pr\{\cdot\}$	the probability of the event (between 0 and 1)

Symbols

Δ	current variable value minus previous value
E_C	overall control error
E_L	low-frequency control error
E_H	high-frequency control error
I_C	overall input effort
I_H	high-frequency input effort
I_L	low-frequency input effort
N_C	control horizon
N_P	prediction horizon
p	number of inputs
q	number of states
r	number of uncertain model parameters
s	number of constraints
v	number of outputs

System variables

More information about the meaning of these variables can be found in Chapter 3, Chapter 6, Chapter 10 and Chapter 11.

Capacity control on the combine harvester (Chapter 7 & 8)

PS	pump setting
HF	hydrostatic flow
GS	ground speed
FR	feed rate
EL	engine load
TD	torque of the threshing drum
GMF	grain mass flow
RF	return flow
SL	sieve losses

Cruise control on the combine harvester (Chapter 10)

PS	pump setting
GS	ground speed
ES	engine speed of the diesel engine

Density control on the square baler (Chapter 11)

V_P	PWM signal to the pressure valve
I_P	current to the pressure valve
P_B	pressure on the bale chamber
F_B	force on the plunger

Acronyms

ASM	Active Set Method
CAN	Controller Area Network
ECU	Engine Control Unit
Fuzzy RLS	Fuzzy Recursive Least Squares
IPM	Interior-Point Method
LMI	Linear Matrix Inequality
LP	Linear Programming
MOG	Material Other than Grain
MPC	Model-based Predictive Control
PWM	Pulse Width Modulation
QP	Quadratic Program(ming)
RLS	Recursive Least Squares
SM	Sliding Mode
SOCP	Second Order Cone Program
Stochastic MPC	Stochastic Model-based Predictive Control

Agricultural & combine harvester terminology

capacity	amount of tonnes of clean grain the combine can process per hour without exceeding a 1% loss level
engine load	fuel consumption expressed relative to the fuel consumption at maximal power
feed rate	the amount of crop that is entering the combine at the header
grain bin	the tank on top of the combine where the threshed and cleaned grain is stored. This bin can contain up to 10500 liters (about 8 tonnes)
machine load	tonnes per hour of clean grain the combine produces relative to the combine capacity (% capacity utilization)
PTO	Power Take-Off

Definitions

input effort	the RMS (Root Mean Squared) value of the input change, which is a measure of the cost to control the process.
control error	the RMS value of the error between the controlled variable and the set point.
control rate	the number of values per second that are outputted by the controller.
model sampling rate	the time resolution used in the state space model of the system (which is used by the MPC).
model update rate	the frequency at which the model used by the MPC is modified. Due to the used solving method (Chapter 9) this is very computationally expensive. This model update rate has no effect at the actual parameter values, it only influences how long it takes before new parameter values are used by the controller.
parameter update rate	the frequency at which the online estimated gains are updated. This is a measure for the online filter action.

Reference symbols

Following symbols in front of a reference indicate that the reference does not belong to the main bibliography.

W	Online references, the online reference list can be found from page 199 onwards
F	Field test references, the field test reference list can be found from page 245 onwards

Contents

Voorwoord **i**

Korte Inhoud **iii**

Abstract **v**

Nederlandstalige samenvatting **vii**
I Inleiding . . . vii
II Geschiedenis en ontwikkeling van MPC . . . vii
III Landbouwmachines . . . ix
III.1 Motivatie . . . ix
III.2 Maaidorsers . . . x
IV Onderzoeksdoelstellingen . . . xi
V Experimentele set-up van de maaidorser . . . xii
VI Systeemmodelering . . . xii
VII Modelgebaseerde Predictieve Controle . . . xiii
VIII Stochastische MPC . . . xiii
IX MPC implementatie . . . xiv
X Applicatie: cruisecontrol . . . xv
XI Applicatie: balendensiteitsregeling . . . xvi
XII Besluit . . . xvi
XII.1 Conclusies . . . xvi
XII.2 Toekomstig onderzoek . . . xvii

Glossary **xix**

Contents **xxiii**

List of Figures **xxix**

1 Introduction **1**
1.1 Problem description . . . 1
1.2 Main objective . . . 2
1.3 Motivation . . . 3
1.3.1 Control of uncertain systems . . . 3
1.3.2 Model-based Predictive Control . . . 4

1.3.3 Agricultural machinery 4
1.4 Chapter by chapter overview 5
1.5 Structure of the thesis 8
1.6 Contributions by this research 9

2 History and development of MPC **11**
2.1 Early applications of control 12
2.2 Model-based Predictive Control 14
2.2.1 Historical perspective 14
2.2.2 Definition of MPC 15
2.2.3 Benefits of MPC 19
2.2.4 Feasibility 19
2.2.5 Stability 21
2.2.6 Nonlinear MPC 22
2.2.7 Applications 22
2.3 Model uncertainty 23
2.3.1 Uncertainty description 23
2.3.2 Robust MPC 24
2.3.3 Stochastic MPC 25
2.3.4 Multiple operating states 26
2.4 Conclusion 26

3 Agricultural machinery **27**
3.1 Automation in agriculture 27
3.1.1 Historical context 27
3.1.2 Environmental & social context 28
3.1.3 Economical context 29
3.1.4 Commercial context 32
3.2 Combine harvesters 33
3.2.1 Working principle of combines 33
3.2.2 Sensors for system states 37
3.2.3 Recent developments 42
3.2.4 Road map for combine automation 43
3.3 Conclusion 45

4 Research objectives **47**
4.1 System model 48
4.2 Capacity control 48
4.2.1 Control objectives 48
4.2.2 Operation constraints 49
4.2.3 Benefits 49
4.2.4 Assessment 49
4.3 Model and process uncertainty 50
4.4 Implementation 51
4.5 Validation of the methodology 51
4.6 Conclusion 51

5 Experimental set-up on the combine harvester 53
5.1 Combine harvester . . . 53
5.2 Sensor overview . . . 54
5.3 Measurement set-up . . . 55
5.4 Field tests . . . 56
5.5 Conclusion . . . 56

6 System modeling 57
6.1 Process structure . . . 57
6.2 System identification . . . 59
6.3 Delay variation . . . 62
6.4 Influence of disturbances . . . 65
6.4.1 Effect of threshing speed and slope on the transport delay . . . 65
6.4.2 Effect of slope on losses . . . 68
6.4.3 Effect of slope and grain bin mass on the propulsion system . 71
6.4.4 Effect of the machine load on the grain mass flow dynamics . 72
6.4.5 Conclusion . . . 76
6.5 Time-varying model . . . 78
6.6 Model for capacity control on the combine harvester . . . 86
6.7 Conclusion . . . 89

7 Model-based Predictive Control 91
7.1 Controller specification . . . 91
7.1.1 Variables . . . 92
7.1.2 Control freedom . . . 92
7.1.3 Model . . . 93
7.1.4 Constraints . . . 93
7.2 Objective function configuration . . . 94
7.2.1 Horizon effect . . . 94
7.2.2 Objective function weight factors . . . 96
7.2.3 Error dependent objective function . . . 97
7.2.4 Control rate . . . 99
7.3 Model configuration . . . 100
7.3.1 Model sampling rate . . . 100
7.3.2 Effect of dynamical models . . . 102
7.3.3 Delays . . . 102
7.3.4 Importance of the adaptive model . . . 104
7.3.5 Model update rate . . . 105
7.4 Constraints . . . 106
7.4.1 Constraint definition . . . 106
7.4.2 Feasibility . . . 106
7.5 Benchmark . . . 108
7.5.1 Basic control . . . 108
7.5.2 MPC . . . 111
7.6 Conclusions . . . 112

8 Stochastic MPC **113**
8.1 State of the art . . . 113
8.2 New method . . . 117
8.2.1 Assumptions . . . 117
8.2.2 System definition . . . 118
8.2.3 Uncertain parameter in $\mathbf{B}$ . . . 119
8.2.4 Uncertain parameters in $\mathbf{A}$ and $\mathbf{B}$ . . . 123
8.2.5 Probabilistic constraints . . . 123
8.3 Analysis . . . 124
8.4 Parameter uncertainty . . . 126
8.5 Simulation results . . . 128
8.6 Field test results . . . 130
8.7 Conclusions . . . 134

9 MPC implementation **137**
9.1 Theory of Quadratic Programming . . . 138
9.1.1 Equality-constrained quadratic programs . . . 139
9.1.2 Inequality-constrained quadratic programs . . . 140
9.2 Explicit MPC . . . 143
9.3 Implementation of an Active Set Method . . . 144
9.3.1 Motivation . . . 144
9.3.2 Eliminate equality constraints . . . 144
9.3.3 Calculate the optimal step . . . 145
9.3.4 Transforming the active set . . . 146
9.3.5 Rotational symmetry . . . 146
9.3.6 Stopping criterion . . . 147
9.3.7 Final remarks . . . 148
9.4 Initialization of an Active Set Method . . . 148
9.4.1 Motivation . . . 148
9.4.2 Slack variables . . . 148
9.4.3 Linear programming . . . 149
9.5 Computation speed . . . 149
9.6 Conclusions . . . 152

10 Application: cruise control **153**
10.1 System description . . . 153
10.2 Objective . . . 154
10.3 System modeling . . . 155
10.4 Nonlinear MPC . . . 161
10.4.1 Objective definition . . . 161
10.4.2 Experimental set-up . . . 163
10.4.3 Results . . . 163
10.5 Conclusions . . . 165

11 Application: bale density control **167**
11.1 System description . . . 167
11.2 Objective . . . 169
11.3 Data exploration . . . 170
11.4 System modeling . . . 171
11.5 Stochastic MPC . . . 173
11.5.1 Control structure . . . 173
11.5.2 Objective definition . . . 174
11.5.3 Parameter uncertainty . . . 176
11.5.4 Results . . . 177
11.6 Conclusions . . . 178

12 Conclusions & future research **179**
12.1 Conclusions . . . 179
12.1.1 Stochastic MPC methodology . . . 179
12.1.2 Applications . . . 182
12.2 Future research . . . 184

Bibliography **187**

Online references **199**

Curriculum Vitae **201**

Publication list **203**

A Measurement platform **207**
A.1 Requirements . . . 207
A.2 Considered hardware/software . . . 207
A.3 Selected hardware . . . 208
A.4 Benchmarking . . . 209
A.5 Conclusion . . . 209

B Software **213**
B.1 Available software at the start . . . 213
B.2 Requirements . . . 214
B.3 Software description . . . 215
B.3.1 Overview . . . 215
B.3.2 Control loop . . . 216
B.3.3 Process loop . . . 216
B.3.4 Interface loop . . . 217
B.3.5 Display loop . . . 218
B.3.6 Control log loop . . . 218
B.3.7 Main log loop . . . 218
B.3.8 Generate control matrices . . . 218
B.4 Conclusion . . . 218

C Computational load of a model update **221**
C.1 Procedure 221
C.2 Computational load 224
C.2.1 Complexity analysis 225
C.2.2 Experimental results 226
C.3 Reduce the matrix sizes 229
C.4 Conclusion 232

D Performance measures **233**
D.1 Transition phenomena 233
D.2 Regime conditions 233
D.2.1 Methodology 234
D.2.2 Input effort 234
D.2.3 Control error 234
D.3 Conclusion 236

E Parameter uncertainty **237**
E.1 Parameter multiplication 237
E.2 Calculate the model uncertainty 239

F System matrices **241**
F.1 Capacity control on a combine harvester 241
F.2 Cruise control on a combine harvester 242
F.3 Density control on a combine harvester 243

G Field test references **245**

Index **249**

Acknowledgements **251**

List of Figures

1 Introduction **1**
1.1 A New Holland CX combine harvester in wheat 5
1.2 Overview of this PhD 6
1.3 Structure of this thesis 8
1.4 Illustration of the working principle of IntelliCruise 10

2 History and development of MPC **11**
2.1 Water clock and self-regulating oven 13
2.2 An example of a fly-ball or centrifugal governor 14
2.3 An illustration of MPC 17

3 Agricultural machinery **27**
3.1 Man and machine productivity in cereal production over the last two centuries 28
3.2 Russian-American grain production compared in 1975 29
3.3 Evolution of the wheat price over the last century 32
3.4 CX combine harvester on tracks in corn 33
3.5 Process diagram of a combine 34
3.6 An axial flow rotary combine utilizing twin rotors 35
3.7 Cleaning system in a combine harvester 37
3.8 The feed rate sensor as mounted on the New Holland CR combine harvester 39
3.9 Grain mass flow sensors 40
3.10 Road map for combine automation 44

4 Research objectives **47**

5 Experimental set-up on the combine harvester **53**
5.1 Sensor overview 54
5.2 Measurement set-up 55

6 System modeling **57**
6.1 Schematic of the proposed process structure 59
6.2 Dynamic models of the individual relations 61
6.3 Calculation of the delay using correlation 63
6.4 Engine load and ground speed during a step response 64
6.5 Boxplot for the loss level (expressed in number of kernels per 200 ms) for four different case: downhill/slow, downhill/fast, uphill/slow and uphill/fast 69
6.6 New Holland CX hillside combine harvesters 71
6.7 Difference in delay between the rising and the falling edge of grain mass flow 73
6.8 The delay between ground speed and grain mass flow as a function of the driving speed 74
6.9 The proposed model for the cleaning section 75
6.10 The effect of limiting the throughput of each sieve element on the simulated grain mass flow 76
6.11 The simulated effect of the machine load on the grain mass flow dynamics 77
6.12 Histogram of the *GS-FR* and the *FR-GMF* amplification over an entire barley field 79
6.13 The qq-plot for the error of different parameter estimation techniques 82
6.14 The histogram of the parameter error for different estimation techniques 85
6.15 Schematic of the capacity control model 87

7 Model-based Predictive Control **91**
7.1 Feed rate control with an error dependent objective function 99
7.2 Feed rate control with two distinct model update rates 101
7.3 Effect of the model parameter on the engine load response 103
7.4 Effect of the model gain on the feed rate response 105
7.5 The performance of a PID, fuzzy and MPC controller on feed rate control 109

8 Stochastic MPC **113**
8.1 The controller output of Stochastic MPC as a function of the relative parameter uncertainty with and without an active constraint 129
8.2 The step response of Stochastic MPC for feed rate control, engine load control and grain mass flow control 132
8.3 The step response of Stochastic MPC for feed rate control with a blocking engine load constraint 133

9 MPC implementation **137**
9.1 The flow chart of an ASM. 142
9.2 The flow chart of the newly designed ASM 144
9.3 Effect of rotational symmetry on the ASM performance 147
9.4 Simple problem used for benchmarking of the optimization methods . 150

10 Application: cruise control **153**
10.1 A scheme of the propulsion system of a combine harvester. 154
10.2 Slope response of pump setting and engine speed 155
10.3 The local amplification between the pump setting-engine speed product and the machine speed as a function of the pump setting 156
10.4 An overview of the model of the propulsion system 157
10.5 The model of the step response of the propulsion system 158
10.6 Schematic of the hydraulic pump 160
10.7 Cruise control with two different NMPC controllers 164
10.8 Effect of an extra integrator state on the ground speed response 165

11 Application: bale density control **167**
11.1 A New Holland big square baler and the density system 168
11.2 The PWM-current and the pressure-current relation on a square baler . 171
11.3 The step response of the density system of a big square baler 172
11.4 Control structure of the bale density control system 174
11.5 Stochastic MPC on the density control system of the square baler . . . 177
11.6 Stochastic MPC with an active constraint on the square baler 178

12 Conclusions & future research **179**

Publication List **203**
12.1 The presentation of the silver medal winner IntelliCruise at Agritechnica 205

A Measurement platform **207**
A.1 A PXI chassis 208
A.2 Test scenarios in LabView Code 210

B Software **213**
B.1 The interface of *CANbus Interface* 214
B.2 Flow chart of the software 216
B.3 The interface of the LabView program for capacity control 217
B.4 The interface of the LabView program used for the implementation of cruise control 219

C Computational load model update **221**
C.1 The time required for a model update as a function of the prediction and the control horizon 225

D Performance measures **233**
D.1 The lowpass and high-pass second order Butterworth filter which is used to split the input signal 235

D.2 The lowpass and high-pass second order Butterworth filter which is used to split the error signal 235

E Parameter uncertainty **237**

F System matrices **241**

G Field test references **245**

Chapter 1

Introduction

Beginning is easy, continuing is hard
-Japanese proverb-

Most of classic control theory deals with linear time-invariant systems (LTI systems). The reason for this is straightforward: LTI systems are easy to model and to control. In real-life however, most systems exhibit some form of nonlinearity, and often the system properties vary in time due to external factors (like weather or input material variability) or internal factors (like engine temperature or transmission temperature).

A way to deal with this reality is to take model uncertainty into account. In recent years several techniques have been designed to deal with model uncertainty. In many cases these techniques are not very practical. Often they are mathematically very involved or simply computationally infeasible, at least within an acceptable time frame. Proof of this is the very limited number of publications of real-life applications using these robust model-based techniques.

1.1 Problem description

This PhD starts from a real-life application, namely capacity control on combine harvesters. Capacity control means that the speed of the combine is adjusted in order to get the combine to run at maximal system capacity. Since there are many ways to express capacity, capacity control also exists in many flavors, such as feed rate, engine load or grain mass flow control. No matter which process variable is chosen to express capacity, the system model always shows considerable delays and nonlinearities. Moreover, since a combine harvester processes biological material, the system characteristics also depend on the biological properties of the crop. Depending on the crop variety, the crop type and its physical properties, the different process parameters of the combine will vary.

Any practical capacity control system also needs to respect a number of constraints. Some are motivated from operator comfort, such as limits on the machine acceleration, others serve to protect the machine by limiting for instance the engine load or the ground speed. Because of the biological variability which renders the

system time-varying, this is quite a challenging application. The main difficulties to control the capacity are:

- maintain the controller performance as high as possible with varying model uncertainty, and work equally well in all common crops such as barley, wheat and corn.
- deal with large process delays.
- drive as close as possible to the constraints.
- develop a controller which can be implemented on hardware that is acceptable for mass production.

1.2 Main objective

This PhD tackles the above described capacity control problem with appropriate advanced control theory. The goal is to bring techniques, which are already available in academia, to the application level. The main objective of this work can be formulated as:

> **Objective.** *To design a practical capacity control system that copes with **model uncertainty** in an **optimal** manner, making use of the **Stochastic Model-based Predictive Control framework** (Stochastic MPC). It has to be possible to implement the controller on standard processing **hardware** available on combine harvesters with an acceptable effort. The methodology has to be **validated** on another application.*

Model uncertainty can have several causes. The process is time-varying, which implies that the model parameters are never exact. There are disturbances which have a significant impact on the process. Moreover, the selected application also contains nonlinearities. All these elements together cause the model to contain a considerable stochastic component.

The optimality has to be seen from a performance point of view. The main controller objective is to make the system response as fast as possible in all circumstances. This is only possible if the controller tuning is adjusted online, depending on the model uncertainty.

Stochastic MPC is selected as an advanced control method, because it can take the model uncertainty into account to calculate the optimal control input. Stochastic MPC is a fairly new control technique with little or no real-life applications. The choice for Stochastic MPC is motivated by the important role of stochastic model information for this type of processes.

Hardware also plays an important role. Eventually the control structure needs to be implemented on commercial applications. It is thus important that the optimization routines can be easily coded on dedicated processors.

Validation on another application helps to assess the general applicability of the Stochastic MPC control methodology.

1.3 Motivation

In this section the main objective is motivated. Three important aspects can be discerned:

- the focus lies on systems for which the model is uncertain.
- MPC is chosen as control scheme.
- the applications are situated in the field of agricultural machinery.

1.3.1 Control of uncertain systems

Model errors The model of these (nonlinear) time-varying systems often contains a considerable uncertainty component. This uncertainty depends on the model error, which can have many different origins:

linearization error: if nonlinear processes are linearized (to be able to apply linear control theory), this leads to a linearization error. It causes model uncertainty, especially during set point changes.

variable feed: if the properties of the material entering the process vary, this will lead to model uncertainty.

environmental variability: external variables such as temperature, humidity or terrain conditions may also cause time-variability, and thus model uncertainty.

process variability: changes in the process that are not caused by any of the above causes, can be categorized as process variability. An example here is the wear of a belt.

Robustness All systems that depend on environmental conditions or that process some material are in practice time-varying. In recent years a lot of research has been performed in academia about these types of systems. Most of the time the system uncertainty is quantified using inequalities, and the controller parameters are determined such that system stability is guaranteed.

The problem of this approach is that the calculation of the controller is so involved, that it is in most cases impossible to perform the calculation online. In other words, this leads to very conservative controllers, since the controller is designed in such a way that it is stable for all possible system conditions. In this PhD a different approach is taken, based on the relatively new methodology of Stochastic MPC.

Examples Uncertainty has been a major issue in control theory since the very beginning. In the real-life examples below the importance of time-varying and nonlinear systems in everyday applications is illustrated.

Example 1.1. *Ball-governor. One of the first known examples of a control system is the ball governor on the Watt steam engine. The governor changed the steam inlet as a function of the engine speed. As the engine turned faster, the balls turned faster around the axle, which caused a larger centrifugal force. This resulted in a correction*

of the inlet valve to bring the engine back to its nominal speed. The ball-governor is a relatively simple example of a feedback loop which is implemented mechanically. It is also an example of an uncertain (time-varying) system. The engine load varies in time, as well as the atmospheric pressure. Both will influence the engine response. The mechanical components also vary from one machine to the next (such as the weight of balls), changing the system parameters. The simple solution for this problem is to tune the controller conservatively. If the controller gain is made sufficiently small (and thus the controller sufficiently slow), the system will remain stable in all possible conditions. □

Example 1.2. *Cruise control. Another commonly known example of a simple control system is cruise control on a car. The controlled system (the car) is time-varying since it depends on external conditions such as wind resistance and slope of the road. Thus, the controller gain is kept small to avoid system instability. This explains why most cruise control systems respond very slowly if they are re-engaged far from their set point. For example, if the cruise control is re-enabled with the set point 70km/h whilst driving at 60km/h, most systems will only accelerate very slowly towards the set point.*

□

1.3.2 Model-based Predictive Control

Model-based Predictive Control (MPC) calculates in every time step the optimal input sequence for the system, given the objective function, the system model and the constraints. The objective function describes the trade-off between set point error and input change. The relative weight on the input change is determined as a function of the model and measurement quality. After all, if the model and the measurements were exact, there would be no reason to penalize the input change, except maybe to guarantee operator comfort.

Since it is very important for time-varying systems to be able to adapt the controller to the current system model, an online adaptive control methodology is needed. With MPC the controller resynthesizes in every time step, rendering it very well suited for this application category.

Some other benefits of MPC for this application type are:

- a model-based approach is perfectly suited for an application where delays are important.
- MPC can easily cope with constraints.
- through the objective function the controller can easily be modified to take the actual process condition into account.
- several control tasks (such as cruise control, feed rate control, engine load control, etc.) can be combined in one controller, reducing the software maintenance required on the controllers.

1.3.3 Agricultural machinery

This section motivates why agricultural machinery, and more specifically a combine harvester (Fig. 1.1) is a good candidate to test this novel control methodology on.



Figure 1.1: A New Holland CX combine harvester in wheat (W, New Holland Benelux (2008)).

Time-varying system A combine harvester is a time-varying system. The process depends on the crop intake, which shows a considerable biological variability. Moreover, the crop properties change over the course of a day. The system also contains quite some nonlinearities. In order to deal with the time-varying behavior and the working point-dependent linearization error, an adaptive model is required.

Economical challenge Due to all the changes in the agricultural sector of the last few years, there is an increasing demand for more efficient, larger machinery. One of the most important aspects is the higher price for agricultural products, which enables larger investments in machinery. Moreover, because of the use of agricultural products for biofuels the fundamental economics of the sector change. The increased demand gives the sector a boost. See also Chapter 3 for more details.

1.4 Chapter by chapter overview

This PhD combines two main application domains, namely Model-based Predictive Control (MPC) and agricultural machinery. Some chapters belong to one of these domains, and others form a bridge between both domains. An overview of the chapters is shown in Fig. 1.2.

First of all, both application domains are discussed in more detail. The origin of MPC and its evolution over the last decades is described in **Chapter 2**. The background of the chosen application, namely a combine harvester, is presented in **Chapter 3**. The state of the art in agricultural automation is discussed, together with the major challenges in the sector and the road map for future developments. Once the

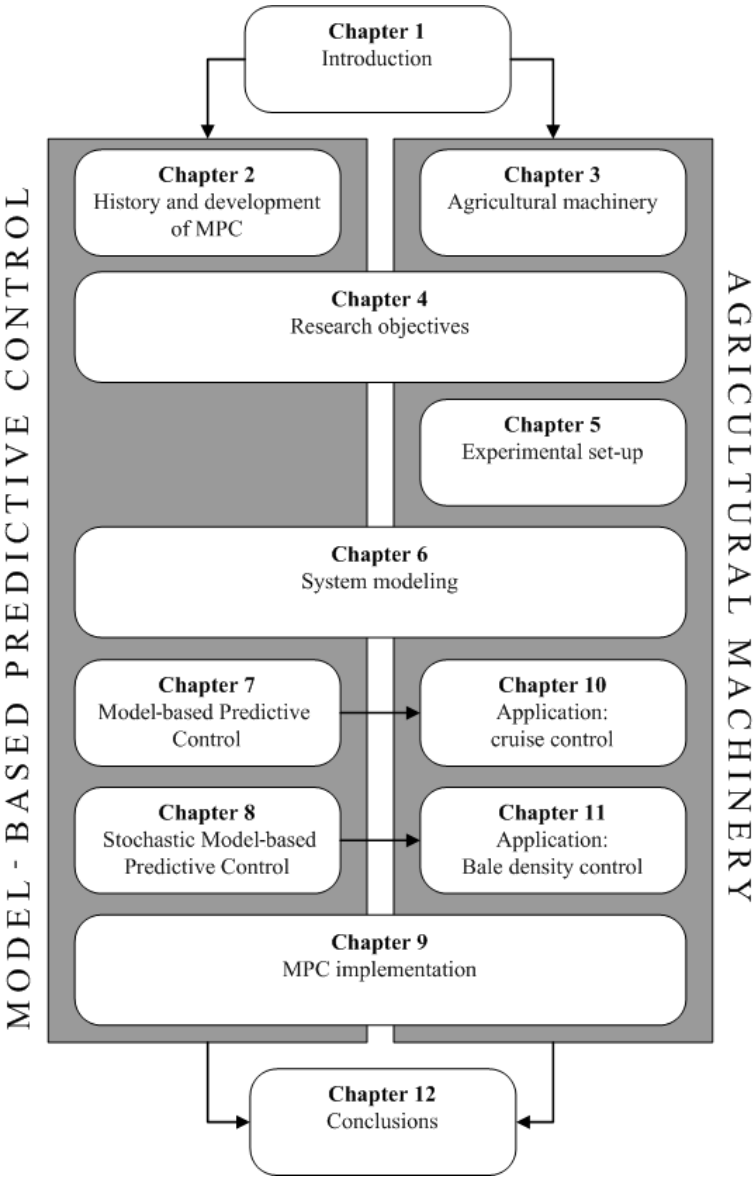


Figure 1.2: Overview of this PhD. The arrows indicate possible reading trajectories outside of the normal chapter flow.

state of the art is known both in the field of MPC and in the field of agricultural machinery, the objectives of this research are made more concrete in **Chapter 4**.

Before one can model the system, some knowledge about the experimental set-up is required. **Chapter 5** gives a detailed account of the available sensors, the measurement set-up and the field test procedure. Controlling an adaptive system is not possible without a good, adaptive system model. **Chapter 6** discusses how the combine harvester process can be modeled to use in MPC. The importance of a couple of disturbances is also illustrated.

Chapter 7 deals with so-called classic MPC. It describes the issues for a practical implementation of MPC, such as horizon length, update rate, etc. Some control specifications, such as respecting constraints, are not reachable with classic MPC because of the model uncertainty. In **Chapter 8** Stochastic MPC is introduced to be able to take the model uncertainty into account. This is one of the first practical applications of this new control methodology. The Stochastic MPC framework is extended for the problem at hand, and the controller performance is illustrated on the capacity control problem.

Implementing MPC requires fast optimization software to perform the calculations online. Due to the limitations of the implementation platform, a novel optimization methodology is introduced in **Chapter 9**. This method is actually a modification of an Active Set Method.

A Nonlinear MPC application, namely cruise control in road conditions on a combine harvester, is discussed in **Chapter 10**. In **Chapter 11** the Stochastic MPC approach of Chapter 8 is validated by implementing a new density control system for a large square baler. This is also a time-varying, nonlinear system. The behavior of the baler is strongly influenced by the crop properties.

The conclusions of this research are presented in **Chapter 12**, both from an application and from a control theory point of view.

The measurement platform, which is used for all the measurements and the control experiments in this PhD, is discussed in **Appendix A**. The investigated alternatives are listed, and the choice for the PXI platform is motivated. The PXI system is also benchmarked with a normal Windows PC.

Software is written for the measurement and control experiments. All the code used for real-time experiments was written in LabView (W, National Instruments (2008)). The structure of the field test program, as well as the motivation for that structure is discussed in **Appendix B**.

Because of the adaptive model, the system matrices need to be updated regularly. A trade-off needs to be made between accuracy (the more model updates, the more accurate the controller) and computational feasibility (each model update requires quite some time). The computational cost of such a model update is calculated in **Appendix C**.

Controllers are compared based on performance measures such as the low-frequency input effort. Details of these measures are presented in **Appendix D**.

In Chapter 8 the parameter uncertainty is used by the controller. The procedure for calculating this parameter uncertainty is described in **Appendix E**.

The model matrices of all the applications discussed in this PhD can be found in **Appendix F**.

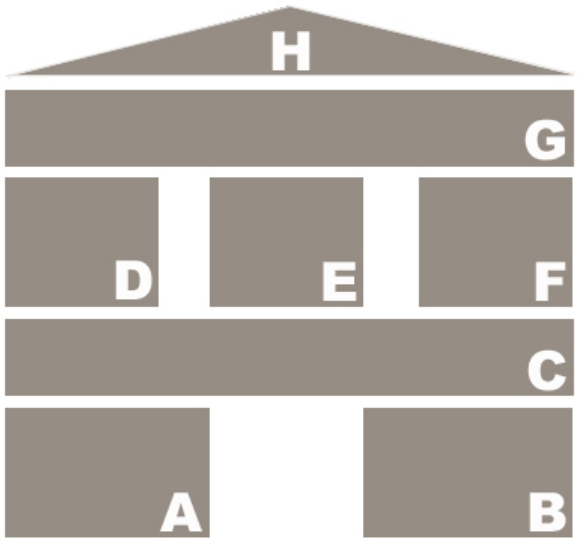


Figure 1.3: Structure of this thesis. A: state of the art in MPC, B: state of the art in agricultural machinery, C: objectives and set-up, D: modeling, E: control, F: optimization methods, G: applications, H: conclusions.

The data files referred to in the text (marked with 'F') are listed in **Appendix G**, together with a description of the performed experiment.

1.5 Structure of the thesis

In Fig. 1.3 the structure of this thesis is illustrated. This picture will be used at the start of each chapter to situate the chapter in the overall structure. As stated before, this thesis is unique in its combination of two very different fields, being MPC and agricultural machinery. The thesis can be split up in the following functional blocks:

- **Foundation**

  A: state of the art in the field of Model-based Predictive Control (Chapter 2).

  B: state of the art in the field of agricultural machinery (Chapter 3).

- **First floor**

  C: the objectives of this thesis (Chapter 4) and the experimental set-up which was used (Chapter 5).

- **Main pillars**

  D: system modeling (Chapter 6).

  E: control theory. Classic MPC (Chapter 7) and Stochastic MPC (Chapter 8).

F: optimization methods (Chapter 9).

- **Second floor**

G: applications. The main application, capacity control on the combine harvester, can be found in Chapters 7 and 8. There also two validation applications, being cruise control on the combine harvester (Chapter 10) and density control on a square baler (Chapter 11).

- **Roof**

H: conclusions (Chapter 12).

1.6 Contributions by this research

Uncertainty model An approach to modeling the state uncertainty is developed. The state uncertainty is assumed to depend purely on the parameter uncertainty. Since the uncertainty originates from the parameter error, it enters the system as multiplicative uncertainty. A realigned model is used to predict the change of output based on the change of input. In the same manner the state uncertainty depends on the input change. In other words, if the input does not change, the state uncertainty is assumed to be zero.

Stochastic MPC In this research some additions are made to Stochastic MPC. An exact state uncertainty model is derived for the case where there are only uncertain parameters in the input matrix. If there are also uncertain parameters in the state matrix, a pragmatic approximation to the uncertainty model is proposed. The new methodology is validated on two real-life applications. This work contains two of the first practical implementations of Stochastic MPC.

Capacity control Using Stochastic MPC a top of the line capacity control system is developed, which can cope with constraints in all circumstances. Moreover, by using an adaptive model for the system and the state uncertainty, the controller performance remains unchanged faced with different crops and conditions.

Implementation A procedure is presented for the implementation of MPC on a real-world application, with focus on time-varying processes. The controller is implemented on a PXI platform (W, National Instruments (2008), Appendix A and Appendix B). A modified optimization method is required to guarantee a swift implementation on the real-time platform. Details of the implemented method are discussed in Chapter 9.

Cruise control A cruise control system is designed to control the travel speed of a combine harvester in road conditions. In road conditions there are two inputs available to control the travel speed, namely the pump setting and the engine speed of the diesel engine. The MPC uses this extra degree of freedom to minimize the engine speed at all times, without losing on acceleration performance. By reducing the engine speed, the noise and the exhausts of the machine are reduced.

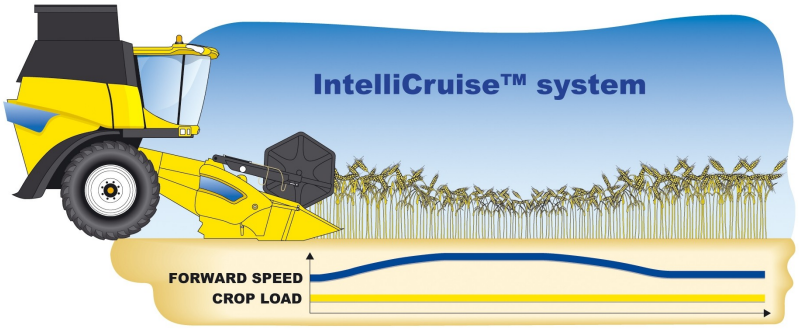


Figure 1.4: Illustration of the working principle of IntelliCruise, a capacity control system designed by the author (W, New Holland Benelux (2008)).

Bale density control The Stochastic MPC methodology is validated as a bale density control system of a large square baler. The objective is to design a controller which can maintain the density of the bales near a given set point while respecting the constraints, and this in all field and crop conditions. The algorithm works on all baler types (different bale chamber dimensions, different hardware, etc.) without modifications. Further field tests are required to validate the benefits of this controller.

IntelliCruise As a result of this research a commercial capacity control system is developed, which was launched on AgriTechnica 2007 (W, Agritechnica (2007)). This is available on CNH combine harvesters under the brand name 'IntelliCruise'. The publicity for 'IntelliCruise' is shown in Fig. 1.4. This figure clearly illustrates the purpose of any capacity control system: if the crop density decreases, the machine accelerates in order to keep the crop load (the so-called feed rate, see Chapter 3) constant.

Chapter 2

History and development of MPC

The past is a work of art,
free of irrelevancies and loose ends
-Sir Max Beerbohm-



A possible definition of a control system is given by Britannica (W, Britannica (2008)):

> A control system is the means by which a variable quantity or set of variable quantities is made to conform to a prescribed norm. It either holds the values of the controlled quantities constant, or causes them to vary in a prescribed way.

People have been interested in control since antiquity. In many ways, one could state that the development of control theory was one of the enabling factors of the industrial revolution in the 19th century. The steam engine for instance would not have been possible without the fly-ball governor.

Control theory serves different goals, depending on the situation. Important driving forces of control, and automation in general, are:

increasing productivity: this is the objective which is most often identified with control engineering. Processes are automated which leads to faster production and lower personnel requirement.

guarding/improving quality: in recent years this application of control engineering is becoming more and more important. One of the factors herein is the rise of visual sensors. This enables quality control for applications which could before only be monitored by people.

operator relief: sometimes control does not lead to increases in efficiency or quality, but just simplifies operator tasks. This lowers the requirements of operators in terms of training and skill level, or just decreases their stress level. An example of such an application to increase the operator comfort is cruise control in a car.

2.1 Early applications of control

Control, in its most general meaning, is as old as mankind itself. Most biological systems contain multiple control systems. For instance, if a person wants to clap his hands, he can do that - even without looking - by making use of the implicit position measurement of his limbs (open loop control). If you want to pick up an object, the feedback loop is closed through a visual measurement of the object position and the position of your hands, which constitutes the feedback. Another example of biological feedback control is the synchronization of fireflies. The male fireflies see each other's flash, and slow down or speed up their own flash in order to synchronize with their neighbors (Strogatz, 2000, p. 103).

Before the discovery of electricity and electronics, feedback loops were implemented mechanically or pneumatically. There are several realizations of feedback control known, dating as far back as antiquity. It was actually the advent of practical control applications that created the need for control theory. The first work that discussed controllers from a more theoretical point of view was the paper 'On governors' by James C. Maxwell in 1868 (Maxwell, 1868).

Up till recently, control technology never existed in its own right. It simply provided a solution for problems occurring in daily life. To put it in economical terms, there was a market-pull, never a market-push. This has changed a bit the last decade. The gap between industry ('real-life') and the academic society has widened, creating very advanced and complicated control theory of which the first practical applications will not be in the near future.

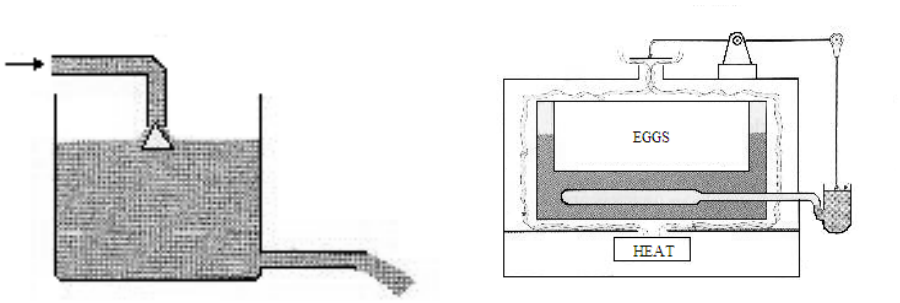
Maybe the finest example of this market-pull of control technology (and technology in general) is the steam engine. The first steam engine was actually built by Hero of Alexandria (Daintith and Gjertsen, 1999) in the first century AD, almost 2 millennia before the steam engine of James Watt. The reason that the steam engine was not brought into practice at that time is the economic reality. In Egypt there was no need for a labor efficiency increase in the first century AD, because of the abundance of cheap slave labor.

In the next paragraphs some of the milestone applications of control theory are discussed (Durham, 2004, Franklin et al., 2006, Bernstein, 2002). More in-depth information can be found in Mayr (1970).

Clocks

One of the greatest human endeavors in antiquity was navigating at sea. In order to determine their position, the first sailors used dead reckoning. They calculated their position based on direction (compass), speed and the starting position. Navigation mistakes had serious consequences, and could easily lead to starvation of the crew. One of the key factors for a good speed measurement was an accurate time measurement. This explains the fascination of mankind with clocks, even in antiquity.

It should thus be no surprise that one of the first known control applications is the water clock. The basics of a good clock is something - in this case water - that moves at a fixed speed. The water clock regulates the flow coming out of the bottom of a bin by controlling the level of the bin. After all the flow depends on the pressure at the bottom of the bin, which is related to the water level. The level control was implemented by introducing a floating plug. The water flows in a cylinder with hour



(a) (b)

Figure 2.1: (a) Float controlled fluid level, as first used in the water clock of Ktesibios. (b) Self-regulating oven design, first conceived by Cornelis (Durham, 2004, Franklin et al., 1994).

markings, of which the hour can be read. The same system was also used by the Romans to regulate water flow in their aqueducts. The concept of the water clock is illustrated in Fig. 2.1 (a). Over the next centuries other timing devices were designed, which eventually led to the mechanical clocks as we know them. The experience of designing regulators for mechanical clocks (the first mechanical clock is reported in the 13th century (Bernstein, 2002)) was very useful to design regulators for other systems as well.

Money

Another fascination of mankind throughout time is money. In the early 1600s a Dutch chemist named Cornelius Drebbel developed an oven that maintained a constant temperature (Fig. 2.1 (b)). The hypothesis was that a base metal would transform into gold if it was kept at a constant high temperature for a very long time. Needless to say that this hypothesis was refuted. The oven however, was a hit as an incubator for chicken eggs. This incubator allowed for a considerable production increase, and was one of the first steps in the automation of food production.

Wind mills

As the population grew, more and more attention was brought onto food production, and the automation possibilities for the sector. An important aspect of food production is the grinding (threshing) of grain. Even in Roman times water mills were used for this task. The largest known mill was located in the Provence (South of France), and was capable of grinding 28 tons of corn per day. This is equivalent to the combined effort of 1 000 slaves. Water mills however, require rivers. In order to bring threshing mills to regions which do not have a large river, other energy sources had to be found. This other energy source turned out to be wind. The use of wind mills flourished. In the 13th century there were already some 120 windmills operating in the vicinity of Ypres, Belgium. Unlike water power, wind power is variable in speed and direction. It would not take long before these issues would be tackled using control technology.

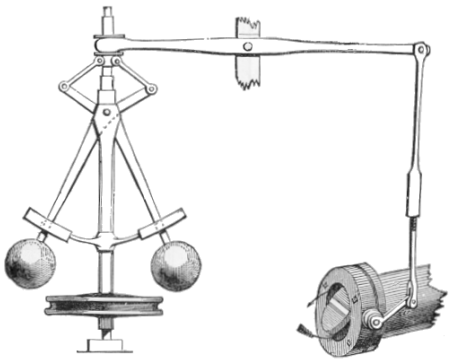


Figure 2.2: An example of a fly-ball or centrifugal governor (from W, Wikipedia (2008a)).

In a British patent application dated 1745, the variability of the wind in both speed and direction was addressed by Edmund Lee, a blacksmith. This included a fan-tail mechanism to direct the mill towards the wind. The wind speed variation caused the gap between the millstones to widen with increasing speed. One of the proposed solutions was the centrifugal governor (Fig. 2.2). Increasing rotation speed would shorten the rotating bar, leading to a feedback action. The same concept would be used by James Watt at the end of the 18th century to control the speed of his steam engine, the new power source for threshing mills.

Feedback amplifier

The next leap forward for control technology came with the feedback amplifier. In 1915 a transcontinental telephone line opened, connecting New York to San Francisco. To compensate for the attenuation over such a long line, amplifiers were installed. Vacuum tubes were the best suited amplifiers of the day, but unfortunately they contained serious nonlinearities. The first telephone had a large cable section, and a limited number of amplifiers. When economic forces (cost) required to reduce the cable section, the greater attenuation because of the smaller section needed to be compensated by placing more amplifiers. Cascading amplifiers like that demanded a whole new amplifier, with a linear amplification. For this purpose the operational amplifier (op-amp) was designed based on feedback theory.

2.2 Model-based Predictive Control

2.2.1 Historical perspective

Origin of MPC The development of MPC (Camacho and Bordons, 2004, Maciejowski, 2002) has followed a somewhat atypical trajectory. As early as 1978, Richalet mentions in the abstract of his paper entitled 'Model predictive heuristic control' (Richalet et al., 1978): 'This method has been continuously and successfully applied to a dozen large scale industrial processes for more than a year's time.'

Although the foundation was laid by academics in the 1960s (Propoi, 1963), MPC did not receive many attention (Morari and Lee, 1998) from the academic society until the 1980s after the publication of the first papers on IDCOM (Identification and Command, a MPC methodology) (Richalet et al., 1978) and Dynamic Matrix Control (Cutler and Ramaker, 1980), and the first comprehensive exposition of Generalized Predictive Control (GPC) (Clarke et al., 1987a,b). In other words, the technique was first implemented in industry before there was serious research into the theory.

This also explains why MPC exists under so many names: IDCOM (Richalet et al., 1978), Dynamic Matrix Control (DMC) (Cutler and Ramaker, 1980), Generalized Predictive Control (GPC) (Clarke et al., 1987a,b), Internal Model Control (IMC) (Garcia and Morari, 1982), Model Algorithmic Control (MAC) (Rouhani and Mehra, 1982), Predictive Functional Control (PFC) (Vivas and Poignet, 2005), etc. All these methodologies have the essential characteristics of MPC in common: an explicit internal model, the receding horizon idea, and computation of the control signal by optimizing predicted plant behavior. However, this does not mean that all these techniques are identical. The ideas underlying the methods are very similar, but the objectives behind the development of GPC and DMC for instance, were very different:

GPC: intended to offer a new adaptive control method. In the tradition of other methods in the field, transfer function models were employed. Stochastic aspects were taken into account from the start, but multivariable control problems were not supported.

DMC: designed to tackle multivariable constrained control problems typical for the oil and chemical industries, where time-domain models are more common.

Essentially all commercial applications of MPC have adopted a DMC-like approach (Qin and Badgwell, 2003), and this road will also be followed in this PhD.

MPC in academia The connection between the closely related minimum time optimal control problem and Linear Programming was first recognized in Zadeh and Whalen (1962). In Propoi (1963) the moving horizon approach which is at the core of all MPC algorithms is proposed. It became known as 'Open Loop Optimal Feedback'.

MPC has been applied to a wide range of applications in academia, such as a simple mixing tank with a heat exchanger (Arkun et al., 1986) or a coupled distillation column system for the separation of a ternary mixture (Levien, 1985, Levien and Morari, 1987). All these applications are multivariable and involve constraints. This is exactly the kind of applications that motivated the development of MPC control techniques.

2.2.2 Definition of MPC

Introduction Since we are interested in multivariable control problems with constraints, we will choose an approach similar to DMC (see above). The purpose of MPC is, as of any optimal control technique, to define the optimal input sequence for a plant. MPC is actually closely related to a LQR (Linear Quadratic Regulator) set-up (Franklin et al., 2006). In LQR an objective function (expressed in the system states,

inputs and/or outputs) is minimized subject to the plant model, over an infinite horizon. This results in an optimal input sequence, which is written as a feedback matrix multiplied by the current process state.

The drawback of LQR is that it is not possible to add constraints (on the inputs, states or outputs), without losing the nice feature that the input can be written analytically as a constant function of the process state. MPC is a similar control problem, but the objective function is only minimized over a given finite prediction and control horizon. Since the horizon is no longer infinite, only a numerical input sequence can be calculated. This solution is however also possible if constraints are added.

Distinguishing elements MPC is a control technique which calculates an optimal input sequence subject to a given *objective function*, *plant model* and *constraints* over a *fixed time horizon*. Most versions of MPC are based on time-domain techniques, which is beneficial for its interpretability by a wider public.

The objective function is a function of the system states, outputs and inputs. The optimal input sequence is calculated in such a way that the objective function is minimized. The objective functions can in theory have any structure, but in practice (and also in this research), the objective function is defined to be a quadratic function. The form of the objective function has a significant impact on the type of solution method, and the solution time required. The general form of the objective function is:

$$V(\mathbf{y}_1, \ldots, \mathbf{y}_{N_P}, \mathbf{x}_1, \ldots, \mathbf{x}_{N_P}, \mathbf{u}_0, \ldots, \mathbf{u}_{N_C-1}) =$$

$$\sum_{k=1}^{N_P} (\mathbf{y}_k - \mathbf{y}_{ref,k})^T \mathbf{S}_k (\mathbf{y}_k - \mathbf{y}_{ref,k}) + \sum_{j=1}^{N_P} (\mathbf{x}_j - \mathbf{x}_{ref,j})^T \mathbf{Q}_j (\mathbf{x}_j - \mathbf{x}_{ref,j}) + \sum_{t=0}^{N_C-1} \mathbf{u}_t^T \mathbf{R}_t \mathbf{u}_t \tag{2.1}$$

with $\mathbf{S}_k$, $\mathbf{Q}_j$ and $\mathbf{R}_t$ weight matrices at respectively time step k, j and t. $\mathbf{y}_k$ is the system output at time step k, $\mathbf{x}_j$ the system state at time step j and $\mathbf{u}_t$ the system input at time step t. $\mathbf{y}_{ref,k}$ is the reference output at time step k and $\mathbf{x}_{ref,j}$ is the reference state at time step j. Depending on the situation, $\mathbf{u}$ may denote the actual system input or the change of the system input. The number of input variables is a measure for the problem complexity, since the input variables are, together with the states, also the optimization variables. The number of input variables depends on the control horizon, which is discussed in the next paragraph.

The prediction horizon N_P denotes the number of time steps for which the output and state sequence is calculated (predicted) and included in the objective function. *The control horizon* N_C denotes the number of time steps over which the input can change. If the control horizon is shorter than the prediction horizon, the input is set to some fixed value for time steps beyond the control horizon and within the prediction horizon. The following statement always holds:

$$N_C \leq N_P \tag{2.2}$$

Note that this does not necessarily mean that the process variables $\mathbf{x}$ and $\mathbf{y}$ do not change anymore after the end of the horizon. This will be discussed below when

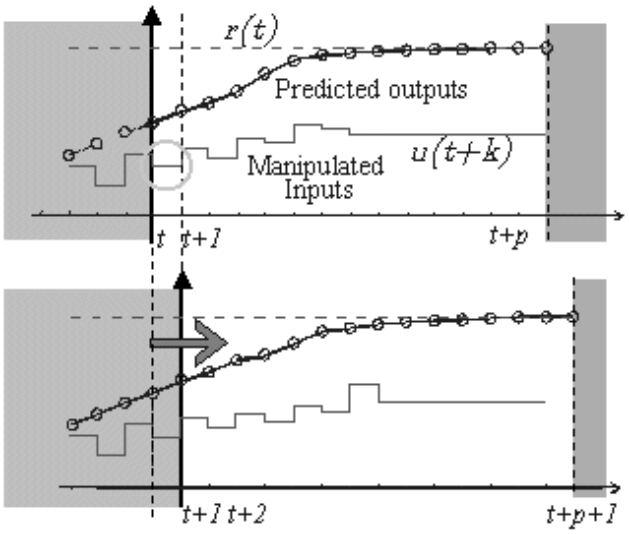


Figure 2.3: An illustration of MPC (from W, Bemporad (2008)).

stability is taken into account. As mentioned above, shortening the control horizon relative to the prediction horizon simplifies the problem, and thus lowers the computational complexity. A shorter control horizon may even improve the numerical condition of the problem. Assume the system contains a delay between the input and the output. If the prediction horizon and the control horizon are equal, this implies that, if the system contains some delays, a number of input variables need to be calculated, which have no impact whatsoever on the output. Needless to say that this is not good for the accuracy of the solution.

The plant model defines the relation between the variables (inputs, states and outputs) in the objective function. The notation of the plant model may differ, for instance impulse or step response models may be used (Maciejowski, 2002). In this research a state-space notation is chosen:

$$P(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k, \mathbf{u}_{k-1}) \quad : \quad \begin{cases} \mathbf{x}_k &= \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k \end{cases} \tag{2.3}$$

This model is an equality constraint on the variables in the objective function for every $k \in \{1, \ldots, N\}$. By setting the direct feed-through from the input to the output to zero, the model is strictly proper.

Constraints are an important part of any real-life control application. Input constraints may be caused by actuator limits (a valve can have an opening between 0 and 100%, maximum pressure set point is x bar,...) or by comfort limits (such as maximal acceleration). State and output constraints may be motivated by safety (maximal engine load) or quality considerations (maximum concentration of a given substance). The feasible region described by the linear inequality constraints is temporarily denoted by C. In the matrix notation these constraints will be specified.

Operating principle Each time step the following optimization problem is solved:

$$\min V(\mathbf{y}_1, \ldots, \mathbf{y}_N, \mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{u}_0, \ldots, \mathbf{u}_{M-1}) \tag{2.4}$$

subject to:

$$\begin{array}{rcll} P(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k, \mathbf{u}_{k-1}), & & & k \in 1, \ldots, N \\ (\mathbf{y}_k, \mathbf{x}_k, \mathbf{u}_k) & \in & C & \end{array}$$

with $\mathbf{u}_k = 0, \forall k \geq M$ and P the equality constraint defined by the plant model of Eqn (2.3).

Since the objective function is quadratic and the constraints are linear, this problem can be solved using Quadratic Programming. The first element of the solution vector is then applied to the system. In the next time step, this procedure is repeated using the new process measurements. This is why MPC is also often called Receding Horizon Control (RHC). This is illustrated in Fig. 2.3. At every time step the horizon recedes one time step.

Matrix notation The optimization problem described in Eqn (2.4) can be recast to a more convenient matrix notation. First of all, stacked variables $\mathbf{x}_s$, $\mathbf{y}_s$ and $\mathbf{u}_s$ are defined as follows:

$$\begin{array}{rcl@{\qquad}rcl} \mathbf{x}_s & := & \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \end{bmatrix} & \mathbf{x}_{s,ref} & := & \begin{bmatrix} \mathbf{x}_{ref,1} \\ \mathbf{x}_{ref,2} \\ \vdots \\ \mathbf{x}_{ref,N} \end{bmatrix} \\ \mathbf{y}_s & := & \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_N \end{bmatrix} & \mathbf{y}_{s,ref} & := & \begin{bmatrix} \mathbf{y}_{ref,1} \\ \mathbf{y}_{ref,2} \\ \vdots \\ \mathbf{y}_{ref,N} \end{bmatrix} \\ \mathbf{u}_s & := & \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_M \end{bmatrix} & & & \end{array} \tag{2.5}$$

The optimization problem of Eqn (2.4) can then be rewritten as:

$$\min W(\mathbf{y}_s, \mathbf{x}_s, \mathbf{u}_s) \tag{2.6}$$

subject to

$$\begin{array}{rcl} \mathbf{b} & = & \mathbf{A}_s\mathbf{x}_s + \mathbf{B}\mathbf{u}_s \\ \mathbf{y}_s & = & \mathbf{C}_s\mathbf{x}_s \\ 0 & \geq & \mathbf{D}_s\mathbf{x}_s + \mathbf{F}_s\mathbf{u}_s \end{array} \tag{2.7}$$

with

$$W(\mathbf{y}_s, \mathbf{x}_s, \mathbf{u}_s) = \begin{aligned} &(\mathbf{y}_s - \mathbf{y}_{s,ref})^T \mathbf{S}_s (\mathbf{y}_s - \mathbf{y}_{s,ref}) \\ &+ (\mathbf{x}_s - \mathbf{x}_{s,ref})^T \mathbf{Q}_s (\mathbf{x}_s - \mathbf{x}_{s,ref}) \\ &+ \mathbf{u}_s^T \mathbf{R}_s \mathbf{u}_s \end{aligned} \tag{2.8}$$

where the subscript s denotes that the matrix or vector belongs to this formulation with stacked vectors. For instance, matrix $\mathbf{Q}_s$ is a block diagonal matrix of the matrices $\mathbf{Q}_k$ of the objective function in Eqn (2.1)).

2.2.3 Benefits of MPC

MPC designs are able to yield high performance control systems capable of operating without expert intervention for a long period of time. Below the most important benefits of MPC are discussed.

flexible constraint handling: it is well known that the economically optimal operating point of a process often lies at the intersection of constraints (Prett and Gilette, 1980). A successful industrial controller must therefore be able to maintain the system as close as possible to constraints without violating them.

simple tuning: most of the tunable parameters are situated in the objective function, which makes them relatively easy to interpret. The algorithm itself is also very easy to understand, which makes it well suited for industrial implementation.

time-varying model: because the optimization is performed online in every time step, the model can easily be updated. This is another advantage of MPC over LQR (Qin and Badgwell, 2003).

time-varying control objective: because of the online optimization, the objective can be updated online as well.

model-based: because it is a model-based control technique, known delays and dynamics can be taken into account.

Multivariable control theory: MPC can easily handle multivariable control problems.

2.2.4 Feasibility

A given solution vector is feasible if it satisfies all the constraints of problem (2.6). Feasibility can be guaranteed by using soft constraints, or by using a two step initialization.

soft constraints: a soft constraint means that a slack variable is added to the constraint. Some norm of this slack variable is then included in the objective function to make sure that the slack variable, which is a measure of the constraint violation, is as small as possible. Most often this penalty term is a 1-norm or an ∞-norm of the slack variables. For the case of an ∞-norm the same slack variable is used for all the constraints. The advantage of both these norms over

the standard 2-norm is that the penalty term is linear in the error. A quadratic penalty term favors the control error as soon as the constraint violation term becomes smaller than the control error term. The violation will in other words never be reduced to zero. More information about penalty functions for soft constraints can be found in Rao et al. (1998).

Example 2.1. *Assume the following scalar constraint*

$$x \leq X_{max} \tag{2.9}$$

where x is a process variable and X_{max} is the upper bound. This constraint can then be softened as follows:

$$\begin{aligned} x &\leq X_{max} + \epsilon \\ 0 &\leq \epsilon \end{aligned} \tag{2.10}$$

with ϵ a slack variable. The term $P(\epsilon)$ then needs to be added to the objective function. If the 1-norm is used, $P(\epsilon)$ becomes:

$$P(\epsilon) = \gamma\epsilon \tag{2.11}$$

with γ a gain factor. Of course this gain factor needs to be chosen sufficiently large to avoid a trade-off between this constraint violation term $P(\epsilon)$ and other terms in the objective function.

□

initialization: the drawback of using soft constraints is that this procedure requires to attach a cost to a constraint violation, which always implies some trade-off between control error and constraint violation. For instance, in Example 2.1, the constraint violation is controlled by the gain factor γ. In most cases reduction of the constraint violation has a higher priority than the reduction of the control error. Instead of calculating the slack variables and the optimal input in the same step, one could split up the problem into two phases:

1. calculate the initialization value of the problem in such a way that the slack variables (which are also added for this approach) are minimized subject to the other constraints.
2. solve the actual control problem starting from the previously calculated initialization value and using the modified constraints (where the slack variables are added to the right-hand side as calculated in the previous step). There is no need for a term in the slack variables in the objective function since these values have already been determined in step 1, and are fixed for this step.

The main advantage of this strategy is that the constraint relaxation is independent from the actual control solution. This two-step approach guarantees that the constraint violation is first minimized as much as possible. After that the remaining degrees of freedom can be used to optimize the other terms in the objective.

Further on the two-step initialization procedure will always be used to guarantee feasibility. Since feasibility is closely related to the solution method, this is also discussed in Chapter 9.

2.2.5 Stability

The early versions of MPC did not automatically ensure stability. It is therefore not surprising that research devoted considerable effort to this topic in the 1990s. An overview of the advances made in this field is given by Mayne et al. (2000).

Since MPC of constrained systems is nonlinear, Lyapunov stability theory is the primary tool to examine stability. This did not receive any attention during the 1970s and 1980s, with one exception: Chen and Shaw (1982) showed that the objective function itself could be used as a Lyapunov function to establish stability of continuous time receding horizon control of unconstrained systems when a terminal equality constraint is employed. Unfortunately, this paper went by unnoticed for quite some years. It was only after the papers by Keerthi and Gilbert (1988) and Mayne and Michalska (1990) that the objective function was almost universally employed as the Lyapunov function for the stability analysis of MPC.

When the prediction horizon is finite, the stability of the controller can no longer be guaranteed (Mayne et al., 2000). After all, a controller might postpone the necessary control actions to the end of the horizon in order to minimize the objective. At a given point this would lead to instability. The original lack of general stability theory for receding horizon controllers with a finite horizon was recognized as a drawback. This was handled in the nineties by modifying the control problem. Each method makes use of a so-called terminal cost $F(\mathbf{x}(N))$ (which is a cost function on the last state of the prediction horizon) and/or a terminal constraint $\mathbf{x}(N) \in X_f$ (which demands that the last state of the horizon is contained in a given set). Four approaches can be distinguished:

terminal equality constraint: in this case the terminal cost is $F(\cdot) \equiv 0$ and the terminal set $X_f = \{x_{ref}\}$ with x_{ref} the reference for the state. The first proposal for this form of MPC for time-varying, constrained, nonlinear, discrete-time systems was made by Keerthi and Gilbert (1988). It is an interesting fact that MPC with a terminal equality constraint can also be employed to stabilize systems that cannot be stabilized by continuous feedback controllers (Meadows et al., 1995).

terminal cost function: here the terminal cost $F(\cdot)$ is non-trivial and there is no terminal constraint (in other words $X_f = \mathbb{R}^n$). The first proposal for the terminal cost function was made by Bitmead et al. (1990). However, a more interesting example is a proposal by Rawlings and Muske (1993). When the system to be controlled is linear and stable, and subject to a convex control constraint, the terminal cost $F(\cdot)$ is chosen to be the infinite horizon value function associated with the stabilizing controller $\mathbf{u} = \kappa_f(\mathbf{x}) \equiv 0$.

terminal constraint set: for this approach X_f is a subset of $\mathbb{R}^n$ containing a neighborhood of the reference signal and there is no terminal cost ($F(\cdot) \equiv 0$). The purpose of the model-based predictive controller is to steer the state into X_f in finite time. Inside X_f a local stabilizing controller $\kappa_f(\cdot)$ is used. This form of MPC is sometimes referred to as dual control, and was proposed by Michalska and Mayne (1993) where a variable horizon N is employed.

terminal cost and constraint set: most recent MPCs belong to this category. Ideally the terminal cost $F(\cdot)$ should compensate the difference between the finite and

the infinite horizon objective function. In practice, this is often not possible in general because of constraints and nonlinearities. This is however possible in a suitable neighborhood of the reference, and this is where the constraint comes in. Stability is shown by Parisini and Zoppoli (1995) and Scokaert and Rawlings (1998).

Note that all the above approaches come down to compensating the difference between an infinite and a finite horizon objective function in one way or another. For the applications treated in this PhD, the horizon will always be taken sufficiently large, which means that a terminal cost or constraint is not necessary (Primbs and Nevistic, 2000).

2.2.6 Nonlinear MPC

Although most industrial processes are nonlinear, most MPC applications are based on the use of linear models. There are several reasons for this, namely (Camacho and Bordons, 2004):

- identification of linear models is generally much easier than of nonlinear models.
- since most controllers operate around a set point, there is not much performance difference between linear and nonlinear models. Moreover, linear models can more easily be made adaptive to deal with slowly changing plant dynamics.

The number of applications of Nonlinear MPC (NMPC) is still very limited (Qin and Badgwell, 1998, Santos et al., 2001, Karacan, 2003, Kawathekar and Riggs, 2007), but it has huge potential in those areas where process nonlinearities are strong and market demands require frequent changes in operation regimes. These transition regimes often contain severe nonlinearities.

There is no conceptual problem with using nonlinear models in MPC. The extension of MPC to nonlinear models is, however, not straightforward. The optimization problem is no longer guaranteed to be convex, which leads to local optima and stability problems. Thus the study of stability and robustness also becomes much more complicated. Moreover, the computational complexity increases considerably.

An application of NMPC in agriculture is presented in Chapter 10. In that chapter a cruise control system for a combine harvester is developed. On the road, the pump setting as well as the engine speed of the diesel engine may be modified to reach the desired speed. As one can expect, the relation between those two inputs and the travel speed is nonlinear. This extra degree of freedom - two inputs for one output - can be used to minimize the engine speed, and thus reduce the noise production and exhaust fumes.

2.2.7 Applications

MPC quickly became popular, particularly in chemical process industries due to the simplicity of the algorithm. An overview of MPC applications in the eighties can be found in Garcia et al. (1989). Whereas Qin and Badgwell (2003) show that in

mid-1999 the majority of the MPC applications was still in the area of refining, petrochemicals and chemicals, MPC is now slowly penetrating other sectors.

Bellemans et al. (2006) applied MPC to a traffic problem, and Keviczky and Balas (2006) illustrated how MPC might work for the control system of an F16 aircraft.

In the meantime MPC has also been used in the agricultural sector. Vranken et al. (2005) used MPC to control the ventilation rate of fans in a stable, and Blasco et al. (2007) applied MPC for climate control in a greenhouse.

2.3 Model uncertainty

2.3.1 Uncertainty description

In many cases the plant model is not exact. This can have several reasons:

- the process is complex and/or nonlinear, which makes the development of an exact process model too expensive.
- an exact process model is available, but this model is too complex to be able to calculate a solution every time step.
- there are unmeasured (and/or impossible to model) process disturbances, such as the temperature, the atmospheric conditions or the properties of the raw materials.
- the model of the plant varies in time. This can be dealt with by making the model adaptive, but this also implies that transition phases occur during which there is a significant difference between the plant model and the actual process.

In literature there are two main methods to quantify this uncertainty (Maciejowski, 2002), namely norm-bounded uncertainty and polytope uncertainty.

Norm-bounded uncertainty works with a transfer function notation of the process model. Assume that the nominal plant model (without uncertainty, a sort of average case), can be written as $P_0(z)$, then the general plant model (including uncertainty) is written as:

$$P(z) \quad = \quad P_0(z)\,(\mathrm{I} + \Delta) \tag{2.12}$$

with Δ a stable, bounded operator. Often $P_0(z)$ is normalized in such a way that:

$$\|\Delta\| \leq 1 \tag{2.13}$$

Various norms can be used, the most commonly used norm is the 'H-infinity' norm $\|\Delta\|_\infty$.

Polytope uncertainty assumes that a number of 'corner' points for the model are known, leading to following 'corner models':

$$\mathbf{x}_{k+1} = \mathbf{A}_i\mathbf{x}_k + \mathbf{B}_i\mathbf{u}_k \qquad\qquad i = 1, 2, \ldots, L \tag{2.14}$$

with L the number of 'corners'. The real plant then lies within the polytope[1] which is the convex hull of these corners:

$$[\mathbf{A}, \mathbf{B}] = \sum_{i=1}^{L} \lambda_i \left[\mathbf{A}_i, \mathbf{B}_i\right], \qquad \sum_{i=1}^{L} \lambda_i = 1 \tag{2.15}$$

The real plant does not have to be fixed. It can vary in time as long as it stays within this polytope (Kothare et al., 1996).

2.3.2 Robust MPC

Robust MPC is an umbrella term for all MPC approaches which take the model uncertainty into account. Up till recently, nearly all of these approaches were deterministic. Further on, in Chapter 8, a stochastic approach to control uncertain systems is described. These stochastic techniques are often collectively referred to as Stochastic MPC, but are nevertheless a form of Robust MPC.

Robustness can be situated at two levels, robust feasibility and robust performance. Since the Robust MPC thought is at the foundation of Stochastic MPC, this section briefly presents the Robust MPC methodology. Kothare et al. (1996) propose a strategy to maintain stability while respecting constraints despite the uncertainty in the model description. This approach can be applied to norm-bounded, structured uncertainty descriptions as well as to polytope descriptions (Maciejowski, 2002, Pluymers, 2006), and is further described below.

Assume that the system is modeled by a time-varying linear model:

$$\mathbf{x}(k+1) \quad = \quad \mathbf{A}_k \mathbf{x}(k) + \mathbf{B}_k \mathbf{u}(k) \tag{2.16}$$

where $\mathbf{A}_k$ and $\mathbf{B}_k$ are the time-varying system matrices at time step k. These matrices are not known over time. The only given is that they are contained in the following polytope:

$$[\mathbf{A}_k, \mathbf{B}_k] \quad \in \quad \Omega \tag{2.17}$$

where Ω is a convex polytope derived from a norm-bounded uncertainty description or a polytope uncertainty description. The cost function is defined to be:

$$V_\infty(k) \quad = \quad \sum_{j=0}^{\infty} \left(\|\hat{\mathbf{x}}(k+j|k)\|^2_{\mathbf{Q_1}} + \|\hat{\mathbf{u}}(k+j|k)\|^2_{\mathbf{R}} \right) \tag{2.18}$$

where the hat (for example $\hat{\mathbf{x}}$) indicates a predicted value. $\hat{\mathbf{x}}(k+j|k)$ for instance indicates the value at time step $k+j$ of $\mathbf{x}$, predicted at time step k

One of the common ways to achieve robustness against the effect of model uncertainty is to use a 'min-max' formulation. This means that the objective function is first maximized in terms of the model uncertainty, and then minimized in terms of the input:

$$\min_{\mathbf{U}(k)} \quad \max_{[\mathbf{A}(k+j), \mathbf{B}(k+j)] \in \Omega} V_\infty(k) \tag{2.19}$$

[1] A *polytope* is a finite region of n-dimensional space bounded by a finite number of hyperplanes. This region is not necessarily convex, though the particular polytope referred to here is convex.

with $\mathbf{U}(k) = \left[\hat{\mathbf{u}}(k|k)^T, \ldots, \hat{\mathbf{u}}(k+N_C-1|k)^T\right]^T$, and N_C the control horizon.

This problem can then be recast by assuming that a function $W(\mathbf{x}) = \mathbf{x}^T\mathbf{P}\mathbf{x}$ exists, with $\mathbf{P} > 0$, such that for any pair $(\hat{\mathbf{x}}(k+j|k), \hat{\mathbf{u}}(k+j|k))$, corresponding to the model of Eqn (2.16),

$$W\left(\hat{\mathbf{x}}(k+j+1|k)\right) - W\left(\hat{\mathbf{x}}(k+j|k)\right) \leq -\|\hat{\mathbf{x}}(k+j|k)\|^2_{\mathbf{Q}_1} - \|\hat{\mathbf{u}}(k+j|k)\|^2_{\mathbf{R}} \quad (2.20)$$

for each $j \geq 0$. This function can serve as an upper bound for the cost $V_\infty(k)$. Assume that $V_\infty(k) < \infty$, then $\hat{\mathbf{x}}(\infty|k) = 0$, and hence $W\left(\hat{\mathbf{x}}(\infty|k)\right) = 0$ (since $W(\mathbf{x}) = \mathbf{x}^T\mathbf{P}\mathbf{x}$). Summing both sides of Eqn (2.20) from $j = 0$ till $j = \infty$ then yields

$$-W\left(\mathbf{x}(k|k)\right) \leq -V_\infty(k) \quad (2.21)$$

Since Eqn (2.20) holds for any model in the uncertainty set, we have that

$$\max_{[\mathbf{A}(k+j),\mathbf{B}(k+j)]\in\Omega} V_\infty(k) \leq W\left(\mathbf{x}(k|k)\right) \quad (2.22)$$

Problem (2.19) can then be replaced by the more convenient problem:

$$\min_{\mathbf{U}(k)} W\left(\mathbf{x}(k|k)\right) \quad (2.23)$$

subject to Eqn (2.20) and $\mathbf{P} > 0$. Replacing the actual objective function by the upper bound on the objective $W\left(\mathbf{x}(k|k)\right)$ introduces some conservatism in the formulation. Kothare et al. (1996) however state that this will not lead to much conservatism in practice. The main advantage of Eqn (2.23) is that this is a convex optimization problem, and can be solved with the aid of Linear Matrix Inequalities (LMIs). More details can be found in Maciejowski (2002).

2.3.3 Stochastic MPC

As stated above, Stochastic MPC is actually a form of Robust MPC. Most robust MPC applications use however a deterministic formulation for the uncertainty, whereas Stochastic MPC chooses a stochastic approach. Consider the following example:

Example 2.2. *Assume the following system with one state:*

$$z_{k+1} = z_k + \alpha u_k \quad (2.24)$$

A deterministic approach to robustness would describe the uncertainty on the parameter α as:

$$\alpha \in [\alpha_{min}, \alpha_{max}] \quad (2.25)$$

whereas a stochastic description of the uncertainty of α is:

$$\alpha \propto \mathcal{N}\left(\bar{\alpha}, \sigma_\alpha^2\right) \quad (2.26)$$

with $\bar{\alpha}$ the expected value of the stochastic variable α. α is in other words normally distributed with mean $\bar{\alpha}$ and variance σ_α^2. □

This approach is very well suited for applications where the main objective is to have the best performance possible at all times, without guaranteeing stability. The state of the art of Stochastic MPC is discussed in Chapter 8, followed by an overview of the extensions made in this research.

2.3.4 Multiple operating states

Suppose you have a plant which has multiple operating states, and of which the process also changes considerably depending on the process state. These different regimes may be caused by malfunctions, shut downs or reconfiguration for preventive maintenance or capacity planning. However, if this type of situations occur regularly, and they form a considerable source of model uncertainty, it is probably best to build a model for each situation. In Narendra and Balakrishnan (1997) it is described how an MPC can automatically switch between the different process models based on some criterion of model performance. This is clearly the best approach if the process models form a countable set. Whether or not to take this approach depends on the trade-off between the economic benefits and the modeling cost.

2.4 Conclusion

First of all this chapter explains the relevance of control theory to society. Since MPC is the chosen control methodology of this research, the chapter then elaborates on the state of the art of MPC. The development of MPC is somewhat atypical. Unlike many other control paradigms, the origin of MPC actually lies in industry. It is only because of the massive interest of industrial practitioners, that academia started to pay attention to MPC.

In the meantime, a lot of research has been performed to make MPC robust to plant uncertainty. Most of these methods are, however, deterministic. The 'best' controller is calculated taking into account the possible parameter range, and thus making sure that the controller remains stable and feasible. Stochastic MPC, as will be explained further on (Chapter 8), takes an entirely different approach. The controller seeks to optimize the expected value of the objective function, and respects the constraints with some predefined probability.

Chapter 3

Agricultural machinery

Farmers only worry during the growing season,
but town people worry all the time.
-Edgar Watson Howe-



This PhD focuses on the application of MPC to agricultural automation. In this chapter the main challenges of agricultural machinery are illustrated, as well as the economical relevance of the agricultural sector. Agriculture has become a hot topic over the last few years because of the advent of biofuels and the pending food crisis. An illustration thereof is that in 2008 there were more than three times as many articles about agricultural products such as wheat than in 2006 (W, De Tijd (2008), W, De Standaard (2008)).

In this chapter, first of all the usefulness of automation on agricultural machinery is further motivated. Secondly some of the basic working principles of combine harvesters are explained, and the recent developments as well as the road map of combine harvester automation are discussed.

3.1 Automation in agriculture

3.1.1 Historical context

One of the first sectors that became mechanized was the agricultural sector. For instance, the first steam powered threshing engine was operational in England as early as the 1830s (Quick and Buchele, 1978). Before the invention of the steam engine, which marked the start of the industrial revolution, the majority of the people had to work in the fields to provide enough food for everyone. The industrialization of the 18th and 19th century could in other words only occur because the efficiency of agriculture was increased.

Since agriculture and food production were such dominant aspects of everyday life, it is no surprise that many of the first control applications were linked to the food industry, such as the wind mill and the egg incubator (Durham, 2004).

The evolution of the productivity of farmers over the last 200 years is shown in Fig. 3.1. The total agricultural production of the USA increased over this period,

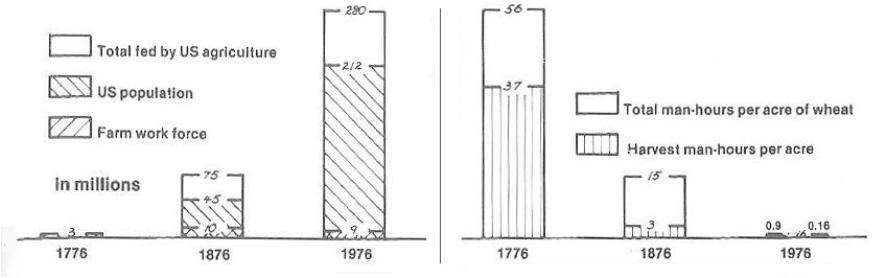


Figure 3.1: Man and machine productivity in cereal production over the Bicentennial period 1776-1976 (Quick, 1976, Quick and Buchele, 1978).

whilst the percentage of the population involved in farming shrunk. This is once more illustrated on the right hand side of Fig. 3.1 where the number of man-hours per acre is expressed over time. If the USA had to harvest its crops using the hand methods of the 1830's, every American, man, woman and child would be needed in the fields every day during the harvesting season.

In Belgium (with a population of a little over 10 million) there are 100 000 farmers left, of which only 50 000 work full-time (W, FOD Economie (2008b)). Belgium is, however, still a net exporter of agricultural products (W, VILT (2008)). The conversion of Belgium from an agricultural to an industrialized society, and now to a service ('knowledge') economy is only possible because of agricultural mechanization.

The effect of mechanization can be illustrated by comparing the agricultural production and work force of the United States and the USSR in the seventies. Fig. 3.2 shows that an American farmer was about 10 times more efficient than a Russian farmer in 1975. This difference cannot be entirely attributed to mechanization, but it gives an indication of the possible efficiency increase.

One may think that the development of agricultural machinery has finished. However, up till now agricultural machinery has been mechanized, but not really automated. This means there still is considerable improvement possible.

3.1.2 Environmental & social context

Environmental The European Community imposes more and more stringent environmental legislation on large machinery concerning noise production and exhaust fumes (European Commission, 2004, European Commission, 2005, European Commission, 2006). The purpose of this legislation is to trigger research and development towards more efficient machines, which for instance have a cruise control system to reduce exhaust fumes and noise production (Coen and De Baerdemaeker, 2007, Coen et al., 2007a).

Pesticide and fertilizer reduction is another hot topic. Farmers will be forced to reduce their use of pesticides to a bare minimum. The most straightforward way to do this, is to apply site-specific spraying (Gerhards et al., 2002). The objective is to collect yield data during harvest, and to use this field information to determine the ideal fertilizer and pesticide use for the next harvest.

In some parts of the world (including Belgium) wind and water erosion is a major

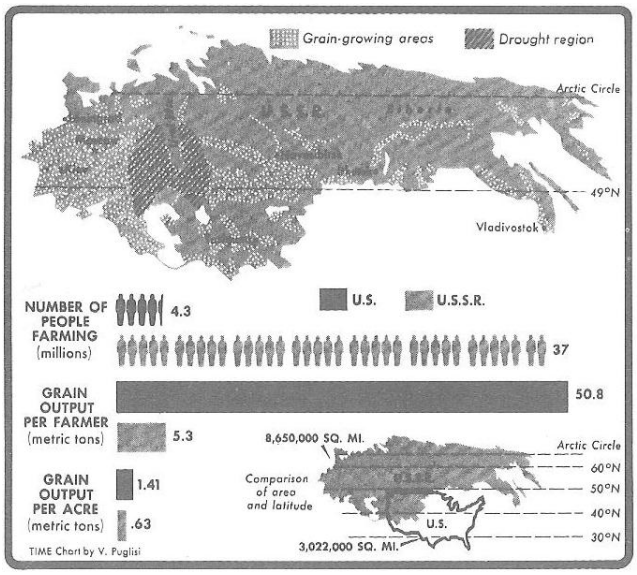


Figure 3.2: Russian-American grain production compared in 1975 (Quick and Buchele, 1978).

problem. Research indicates that there is a relation between tillage and erosion (Hevia et al., 2007, Marques da Silva et al., 2004). Therefore farmers use minimal tillage (replacing tillage by alternative field preparation methods) and keep the soil crop-covered for the larger part of the year. These changes in the agricultural practice give rise to new specifications for combine harvesters and other agricultural equipment (Kutzbach, 1997).

Social In the last few years several contaminations of the food chain occurred, which renewed the awareness of the authorities and society to the importance of good food quality and traceability. For this purpose extra sensors are placed on harvesting machinery, which allow to check and log the instantaneous quality of the crop. The potential of these sensors has been shown, but the adoption on commercial machinery will require serious investments from the machine manufacturers (De Castro, 2002, Sarig, 2003).

3.1.3 Economical context

In this section it is examined whether there is a demand in the market for more advanced (i.e. more automated), and thus more costly equipment. The demand for new machinery is driven by legislation (current legislation as well as legislation that can be foreseen to be implemented in the next few years) and economics. Legislation determines which machinery is acceptable, the further decisions depend on the financial picture. Elements such as product price, scale size (of the farms and of the machine manufacturers) and the demand in the market (next to the price, also the diversity of the demand) influence which type of machinery is developed.

Customer scale size Machine manufacturers have two types of customers, namely farmers and contractors. In Belgium there is a clear tendency towards fewer but larger farms (STM, 2006, RMG, 2007). There is also a world-wide trend noticeable to work with contract farming (Kutzbach, 1997). This means that the contractor undertakes all operations from soil preparation to harvest and transport. The advantage of this approach is that such contractors are specialized in a limited selection of crop types, which allows them to invest in more specific (and also more advanced) machinery.

Manufacturer scale size In the last decades the capacity of farming machinery has increased enormously, which implies fewer machines are needed for the same production level. For combine harvesters for instance, even a global player only produces a couple of thousand combines. The worldwide combine harvester market was a little under 30 000 units in 2007 (W, CNH Global (2008a)).

After a couple of bad years (due to bad harvests), investments in farming machinery in Belgium are increasing again (VDB, 2007). This trend is also clear on a global scale, where the worldwide combine harvester industry is even up 40% in the first quarter of 2008 compared to 2007 (W, CNH Global (2008b)).

The changes on the demand side (next paragraph) will further boost the turnover of machine manufacturers, enabling those manufacturers to free up more resources for R&D and build more advanced machinery.

Demand in the market The price of a product depends on the balance between supply and demand. If the demand increases, this will often lead to higher prices. This is not any different for agricultural products such as wheat and corn. Moreover, if the market for agricultural products grows, the market for combine harvesters will follow.

The demand for agricultural products rises because of the following facts:

growing number of consumers: as the world population increases, and especially as more and more parts of the world develop, the demand for agricultural products grows (W, Club of Bologna (2008)).

growing meat requirement: each kilogram of meat requires 8 kilogram of grain (to feed the livestock). As the income in China and India grows, people eat more meat, leading to a larger demand for grain (Mooijman, 2007a).

biofuels: biofuels have many advantages over traditional fuels. Biodiesel for example is environmentally superior to traditional fuels (50% less CO, 80% less CO_2), it actually cleans engines, and crops for biodiesel can be grown in arid, marginal, degraded lands (Gasparetto, 2005, Best, 2005). The main origin for biodiesel are the following crops: soybean, rapeseed, sunflower, palmoil, linseed and canola. A lot of these crops are harvested with combine harvesters, but up till now their volume was not important enough to have an influence on the design of combine harvesters. In the new biofuel context, manufacturers will have to invest in improving the performance of their combine harvesters in crops such as canola. This may even lead to new, dedicated machines.

The biofuels will also lead to a significant increase in crop production, often in areas which are not interesting for classic food crops such as wheat and corn.

Because of the European decision to have at least 5.75% of biodiesel in fuels by 2010 (and even 10% by 2020), a lot of extra land is required. For Belgium alone this would mean an extra 260 000 ha of agricultural area (SDW, 2006). This is of course not realistic since this is almost 9% of the total surface of Belgium, but on a global scale a lot of extra agricultural area will be created in regions which are not interesting for classic crops.

At this moment it is not clear whether biofuels will become as important as is currently set out by European legislation (through the biodiesel quota). The last few months there was a lot of criticism on biodiesel. According to the critics, biodiesel would have led to the steep price increase of food crops such as wheat and corn (Ghijs, 2008, CP, 2008). These statements are however contradicted by the European Commission (DDW, 2008). Environmental organizations on the other hand, even refer to biofuels as dangerous for the environment (Delepeleire, 2008). Whatever the truth may be, biofuels will take some place in our energy supply for the years to come.

This growing grain requirement inevitably leads to an increased demand for agricultural machinery. Moreover, in the Western world, the automation of agriculture was an enabling factor for further economical growth. In large parts of the world a lot of work is still performed by hand. As the industrialization of a country proceeds, the necessary mechanization of agriculture will boost the sales of agricultural equipment. Mechanization is already a hot topic in China (Li, 2004) and Central European countries (Segre, 2004), and will become important in Africa (Clarke, 2003) as well in the next few decades.

Note that, in light of the previous section about the environmental and social issues, the manufacturers will need this increased turnover to be able to invest in the development of more advanced machinery which meets the requirements of modern society.

Product price Any combine harvester has grain losses, as will be shown in section 3.2.1. However, the amount of grain losses can be minimized by using a state of the art combine harvester. The investment decision for more advanced combines depends to a large extent on the grain price, since it is actually a trade-off between the cost of losses and the cost of the machinery.

From mid 2007 onwards there has been a steep increase of the price of agricultural products (Mooijman, 2008, Verhoeven, 2007, Bloomberg, 2007, WLE, 2008, De Preter, 2008, De Greef, 2008). This may have been a temporary effect, but there is reason to believe that the price of agricultural products will never return to the levels of a couple of years ago, which would mean that the budget available for investments in agricultural machinery has been permanently increased. Food products were simply too cheap. In Belgium for instance in the period 1990-2007 the inflation was about 2.1% (W, FOD Economie (2008a)), whereas the prices of food products only rose at 1.6% a year. In other words, food products became cheaper (Mooijman, 2007b) year after year.

In Fig. 3.3 the evolution of the grain price over the last 100 years is shown. To get an idea of whether there is a real price increase, the inflation over this period has to be taken into account as well to convert past grain prices to today's money. The

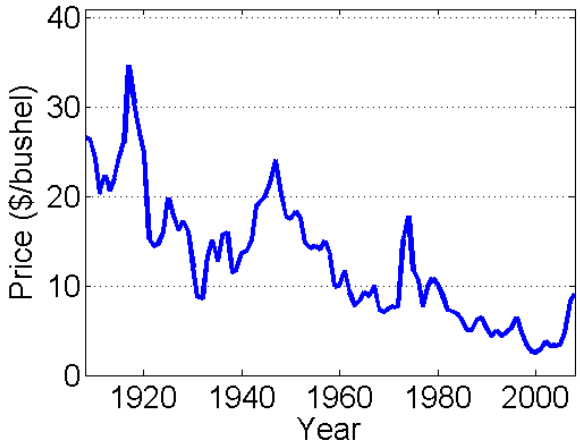


Figure 3.3: Evolution of the wheat price over the last century (W, National Agricultural Statistics Service (2008) and W, De Tijd (2008)). The past grain prices are converted to today's money by taking the inflation into account (W, US Department of Labor, Bureau of Labor Statistics (2008) and W, Wikipedia (2008b)).

price per bushel in 1908 corrected for the inflation was for instance almost $30 per bushel, which is three times today's price. Wheat clearly became cheaper through the mechanization of the agriculture which started in the fifties.

3.1.4 Commercial context

The previous section described the drivers for more advanced machinery, as well as the economics of the sector. A manufacturer will, however, invest in the development of new machinery based on commercial arguments, within the legislative constraints. This section illustrates how advanced methodologies (such as MPC) may bring added value for the customer, and how this warrants more expensive machinery.

Operator In the earlier years of mechanization operators were employed by the farm owner. Nowadays it happens more and more that the farm (and machine) owner operates the machine himself (whilst he is making logistic arrangements), or that an unskilled operator (a seasonal worker) drives the combine. This is the only option because the farmer no longer requires a large workforce outside of the harvest period. Because of this evolution skilled operators are increasingly hard to find, which creates a demand for systems that relieve the operator task (to free up time for other activities) or simplify it (to operate the combine at maximal capacity without having knowledge of the machine). This makes that farmers and contractors are willing to pay more to have a machine which can be operated by a less skilled (and thus less expensive) work force.

Moreover, as crops become more diverse (for instance crops as canola have become popular because of the biofuels), it becomes harder to find operators that have



Figure 3.4: CX combine harvester on tracks in corn (W, New Holland Benelux (2008)).

experience in each of those crops.

Return on investment As mentioned above, due to the rising price of crops, more advanced machinery can become the best choice because they have lower losses.

3.2 Combine harvesters

3.2.1 Working principle of combines

In Fig. 3.4 a combine harvester during corn harvest is shown. Such a combine harvester is a versatile machine that has to operate in diverse environmental circumstances and in a whole range of different crops going from peas to corn. In between two crop types a combine typically needs to be modified: a different header (corn header in the picture of Fig. 3.4), different threshing concaves, different sieves, and for muddy conditions the front wheels can be replaced by tracks.

At this time, there are no combine harvesters that are perfect for all crops. The design of a combine depends on the ratio between the different crops. If a given crop (such as canola) becomes more important, this will have an effect on the design of combine harvesters. There are also local sensitivities. For North America for instance, the quality of the straw is much less important than for France or Belgium. In South America it is important that the combine can still operate (possibly at a lower efficiency) if all the electronics break down. A combine harvester is thus designed with a certain customer type in mind.

However, almost every combine harvester obeys the same operation schematic. The different parts of the combine process are shown in Fig. 3.5. Note that transport

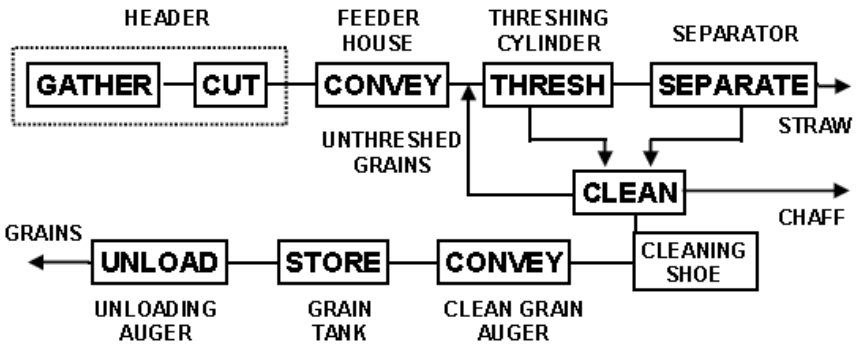


Figure 3.5: Process diagram of a combine (Srivastava et al., 2006).

through the machine is a very important part of the process. This implies that it can be expected that time delays between input and output of the process will be considerable.

Crop collection First of all, the crop is gathered or cut by the header. The principle of the header may vary depending on the crop or the harvesting procedure (pick-up header, corn header, small grain header, etc.), but each header contains some kind of auger, which transports the crop to the center of the header where it enters the machine.

A canola header for instance is a wheat header with some special precautions (scissors on the sides, sometimes a flexible bottom plate). In corn only the ear (the corn cup) enters the machine and the straw (the stem of the corn) is chopped on the field. A corn header strips the cups from the stems and has choppers under the header to chop the stem. A corn header is an entirely different construction than a header for small grains, such as wheat and barley. In some regions of the world the grain does not ripen, sufficiently because of the wet climate. To start the ripening process, the grain is mowed, and put in swaths. Afterwards a combine equipped with a pick-up header - which gathers the crop instead of cutting it - threshes the grain. Designing headers is an art in its own; proof of this are the many independent companies designing and selling headers (W, Harvestec (2008), W, Northstar (2008), W, Shelbourne Reynolds (2008)).

Threshing The feeder (or straw elevator) transports the unthreshed grains from the header to the inside of the combine. A combine is equipped with one of the two following threshing systems:

conventional threshing: a conventional system contains several threshing drums which are placed transversal to the machine axis, and in series with each other. A conventional combine has an independent separation system using straw walkers behind the threshing system.

rotary threshing: a rotary combine contains one or two cylinders along the longitudinal axis of the machine. The first section of the cylinder is used for threshing, whereas the second half serves as separation section (Fig. 3.6).

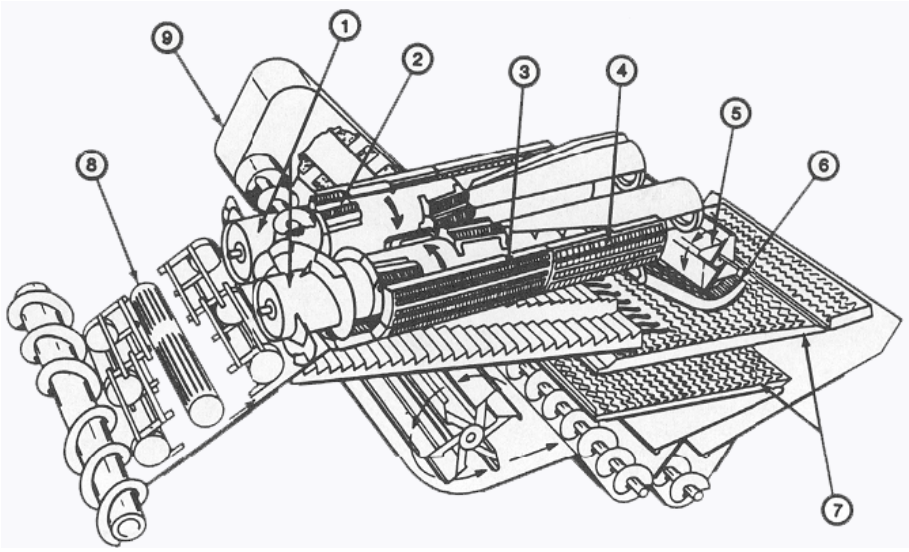


Figure 3.6: An axial flow rotary combine utilizing twin rotors: (1) rotor, (2) rasp bars, (3) threshing concave, (4) separating concave, (5) discharge beater, (6) beater grate, (7) cleaning shoe, (8) feeder housings, (9) tailings auger (courtesy of Prairie Agricultural Machinery Institute, Canada) (Srivastava et al., 2006).

In both conventional and rotary combines threshing is accomplished by rotating a cylinder against a concave grate. As the cylinder rotates, crop is forced through the gap between the concave and the cylinder, and the crop is subjected to impact and rubbing action that causes grains to be detached from the ears. In a conventional combine the crop flow is transverse to the axis of rotation, whereas in a rotary combine the crop flow is parallel to the axis of rotation. Since in this research a rotary combine harvester is used (Chapter 5), the rest of this paragraph will focus on rotary threshing.

The cylinder speed is determined by the crop type and condition. Wet, hard-to-thresh conditions require higher speeds, but grain damage increases as cylinder speed is increased. Another factor affecting the quality of threshing is the cylinder-concave gap. If the gap is too large, the crop is not threshed completely. A too narrow gap results in excessive power and grain damage. The length of the threshing cylinder is proportional to the threshing capacity of the combine harvester.

The performance of threshing mechanisms is measured by the threshing efficiency, the separation efficiency, the amount of grain damage and the amount of straw breakup. Threshing efficiency is the percentage of the threshed grains relative to the total amount of grain entering the threshing mechanism. It increases asymptotically with cylinder length up to a certain point. The unthreshed grains are called threshing losses. Threshing losses increase with material feed rate, which is generally expressed in terms of tons of biomass per hour.

The separation efficiency of the threshing cylinder is defined as the percentage of grains separated during the threshing phase relative to the total amount of grain entering the combine. For a rotary combine this threshing phase only includes the

first half of the rotor. A major portion of the total grain separation is done at the threshing cylinder, and a high cylinder separation efficiency generally translates into higher separation and cleaning efficiencies of the combine. Cylinder separation efficiency varies from 60% to 90%.

Grain damage refers to mechanical damage to grain during the threshing process. It includes broken kernels, kernels with skin damage, and kernels with internal damage. Mechanical damage to grain results in poor germination, poor storability, and poor processing characteristics. Cylinder speed has the most profound effect on grain damage during threshing, as increasing cylinder speed increases damage exponentially (Srivastava et al., 2006).

Excessive straw breakup during threshing results in an increased load on the cleaning shoe, which causes additional cleaning losses. Straw breakup is caused by threshing with too much power (as a result of a too narrow concave opening for instance).

Separation After the threshing, a large number of grain kernels are still trapped within the straw flow. In a conventional combine the separation is performed by the straw walkers. For a rotary combine the separation occurs in the second half of the axial rotor. The rotor surface and the type of concaves are different in this part of the rotor. Grain kernels that exit the machine captured in the straw flow are called separation losses.

Straw walkers consist of several long channel sections mounted on a crankshaft. As the shaft turns, the channel sections follow an elliptical or circular path that causes the straw to bounce on top of the channels and move towards the rear of the combine. This oscillating action causes the grains and some chaff to be sifted down and separated from the straw. There are three to eight sections next to each other in a combine, depending upon its size. The contents sifted out of the walkers is thrown onto the cleaning system.

Rotary separators are based on an entirely different concept. The main force causing the grain to move through the mat of straw is the centrifugal force caused by rotation of the straw mat by the rotor. Note that for the straw walkers the separation is actually performed by gravity. The rotor, which makes the crop rotate inside the stationary cylindrical screen, generates a centrifugal force field which is several times that of gravity. The paddles mounted on the rotor surface cause the crop to take a helical path in the angular space defined by the rotor and the screen. In rotary separators the crop motion is forced rather than induced (as in the case of straw walkers). This results in higher capacity per unit grate area, but requires higher power. Since the separation is not gravity dependent, irregularity of the ground surface has no effect on the separation process. The rotary separator is on the same axis as the threshing drum, and thus turns at the same speed.

Cleaning Cleaning refers to the final separation of grain from other crop material, which consists mainly of chaff and broken straw pieces. The grains separated at the threshing cylinder and at the separation unit are combined on an oscillating conveyor or a set of augers that feed the mixture of grain and chaff to the cleaner, often referred to as the cleaning shoe. A common cleaning shoe arrangement is shown in Fig. 3.7. The separation is accomplished due to aerodynamic and mechanical actions. The cleaning shoe design consists of two or three oscillating adjustable-opening sieves and

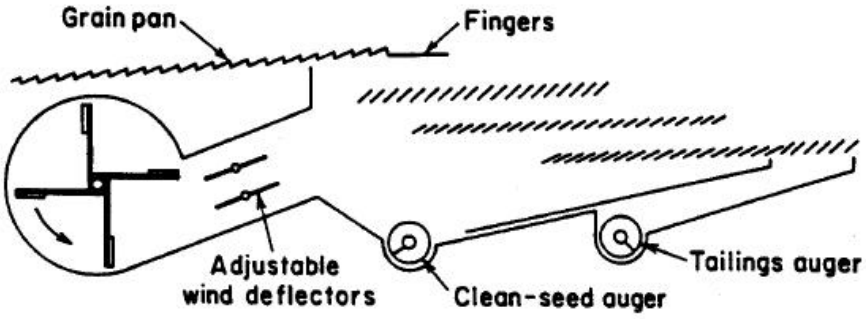


Figure 3.7: A schematic diagram of a cleaning shoe showing an auger bed for feeding the grain-chaff mixture. This picture shows three sieves on top of each other (Srivastava et al., 2006).

a paddle-type fan to blow air through the sieve openings. The crop is dropped on the top sieve (chaffer sieve) near the front of the shoe. The chaff gets blown off by the air and the grain falls through the openings onto the lower sieve (cleaning sieve). Here this process is repeated once more. The clean grain falling through this sieve is transported to the clean grain auger and conveyed to the grain tank. The separation occurs due to difference in the terminal velocities of grain and chaff material. This pneumatic separation is efficient because the terminal velocity of wheat, oat, and barley grains range from 5 to 10 m/s, whereas the terminal velocity for short pieces of straw ranges from 2 to 6 m/s and for chaff from 1.5 to 2.5 m/s.

The unthreshed grain which falls through the top sieve, but is too large to go through the lower sieve and too heavy to be blown off by the fan is commonly referred to as return flow or tailings. The tailings travel on top of the lower sieve towards the rear of the combine due to the oscillations and are collected by an auger and conveyed to the threshing cylinder for rethreshing. The kernels that do not fall through the top sieve exit the machine, and are called sieve losses. Rotary combines utilize the same cleaning shoe design as conventional combines. In some rotary combines there are augers placed longitudinally under the rotor to carry the grain-chaff mixture to an oscillating grain pan that feeds the mixture to the cleaning shoe. Some rotary designs create air flow through the rotor to remove chaff. This may be considered a form of pre-cleaning.

More information in the working principles of combine harvesters can be found in Quick and Buchele (1978) and Srivastava et al. (2006).

3.2.2 Sensors for system states

In order to extract information from the combine process, the machine needs to be equipped with different sensors. Signals related to the harvest rate, crop throughput, grain flow, losses and clean grain quality are required for capacity control. The actual measurement set-up used for the field experiments is detailed in Chapter 5.

Harvest rate

The instantaneous area rate s [ha/h] can be measured by estimating the actual cutting width w [m] and the machine speed v [km/h] (see below). The area rate directly determines the instantaneous crop throughput or harvest rate $\dot{m}_{ct}$ [t/h] via the local crop yield y_l [t/ha]:

$$\dot{m}_{ct}(t) \quad = \quad y_l(t)w(t)v(t) = y_l(t)s(t) \tag{3.1}$$

Note that the local yield $y_l(t)$ is unknown. Measurements of the instantaneous crop throughput are discussed in the following section.

Actual cutting width The cutting width w can be measured by an ultrasonic distance sensor in the crop dividers at both sides of the cutter bar (Vansichen and De Baerdemaeker, 1992). The instantaneous cutting width can also be entered by the operator in his console. Of course this value is only accurate if the header is row-dependent (for instance in corn). This operator input is already used to improve the quality of yield reporting.

Machine speed The rotation speed of the driving wheels is commonly used to estimate ground speed. The signal is however strongly affected by wheel slip phenomena and calibration errors. Alternative measurements with radar and GPS (not at low speeds) are possible.

Crop throughput

The total amount of crop material $\dot{m}_{ct}$ [t/h] entering the machine per hour is a mixture of grain kernels $\dot{m}^g_{ct}$ [t/h] and other material $\dot{m}^o_{ct}$ [t/h]. This crop throughput directly determines the work load of the harvester. For capacity control it is crucial to measure the crop throughput at an early stage of the process. There are several possibilities to measure throughput. Because of patent protection each major manufacturer has selected its own measurement methodology.

engine load: this is defined as the instantaneous fuel consumption. This signal is mainly affected by the driving torque on the header and separation drums, and offers a good indication of the immediate crop throughput. However, the influence of other parameters such as machine speed v, backward/forward machine inclination β_{bf} and local soil conditions is non-negligible and makes the engine load measurement unreliable as crop throughput sensor. The measurement is also sensitive to machine acceleration, especially with a full grain bin and/or uphill. The engine load signal can be used as a constraint to avoid machine overload in crops where engine power is the limiting factor (e.g. corn in humid circumstances).

driving torque on the auger and the straw elevator: this does not only include the torque which is necessary to transport the crop material from the feeding auger to the threshing drum (the torque driving the straw elevator), but additionally contains the driving torque on the feeding auger and the pick-up reel. As a result, a better signal-to-noise ratio is achieved than when only the torque in the

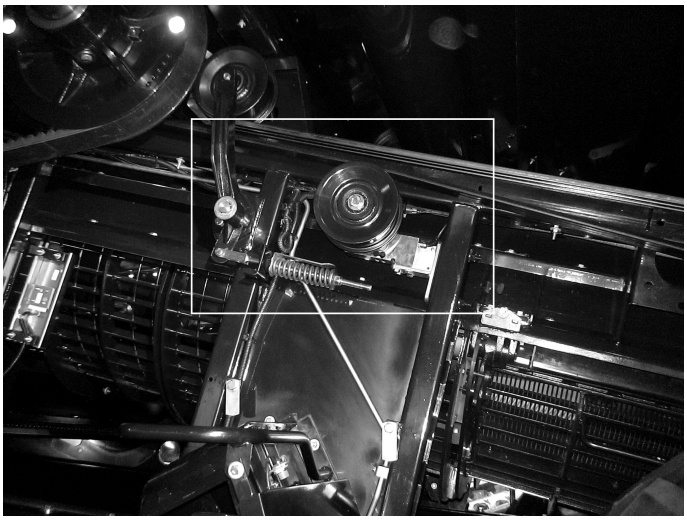


Figure 3.8: The feed rate sensor (marked by the white rectangle) as mounted on the New Holland CR combine harvester. This sensor measures the driving torque on the auger and the straw elevator (W, New Holland Benelux (2008)).

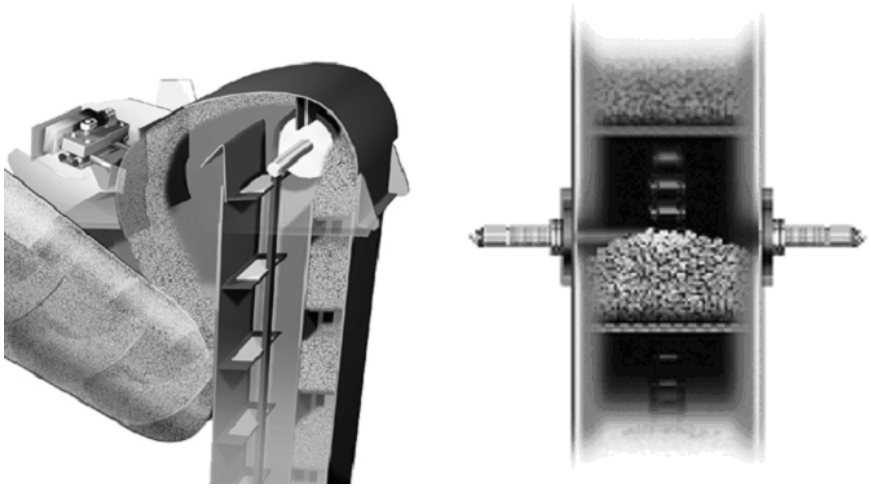
straw elevator is measured, especially during harvest in low yielding crops as canola or summer barley. This measurement yields the fastest response to crop density changes of any system on the market. (patented by Case New Holland (Strubbe and Missotten, 1999), Fig. 3.8)

straw elevator layer thickness: this measurement is also an indication of the crop flow in the machine. It responds quite fast to changes in crop density, but has a significantly slower response than the measurement of the driving torque on the auger and the straw elevator. Moreover, it is not suited for crops where the straw does not enter the machine, such as corn. (patented by Claas (Diekhans, 1998))

driving torque on the threshing cylinder: this was used in Eimer (1974) and Jensen (1988) as input for a ground speed control system. The main disturbances affecting this throughput measurement are the concave clearance, the threshing speed, the physical crop properties and the grain/Material Other than Grain (MOG) ratio. Another downside is the large delay between ground speed changes and threshing drum torque changes and the relatively low signal-to-noise ratio (patented by John Deere (Fechner et al., 2005)).

Crop moisture

The measurements for crop throughput, discussed in the previous section, measure the amount of biomass entering the combine. The mass of the crop varies with the moisture content. For some machine components the biomass may be the most relevant variable, for other components the amount of crop in a volumetric sense is more important. The clean grain flow for instance is more closely related to a volumetric crop



(a) (b)

Figure 3.9: (a) The grain mass flow sensor on a New Holland combine harvester (W, New Holland Benelux (2008)) which is based on a friction independent force measurement on a circular chute.
(b) The grain mass flow sensor on a Claas combine harvester (W, Claas (2008)) which measures the volume. This requires a regular calibration of the sensor since the crop density varies in time.

throughput measure, whereas for the straw elevator the required power depends on the amount of biomass. Moreover, crop moisture plays a significant role in the separation and cleaning processes.

Up till now, only capacitive sensors are available commercially to measure the moisture content of the clean grain material (Missotten, 1998, Van Loo, 1978). This capacitive measurement principle requires a frequent calibration, and is thus not always very accurate (Reyns, 2002). As a consequence, the signal should be considered as a relative indication for moisture content. However, research is being done to find a more robust measurement principle (Amoodeh et al., 2006).

Grain mass flow

Most commercial harvesters are equipped with a clean grain flow sensor for yield mapping purposes. There are different sensor types:

- torque-based sensors (Fig. 3.9 (a))
- impact plate-based sensors
- volume-based sensors (Fig. 3.9 (b))

The first two sensor types (torque-based and impact plate-based) are installed at the top of the clean grain elevator, and measure the clean grain throughput just before it enters the grain tank. The impact plate-based sensor measures the impact energy of the

grain mass flow on the sensor, whereas the torque-based sensor measures the torque on a turning plate. The torque-based sensor which is used in this research is designed in such a way that it is indifferent to crop type, moisture content and field slope with an accuracy of about 1% (Maertens and De Baerdemaeker, 2003, Anthonis et al., 2003, Missotten and Busschaert, 2003). This torque-based sensor outperforms the impact plate sensor (Burks et al., 2004) in a stationary test in the laboratory (Chaplin et al., 2004). A volume-based sensor measures the volume rate in the grain elevator with a laser eye. The main disadvantage of this measurement principle is that it assumes a constant density to convert the volume rate into a mass flow rate (W, Claas (2008)).

Losses

Losing grain at the end of the process means losing money. One of the Key Performance Indicators (KPIs) for a successful capacity control system is the loss level. The capacity control system has to make a trade-off between harvesting speed and, amongst others, losses, just like an operator would.

As mentioned in earlier sections, there are several types of losses, namely:

threshing losses: unthreshed grains that exit the machine with the straw

separation losses: the grain kernels that are not separated from the straw flow before the straw exits the machine

sieve losses (cleaning losses): the grain kernels that are not separated from the chaff-grain mixture at the end of the upper sieve

The measurement principle used for the last two types of losses is the same. After all, the objective of such a loss sensor is to detect grain kernels in a material stream exiting the machine. For threshing losses there is no sensor available at this time.

Measurement principle The kernels are detected using an impact sensor. The impact sensor consists of a metal plate with an accelerometer. A heterogenous flow of kernels, straw, chaff and other particles falls on the sensor plate and excites the accelerometer. A band pass filter is used to discriminate between impacts from grains and signal peaks caused by other particles (Strubbe, 1991, Diekhans and Behnke, 1990). The number of detected impacts during a fixed time interval is used as a measure for the loss flow.

Trade-off between saturation and variability The quality of loss sensors is mainly distorted by two phenomena originating from the impact-based measurement principle:

- a minimal detection time is necessary to register a kernel impact. Multiple impacts during one detection interval will be counted as one single impact and *saturation phenomena* will arise when dealing with high impact rates. A default impact detection time of 1 ms is chosen in the current generation of loss sensors.
- only one grain kernel impact can be registered every ms. Consequently, high frequency *discretization noise* will be important when harvesting at a low grain

loss level or when using short measurement intervals. The length of this measurement interval cannot be raised as one would wish, since the same interval determines the delay on the loss signal. In practice, a measurement interval of 200 ms is used to generate good quality loss signals with a sufficiently fast response for grain loss control purposes.

The quality of the loss signal is a serious issue for a loss control system. Research is being done to improve the loss sensor. One of the approaches is to work energy-based (estimate the kernel weight) rather than impact-based (estimate the number of kernels) (Coen, 2008). The characteristics of the loss signal are further examined in Chapter 6.

Commercial grain loss sensors Two grain loss signals are currently measured on combine harvesters by means of the impact-type sensors. The relative magnitude of both loss indications depends on the local crop conditions, harvest rate and applied machine settings. The current sensor locations are:

- two short impact sensors are installed at the end of the rotors to estimate the amount of residual grain kernels that is hidden between the larger straw particles (*separation losses*).
- one impact sensor is installed across the full width of the machine at the end of the cleaning section. This sensor detects kernels between the shaking layer of chaff and short straw ends (*cleaning losses*).

Clean grain quality

The price of delivered grain is not only determined by the moisture content, but also by the product quality q, a function of parameters such as hectoliter weight, MOG content (Material Other than Grain) and the amount of broken kernels. Recently a sensor was designed which enables online measurements of the MOG content during harvest (Wallays, 2008).

3.2.3 Recent developments

A good indication of the recent developments in combine automation are the gold and silver medal awards for innovations at the two-yearly fair for agricultural machinery, Agritechnica (W, Agritechnica (2007)). At Agritechnica 2007 following gold and silver medals were accorded to innovations on combine harvesters:

- ActiveTrac all-wheel drive: this is a slip control system for combine harvesters (silver medal, W, Claas (2008)).
- Multi-cylinder straw walker-material flow concept: a novel separation technique to enable higher machine capacities (silver medal, W, John Deere (2008)).
- Opti-clean cleaning system: a novel cleaning technology to boost capacity (silver medal, W, New Holland Benelux (2008)).

- IntelliCruise throughput regulator: a feed rate control system (silver medal, W, New Holland Benelux (2008); Coen (2008); Coen and Missotten (2007)).

- Grain Cam broken grain and MOG sensor: a sensor to quantify the quality of the grain transported to the grain bin. This is an important step in machine automation (gold medal, W, New Holland Benelux (2008); Wallays (2008)).

A gold medal is accorded to new features and a silver medal is accorded to an innovation of an existing feature.

Feed rate control Feed rate control, and more in general capacity control, has been a hot topic for some years now. All major manufacturers have put some solution on the market (Bussmann and Diekhans, 2007, Herlitzius and Boesch, 2004, Baumgarten et al., 2005, Craig, 2005), with varying degree of success. Since the first generation of feed rate control systems focuses on crop throughput, and does not yet take clean grain flow or losses into account, the most important part of the control system is the sensor.

As mentioned above, the feed rate sensor used by Case New Holland, which is also used in this research, is based on a torque measurement of the auger and the straw elevator (Strubbe and Missotten, 1999).

Guidance All manufacturers have brought one or more guidance systems on the market. There are three main types of combine guidance:

row guidance: for row crops such as corn, based on sensors on the header.

laser guidance: for small grains such as wheat and barley, a laser scanner is used to detect the edge of the crop (Diekhans, 2003).

GPS guidance: for all crops, a track parallel to the previous trajectory is followed based on GPS measurements. (Pangels et al., 1998).

Cleaning Automation A lot of work has also been dedicated to the automation of the cleaning system (Craessaerts et al., 2008). A MOG sensor, which quantifies the quality of the clean grain flow, is now commercially available (Missotten et al., 2006, Wallays and Missotten, 2008a,b). Based on this sensor a further automation of the cleaning system has become possible.

3.2.4 Road map for combine automation

In Fig. 3.10 the combine harvester development road map for one of the major manufacturers, John Deere (W, John Deere (2008)), is shown.

The final objective of John Deere, and of the combine harvester industry in general, is a fully autonomous combine. This is especially important for the large farms e.g. those in North America and France. These fields are often dozens of hectares large, and are harvested by multiple combine harvesters, sometimes even driving next to each other. These autonomous combines would clearly increase the overall efficiency:

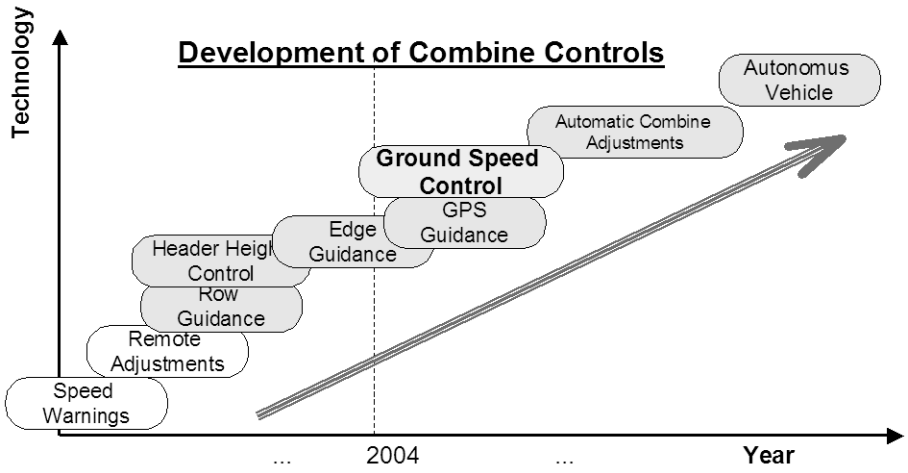


Figure 3.10: Road map for combine automation by John Deere. (W, John Deere (2008), from (Herlitzius and Boesch, 2004))

- if several combines are following each other (in separate tracks parallel to each other), it should be possible to let them operate with one operator for the entire fleet (possibly sitting at the side of the field), instead of one operator per combine.
- the combine harvester temporarily stores the grain in its own grain bin, but for these large fields unloading all the combines and getting the grain to the trucks at the side of the field becomes quite a logistic problem. This could clearly become much more efficient by controlling the combines automatically.
- if the operator is replaced by a controller, this controller would perform at least as well as the best operator, and thus much better than the average operator.

The most logical path towards an autonomous combine harvester is to automate the operator tasks, starting with the most demanding tasks. Some of these tasks have already been tackled in research. In order of descending resource requirement, the most important operator tasks are:

1. controlling the header height (Lopes et al., 2002)
2. keeping the combine on the right track (the right row in corn, along the edge in grain) (Chateau et al., 2000, Coen et al., 2008c)
3. **modifying the combine harvester speed to run the combine at full capacity** (Bussmann and Diekhans, 2007, Herlitzius and Boesch, 2004, Coen et al., 2007b)
4. unloading the grain bin in a lorry (Behnke et al., 2002)
5. adjusting the sieve settings to ensure that the grain in the bin is clean, and that the losses or return flow do not become too high (Craessaerts et al., 2007a,b, 2008)

6. adjusting the threshing parameters to avoid under- or overthreshing (Behnke, 2002, Spokas et al., 2007)

7. making the combine autonomous (path planning,. . .) (Wilson, 2000, Murakami et al., 2008, Coen et al., 2008c)

In this PhD the focus lies on step 3, since this was the next step at the start of this research. Note that these steps can also be found in a similar order on the combine automation road map of John Deere. Because of the recent food crisis, there is a clear demand to check the quality of the harvested product already in the field, and to ensure traceability of the crop from the field to the plate of the consumer (De Castro, 2002, Sarig, 2003). This traceability requirement will in the future also be added to the combine harvester road map.

3.3 Conclusion

The automation of agricultural machinery is driven by environmental, social, economical and commercial aspects. Although the farms are becoming larger, there are fewer people working on each farm. This makes it increasingly difficult to find experienced personnel for specialized machinery. The higher product prices, and the evolution that agricultural products are also used for other purposes than food, make larger investment budgets available. All these effects together give a boost to automation in agriculture, which is also confirmed by recent sales figures of manufacturers.

This chapter also discusses the working principle of a combine harvester, and indicates which process variables can be measured. The most important process variable for capacity control is feed rate, which is a measurement for the amount of biomass entering the combine harvester. There are several sensors which measure this biomass rate, but the torque measurement on the auger and the straw elevator which is used by CNH has the best properties. This sensor is also used on the field test machine (see Chapter 5).

The ultimate objective of combine harvester automation is the design of an autonomous combine. This is especially economically relevant for the large fields in France and North America. The next logical step on this combine automation road map is capacity control. After all, since it is no longer possible to enlarge combine harvesters without losing the capability of driving on normal roads, the only way to increase the capacity is to improve the machine efficiency.

Chapter 4

Research objectives

Obstacles are those frightful things you see
when you take your eyes off your goal
-Henry Ford-



In the previous chapters the state of the art in MPC and agricultural machinery has been described. The purpose of this chapter is to formulate the research objectives. These objectives combine both domains which are at the foundation of this thesis.

As the title of this work states, the overall objective is to increase the capacity of harvesting machinery, and more specifically a combine harvester by using MPC. In Chapter 1 the main objective was already formulated as follows:

> **Objective.** *To design a practical capacity control system that copes with model uncertainty in an optimal manner, making use of the Stochastic Model-based Predictive Control framework (Stochastic MPC). It has to be possible to implement the controller on standard processing hardware available on combine harvesters with an acceptable effort. The methodology has to be validated on another application.*

This main objective can be split up in the following research objectives:

- a time-varying system model for capacity control
- a high performance capacity control system
- a control system which is robust to process variations
- implementation of the approach on the real-life system
- validation on other agricultural applications

These research objectives are further elaborated in the following sections.

4.1 System model

Because of the complex nature of the combine harvester and the harvesting process, the system model will have to be time-varying. After all, the model needs to be able to deal with the following effects:

linearization error: nonlinear effects cause the model to be dependent on the working point. If the working point changes, the parameters of the linear model have to change as well to minimize the linearization error.

variable crop properties: the crop properties vary continuously across the field and influence the combine harvester process.

unmodeled effects: the terrain condition and slope may influence the combine harvester model.

The objective in this research is to define a time-varying model, and to examine the effect of some disturbances to get an idea of the model reliability. The stochastic approach of the controller (Objective 4.3) requires that the parameter error is normally distributed.

Research objective 4.1. *Design a time-varying model which can follow the process variations such that the parameter error is normally distributed.*

4.2 Capacity control

The main application throughout this PhD is capacity control for a combine harvester. This focus on capacity control can be motivated by the development road map of section 3.2.4. Capacity control is the next step towards a fully autonomous machine from the current state of the art. Moreover, the capacity of a combine can no longer be increased by building larger machines due to road restrictions (Kutzbach, 2000), which means that the only option is to automate the machine in order to further increase the capacity. Thus, capacity control is high on the agenda of every combine manufacturer.

MPC (see Chapter 2) is very well suited for this application. MPC can deal with large time delays, and is able to accommodate state and input constraints. Moreover, the controller objective can easily be changed online. This means that only one controller needs to be maintained for all the different scenarios.

4.2.1 Control objectives

The control objective is to maximize the capacity utilization of the combine harvester which may be expressed in terms of crop throughput, grain mass flow or speed, taking into account some constraints. The control objective of a capacity control system depends on the circumstances. Two possible scenarios are:

a contractor: he has to satisfy certain requirements of the grain dealer and the farmer. The objective here would be to maximize the speed, taking into account the machine integrity and comfort constraints and the requirements of the grain dealer. Requirements of the grain dealer and the farmer will be expressed in terms of losses, MOG and broken grain.

a farmer on flat terrain: the objective would be to drive on a feed rate or grain mass flow set point, respecting the machine integrity and comfort constraints. A farmer normally does not want to drive against the constraints, but simply at some constant capacity utilization.

4.2.2 Operation constraints

A number of constraints has to be satisfied at all times. There are two main constraint categories:

comfort constraints: limit the machine inputs and variables to guarantee a safe and comfortable ride for the operator.

machine integrity constraints: limit certain process variables to avoid damage to the machine. Constraints can for instance be placed on engine load, feed rate and ground speed.

4.2.3 Benefits

The most important benefits of capacity control are:

capacity gain: the controller will yield a performance equivalent to that of the best possible performance by an operator. This is a clear improvement compared to the average performance of an operator. After all, even the performance of the best operator will degrade in the course of a long working day, whereas the performance of the control system remains constant.

operator load: the operator will have more time to pay attention to other tasks, for example the arrangement of the grain transport.

operator profile: it becomes increasingly hard to find skilled workers which can operate a combine close to its limit. Since most operators are only required during the (short) field season, it is economically not interesting to invest in training.

4.2.4 Assessment

The controller performance is assessed relative to several criteria, of which the most important ones are:

input effort: the input changes the controller makes. The input effort is split into a low-frequency part and a high-frequency part. The operator comfort is related to the high-frequency input effort.

control error: the error on the set point. The control error is also split into a low- and a high-frequency part. The high-frequency error is caused by noise, and is thus irrelevant. The low-frequency error is an interesting measure for the controller performance.

A good controller has a minimal high-frequency input effort and a minimal low-frequency control error. More details about the controller performance measures can be found in Appendix D.

Research objective 4.2. *Design a capacity control system with MPC which can control the feed rate, engine load and the grain mass flow to the desired set points, while respecting the constraints on the inputs and the states.*

4.3 Model and process uncertainty

As already discussed in Chapter 3, the model of the combine process strongly depends on the properties of the feed. This is typical for all systems that process material with varying properties. The behavior of a roaster oven burning waste for instance, is also influenced by the composition of the waste.

In this research, the strategy to deal with uncertainty is based on the structure and the requirements of the main application, namely capacity control on the combine harvester. Later on it will be shown that the developed methodology can be generalized. For the capacity control application, three types of model error can be considered:

slow drift: the process parameters slowly vary in time. For instance the crop density fluctuates in a continuous manner across the field.

input induced error: the slow drift of the process parameters and the nonlinearities in the system, imply that the linear adaptive model is never exact. A model error will occur each time the input is changed.

step disturbances: due to terrain conditions (mud, change of slope, switching on/off of the 4×4, etc.), process variables may suddenly increase or decrease.

Slow drift of the parameters is dealt with by the adaptive model. Moreover, since the controller uses a relative model (realigned model), slow drift of the model parameters does not affect the controller performance. The input induced error, however, plays an important role for this application. In order for the controller to respond as fast as possible at all times, this input induced error needs to be taken into account by the controller. Robustness to step disturbances is only possible by adding conservatism to the controller, for instance by keeping a certain back-off from constraints at all times. For this application it is important that the controller can correct the disturbance if it occurs, but it is not necessary to guarantee that the constraints are robust to this type of disturbance.

Based on the parameter uncertainty the state uncertainty caused by input changes needs to be predicted. The state uncertainty thus depends on the input. Although most treatments in the literature make use of additive uncertainty, multiplicative uncertainty is more natural for this application.

In this research the model uncertainty will be taken into account by the MPC to tune the objective function online. A good process model allows for swift, aggressive control, whilst a bad model quality requires a slow, careful response to avoid instability.

Research objective 4.3. *Predict the state uncertainty caused by input changes online. Design a controller such that the controller performance is always as good as the model quality allows by using the Stochastic MPC framework.*

4.4 Implementation

An important aspect of this PhD is that the actual controller needs to be implemented on a real-life machine. The field experiments are performed with a PXI system (graphical programming in LabView), but the system should also be easy to implement on the standard hardware used on the commercial combine harvester (so-called CCMs or Combine Control Modules, see Chapter 5).

> **Research objective 4.4.** *Implement (Stochastic) MPC on a real-life machine subject to the existing hardware and software constraints. The implementation needs to be portable to the CCMs.*

4.5 Validation of the methodology

The methodology of this PhD, and especially the approach to Stochastic MPC needs to be validated on a second application to prove its benefits. For this purpose two applications are selected, namely cruise control on the combine harvester and bale density control on a square baler. Cruise control is developed for driving on the road, which allows for control of both the pump setting and the engine speed of the diesel engine. The cruise control application makes use of classic (Nonlinear) MPC. The objective of bale density control is to keep the density of the bale constant by varying the pressure on the bale chamber. Since the baler process strongly depends on the biological properties of the crop, it is perfectly suited as a validation application of Stochastic MPC.

> **Research objective 4.5.** *Show that the methodology developed in this research, and especially the Stochastic MPC approach, also works on other applications than capacity control on the combine harvester.*

4.6 Conclusion

In this chapter the research objectives of this PhD have been made more concrete. The next step is to describe the experimental set-up on the combine harvester. Once that it is clear what is measured, we can move to modeling and control. In Chapter 6 a model for the capacity control system is designed (Objective 4.1). Chapter 7 discusses capacity control with classic MPC (Objective 4.2). Model uncertainty is introduced in the controller based on the Stochastic MPC methodology in Chapter 8 (Objective 4.3). An optimization method that satisfies the implementation constraints is designed in Chapter 9 (Objective 4.4). Chapter 10 and Chapter 11 present validation applications of respectively MPC and Stochastic MPC (Objective 4.5).

Chapter 5

Experimental set-up on the combine harvester

A theory is something nobody believes,
except the person who made it.
An experiment is something everybody believes,
except the person who made it.
-Albert Einstein-



Real-life implementation is an important aspect of this thesis. As stated before, the main application is the combine harvester. In order to to implement and test capacity controllers, we need to be able to measure the relevant process variables, and control the pump setting (which controls on its turn the speed).

This chapter presents the experimental set-up as it was used during the field tests. The sensors available on the test machine are described. The hardware and software used for the experiments is discussed. Finally some details on the field tests are given.

5.1 Combine harvester

For the field experiments of this PhD a New Holland CR combine harvester was used. This is a top of the line combine harvester from New Holland (W, Case New Holland (2008)) with an approximate value of EUR 200 000.

The CR960 combine harvester has an empty weight of about 15 tonnes, and a grain tank with a capacity of about 9 000 liters. Depending on the crop type, the grain tank can contain between 5 and 7 tonnes of grain. The combine harvester is driven by a diesel engine of nearly 400 hp. The engine speed of the diesel engine is allowed to vary between 1 300 and 2 100 rpm, as long as the harvesting mechanisms are turned off. Once the harvesting mechanisms, such as the threshing and the cleaning system, are engaged, the diesel engine has to run at the maximal engine speed.

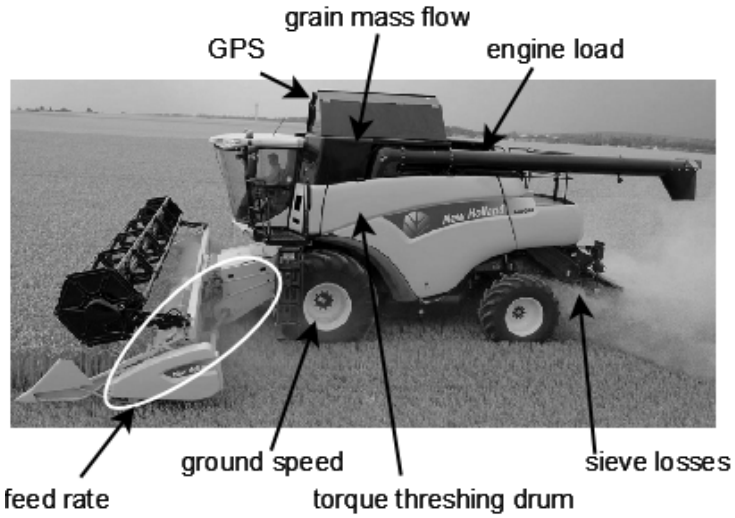


Figure 5.1: An overview of the sensors used in this research.

5.2 Sensor overview

A number of sensors were already discussed in section 3.2.2 of Chapter 3. This section summarizes the relevant sensors available on the test machine. The sensors are indicated on Fig. 5.1.

feed rate sensor: this sensor measures the crop throughput by measuring the torque on the straw elevator and the auger of the header.

ground speed: the ground speed is measured on the drive train of the front wheels. In muddy conditions this measurement may be inaccurate.

torque threshing drum: a throughput measurement, which is much less accurate than the feed rate sensor.

sieve losses: the sieve losses are measured where the chaff exits the machine. At this time the sensor is not sufficiently accurate to use as a control objective. It can however be used as a constraint to avoid too high losses.

engine load: the engine load can also serve as a crop throughput measure, but is strongly influenced by the propulsion system. Driving uphill/downhill or accelerations considerably affect the engine load. The engine load can be used as an objective or constraint in conditions where the engine power is the limiting factor. This happens for instance in corn (especially on slopes).

grain mass flow: the grain mass flow is measured at the top of the grain elevator, just before the grain enters the grain bin with a so-called torque-based sensor (section 3.2.2 of Chapter 3). Because of the sensor location at the top of the elevator, there is a large delay between a speed change and a change in grain mass flow.

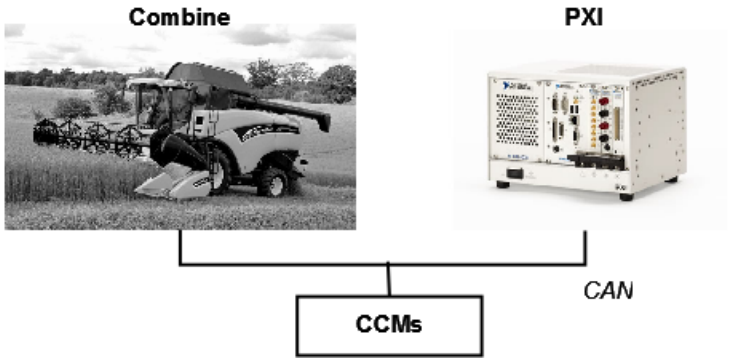


Figure 5.2: The measurement set-up. The PXI system is connected to the CCMs and the combine harvester sensors through the CAN network.

GPS: The combine is equipped with a GPS system from Trimble (W, Trimble (2008)). The location and the altitude of the combine harvester as well as the feed rate, grain mass flow and other relevant process variables are logged.

5.3 Measurement set-up

The brains of the combine harvester are the CCMs or Combine Control Modules. These are the processors which control all the functionality of the combine harvester. Depending on the purchased features two or even three CCMs are installed. All actuators and sensors are either connected to the CAN network or directly to the CCMs.

As measurement and control platform a PXI system from National Instruments (W, National Instruments (2008)) is used. This platform runs LabView RT as operating system, which is a stripped version of Microsoft Windows. More details about this platform and its performance can be found in Appendix A. The PXI system is actually an industrial computer, which can be programmed in LabView. The PXI can connect directly to the CAN network. Specifically for this research the combine manufacturer has created a protocol in the CCM software which makes it possible to control for instance the combine speed from an external system such as the PXI. An overview of the measurement set-up is given in Fig. 5.2.

The measurements are sampled at 5 Hz for control purposes or at 20 Hz for system identification of the propulsion system. Most of the sensors however, do not provide new values faster than at 5 Hz. Only the ground speed and the engine speed are available at 20 Hz.

The models and the controllers described in the following chapters are implemented in LabView, and run on the PXI system. In order to be able to guarantee real-time performance, the program structure is extremely important. More details about the software written for the experiments is available in Appendix B.

5.4 Field tests

All the field tests on the combine harvester are performed in the wide neighborhood of Leuven. The combine harvester test results come from four test seasons: 2004, 2005, 2006 and 2007. Most of the tests were in corn, but barley and wheat experiments were also performed. Corn is the ideal crop for capacity control experiments because:

- it has the longest test season (four to six weeks).
- corn can be harvested in humid circumstances. The corn kernels themselves are very well shielded from the rain. Thus there is less time pressure caused by changing weather conditions.
- corn does not need to be dry, a corn harvest day is much longer - and thus allows far more tests - than a wheat harvest day. Whereas corn can be harvested 24/7, wheat harvest in Belgium only starts around two o'clock in the afternoon.

Most of the experiments on the square baler have been performed abroad, namely in France, Scotland and Spain in the summers of 2007 and 2008, mostly on wheat straw.

Throughout the text is indicated which field test (day and filename) contains the experiment at hand. These references are preceded by 'F'. The field test reference list can be found in Appendix G, on page 245.

5.5 Conclusion

A New Holland CR combine harvester will be used for the system modeling and capacity control experiments. On this machine sensors are available to measure the feed rate, the ground speed, the torque of the threshing drum, the sieve losses, the engine load, the grain mass flow and the GPS position. This machine is also used for the cruise control experiments in Chapter 10.

The logging and the controllers are implemented in LabView, and run on a PXI platform. This PXI system is connected to the combine through the CAN network, which enables us to control the machine speed. Most of the experiments are performed in corn because this crop is best suited to perform extensive field tests in Belgium.

Now we know what data is available and in what form, we can move to explore the system in order to derive a model suited for control.

Chapter 6

System modeling

If you are out to describe the truth,
leave elegance to the tailor
-Albert Einstein-



Capacity control on the combine harvester is the main application of this research. This chapter presents the methodology which was followed to derive the model of the combine harvester used for control. This chapter is the first of three pillars to arrive at real-life applications. Although this chapter is applied to the combine harvester, it can be seen as a general step by step overview of the procedure used to model a time-varying system for MPC.

First the process structure is presented based on expert knowledge. This section describes which variables are related. Then the dynamics of these relationships are modeled. Taking these dynamics into account, the variation of the delays is quantified.

A model is always an approximation of reality, where only the most important effects are modeled. Choosing to model an effect is a trade-off between model performance (how much improvement can be obtained by modeling the effect) and model complexity (how complex is it to model this effect). Moreover, it is not always feasible or economically desirable to extend the model. For this application, several disturbance effects are shown and analyzed. This also serves to give an idea of the complexity of the system.

The model parameters are time-varying, and depend on the crop and field conditions. The parameters thus need to be estimated online. The variation of the instantaneous parameter values across a barley field is discussed, and different estimation techniques for the parameter values are compared.

At the end of the chapter the model of the combine harvester used for capacity control is presented.

6.1 Process structure

The purpose of this section is to define the relationships between the process variables. How this relation is best modeled, for instance by a simple gain factor or a dynamic

model, is examined in the following sections. For the combine harvesting process, the relevant process variables are (see Chapter 3):

- pump setting (*PS*)
- ground speed (*GS*)
- feed rate (*FR*)
- engine load (*EL*)
- torque of the threshing drum (*TD*)
- grain mass flow (*GMF*)
- sieve losses (*SL*)
- return flow (*RF*)

Since the engine speed of the diesel engine is constant during harvesting operation, the ground speed is fully determined by the pump setting. Feed rate, engine load and torque of the threshing drum are all related to the amount of crop entering the machine, thus these variables are related to the ground speed. This relation will of course be influenced by the crop density, the crop type, etc. For engine load there are some extra influence factors, such as the mass in the grain bin and the terrain slope.

Grain mass flow is proportional to the amount of crop entering the machine. This crop flow can be quantified by feed rate, engine load or ground speed. The torque of the threshing drum also indicates the biomass flow rate entering the machine, but is a less suitable measure because of its low signal to noise ratio (section 3.2.2), and will thus not be used to predict other process variables.

The sieve losses and the return flow are determined by the load on the cleaning section. On the machine used for these experiments feed rate and engine load are the best indicators for this load. Recently new sensors for the cleaning section have been proposed by Craessaerts (2008) to measure the load on the cleaning section. These sensors were not yet available for this research.

The process structure is illustrated in Fig. 6.1. Note that grain mass flow, sieve losses and return flow depend on several process variables. Since feed rate and engine load depend on ground speed, it makes no sense to define a grain mass flow model based on feed rate, engine load *and* ground speed. The gain between ground speed and feed rate or between ground speed and engine load, and the gain between feed rate and grain mass flow or between engine load and grain mass flow are however two very different properties. The gain between ground speed and feed rate or between ground speed and engine load is a measure of the amount of biomass that is entering the machine. The gain between feed rate and grain mass flow or between engine load and grain mass flow is a measure for the grain-to-straw ratio of the crop. If these parameters need to be determined online, the results will improve considerably by separating both biological effects. The grain-to-straw ratio will on average vary slower than the crop density, which can change in steps (by starting a different part of the field). Due to the smaller time delay between ground speed and feed rate (or ground speed and engine load) than between ground speed and grain mass flow, the lag between the field condition and the estimated condition will also be much smaller for

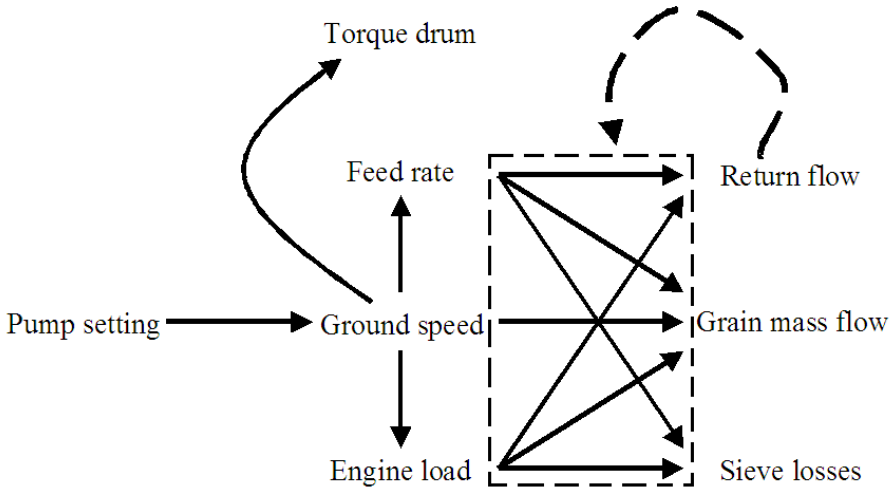


Figure 6.1: Overview of the relations between the different process variables. The dotted arrow indicates that the return flow is injected back into the system. This means that the return flow level influences the relations between feed rate, ground speed and engine load on one hand, and grain mass flow and sieve losses on the other hand.

the ground speed-feed rate or ground speed-engine load estimator than for the ground speed-grain mass flow estimator. Thus it makes sense to split up the estimator, and predict grain mass flow with feed rate or engine load.

The direct relation between ground speed and grain mass flow can be interesting in special circumstances during corn harvest. If the natural variation of feed rate and engine load becomes too large, the models based on these variables will have a hard time to predict grain mass flow. In these cases, it is good to have a back-up model for grain mass flow based on ground speed that can take over. The procedure to merge the different models for grain mass flow will be discussed in the chapter on Stochastic MPC (Chapter 8).

6.2 System identification

The controller needs a model with good prediction behavior, and which has a good performance on the most common control inputs, namely step or slope inputs. Since the controller operates in the time domain, time domain techniques are best suited to model the dynamics. A block wave is applied to the pump setting (F, 20051121a) in corn to measure the step response of the system. As mentioned earlier, delay plays an important role for this application. Where necessary (towards feed rate, engine load and grain mass flow), the model is calculated for different values of the delay. The delay which leads to the best model performance is selected. In the next section the variation of this delay in a field is examined.

First of all, the step responses of the process variables are compared. The variables are then modeled according to the relations shown in Fig. 6.1. Since for the

controller only pump setting, feed rate, ground speed, engine load and grain mass flow are relevant, this section focuses on modeling those relations.

Ground speed is modeled based on the pump setting. In Fig. 6.2 (a) the ground speed is modeled as a gain times the pump setting. For step responses this suffices. Thus, although the relation does contain dynamics, these do not need to be modeled to get a good prediction of step responses. Of course this depends on the expected speed range and the used sampling frequency. For the cruise control application discussed in Chapter 10, the dynamics do need to be modeled. Whereas for capacity control the speed remains smaller than 10 km/h, the cruise control system has to deal with speeds up to 40 km/h. Moreover, the dynamics of the system are also more excited by varying both the pump setting and the engine speed. This is why the sampling rate of the controller is 20 Hz instead of the 5 Hz which will be used for capacity control. During harvesting operation the engine speed of the diesel engine is fixed to the maximal engine speed.

The step response of feed rate (Fig. 6.2 (b)) can be modeled as a gain times the ground speed (combined with a delay of 0.6 seconds).

Engine load (Fig. 6.2 (c)) and grain mass flow (Fig. 6.2 (d)) however, do show significant dynamics. Engine load exhibits a peak at each step response (Fig. 6.4). This peak is caused by the propulsion system. At the start of the acceleration the engine speed drops, which is then compensated by the embedded controller (ECU). This causes a peak in the engine load, which is measured as the fuel injection in the diesel engine. The purpose of the dynamic model is only to model the part of the response which is caused by the combine process. The short peak is not relevant for the capacity controller. After all, the capacity controller uses engine load as a capacity measure. The engine load (*EL*) can be predicted by the following model (discrete time, sampling rate 5 Hz):

$$EL_{k+1} \quad = \quad (1-\alpha) \cdot EL_k + \alpha \cdot K_{EL} \cdot GS_k \tag{6.1}$$

where $\alpha = 0.1036$ is a model parameter, calculated on the step response, and K_{EL} is a gain parameter. The optimal delay is 0.4 seconds. This value of α corresponds to a cut-off frequency of $f_{C,\alpha}$=0.087 Hz. The step response of this model is shown in Fig. 6.2 (e). Gain parameters are assumed known in this section. The calculation of these parameters is more extensively discussed in section 6.5.

The dynamics of the rising and the falling edge of grain mass flow differ. This is discussed further on in section 6.4.4. Since especially the rising edge is important for control purposes (each crop entry), the model focuses on the rising edge. Let us first consider the relation between feed rate and grain mass flow. Grain mass flow is predicted using a first order model like Eqn (6.1):

$$GMF_{k+1} \quad = \quad (1-\beta) \cdot GMF_k + \beta \cdot K_{GMF-FR} \cdot FR_k \tag{6.2}$$

with $\beta = 0.0236$ which corresponds to a cut-off frequency of $f_{C,\beta}$=0.019 Hz. The optimal delay is 10.2 seconds. This is illustrated in Fig. 6.2 (f). The relation between engine load and grain mass flow is analogous, and can be modeled as:

$$GMF_{k+1} \quad = \quad (1-\gamma) \cdot GMF_k + \gamma \cdot K_{GMF-EL} \cdot EL_k \tag{6.3}$$

with $\gamma = 0.0320$ which corresponds to a cut-off frequency of $f_{C,\gamma}$=0.026 Hz. The optimal delay is 10.0 seconds. In the next section, to examine the delay variation, a

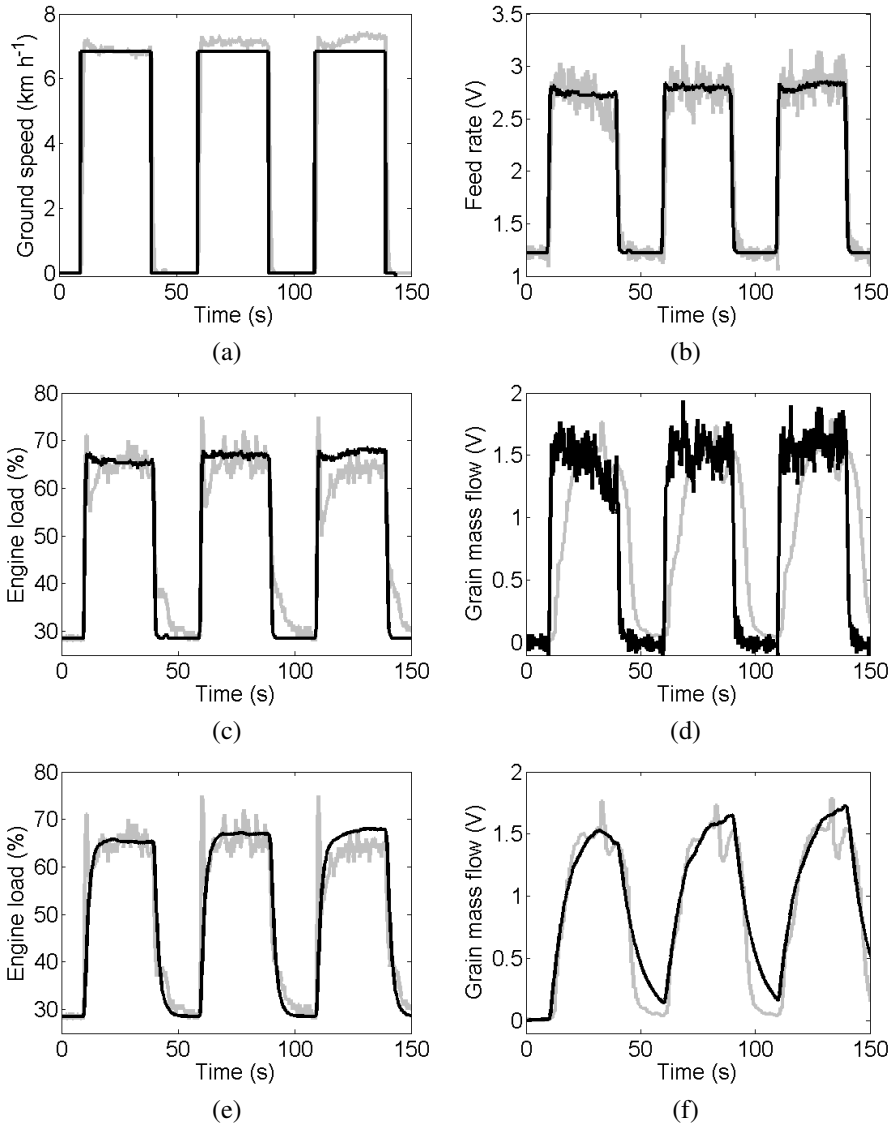


Figure 6.2: (a) Ground speed (grey line) and rescaled pump setting (black line) (b) Feed rate (grey line) and rescaled ground speed (black line) (c) Engine load (grey line) and rescaled ground speed (black line) (d) Grain mass flow (grey line) and rescaled feed rate (black line) (e) Engine load (grey line) and modeled engine load (black line) (f) Grain mass flow (grey line) and modeled grain mass flow (black line).

Table 6.1: Delays (in seconds) for wheat (F, 20050810b) and corn (F, 20051121b) where '*_M*' indicates that the process variable is first filtered with the process model. LB and UB denote respectively the lower and upper bound of the 95% confidence interval around the difference of the wheat and the corn delay. p is the probability that the delays are different for both crops.

	WHEAT		CORN		ANOVA		
	Mean	Std	Mean	Std	LB	UB	p
GS→FR	1.6	0.23	0.6	0.09	0.62	1.17	2.0e-4
GS_M→EL	1.0	0.53	0	0.00	3.15	3.39	7.4e-10
GS→TD	3.4	0.12	5.4	0.17	-5.04	-3.84	1.8e-6
GS_M→GMF	11.6	0.53	10.2	0.38	0.58	2.14	5.4e-3
FR_M→GMF	10.4	0.23	9.2	1.28	-0.62	3.15	1.5e-1
EL_M→GMF	10.0	0.50	8.6	1.21	-0.34	3.35	9.2e-2
	3 runs		5 runs				

model between ground speed and grain mass flow is also used to exclude the influence of the dynamics on the delay calculation. The relation between ground speed and grain mass flow can be modeled as:

$$GMF_{k+1} = (1-\delta)\cdot GS_k + \delta \cdot K_{GMF-GS}\cdot EL_k \tag{6.4}$$

with $\delta = 0.0267$ which corresponds to a cut-off frequency of $f_{C,\delta}$=0.022 Hz. The optimal delay is 11.8 seconds.

For wheat the values of the parameters calculated in this section are slightly different:

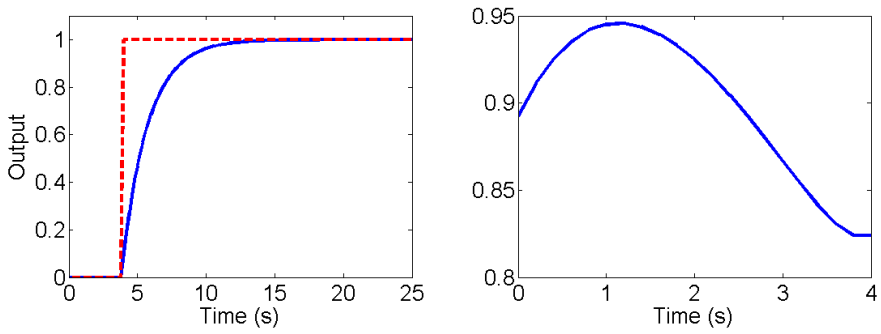
$$\begin{aligned} \alpha &= 0.0968 \quad f_{C,\alpha} = 0.081 \text{ Hz} \\ \beta &= 0.0410 \quad f_{C,\beta} = 0.033 \text{ Hz} \\ \gamma &= 0.0496 \quad f_{C,\gamma} = 0.041 \text{ Hz} \\ \delta &= 0.0295 \quad f_{C,\delta} = 0.024 \text{ Hz} \end{aligned} \tag{6.5}$$

6.3 Delay variation

In the previous section the optimal delay was determined by selecting the delay which leads to the best model performance. In this section the delay between the process variables is determined using correlation based on several step responses, and this for both wheat and corn. All measurements were performed at 5 Hz sampling rate.

As described in Chapter 3, a combine harvester operates completely different in corn than in wheat. In corn only the ear (the corn cup) enters the machine, whilst the 'straw' (the stem and the leaves) is chopped on the field by knifes mounted on the header. Thus one can expect significant differences between both situations.

The objective of this section is to quantify the delay and its variation, and to examine the difference in delay between wheat and corn.



(a) (b)

Figure 6.3: Calculation of the delay using correlation
(a) Two signals x (dashed line) and y (solid line). y is the model output of Eqn (6.1) with x as input.
(b) The correlation function between both signals. Maximum correlation occurs at a lag of 1.2 seconds.

The delay is determined using correlation. The correlation function $c(x, y, m)$ is calculated as follows:

$$c(x,y,m) = \frac{\sum_{k=1}^{N-m}\left(\left[x(k)-\frac{1}{N-m}\sum_{k=1}^{N-m}x(k)\right]\cdot\left[y(k+m)-\frac{1}{N-m}\sum_{k=1}^{N-m}y(k+m)\right]\right)}{\sqrt{\sum_{k=1}^{N-m}x^2(k)}\sqrt{\sum_{k=1}^{N-m}y^2(k+m)}} \tag{6.6}$$

where x an y represent the process variables between which the correlation is calculated. N measurements are available. The correlation function is determined for several lag values m, for example $0 \leq m \leq 100$. The delay between both variables is then the lag which maximizes the correlation function. This is a good way to calculate the delay if the dynamics of both signals x and y are similar.

The situation where the dynamics are not similar is illustrated in Fig. 6.3. Consider two signals x and y with a sampling rate of 5 Hz. x is a step input, y is the model output of Eqn (6.1) with $\alpha = 0.1$ and x as input. In reality, there is no delay between both signals. The correlation function of Eqn (6.6) however, indicates a delay of 1.2 seconds because of the difference in dynamic behavior between both signals. To avoid these errors due to a mismatch in dynamics, the models of the previous section (except for the delay) are used to preprocess the input signals x. This may lead to slightly different delays than obtained in the previous section.

The engine load signal is, as discussed in the previous section, influenced by the propulsion system. In order to determine the delay between the different signals, step inputs are applied to the pump setting of the machine. This leads to a maximal machine acceleration, which on its turn causes a peak on the engine load signal (Fig. 6.4). This has nothing to do with the combine harvesting process itself. A nonlinear

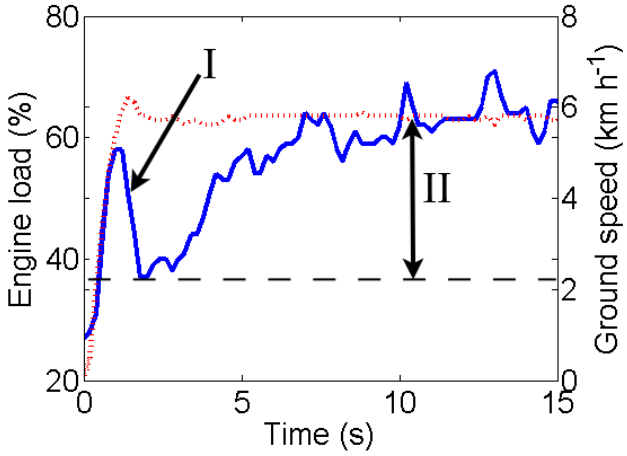


Figure 6.4: Engine load (solid line, left axis) and ground speed (dotted line, right axis) during a step response. The engine load depends on the propulsion system (I) and the crop throughput (II). Note the peak in engine load during acceleration (F, 20050810a).

filter is applied to the engine load signal to remove the peak. This is a filter of which the cut-off frequency depends on the rate of change of the signal. A peak in the engine load thus implies a stronger filter action. Removing this acceleration peak from the engine load signal, allows the delay to be determined more accurately.

In the previous section the dynamics for four relations were calculated, namely ground speed-engine load, ground speed-grain mass flow, feed rate-grain mass flow and engine load-grain mass flow, and this for both corn and wheat. These dynamics will be used in this section to model for instance engine load based on ground speed using Eqn (6.1). The variables which are replaced by the output of the appropriate model are marked with '$_M$'.

For this experiment several step responses are measured in wheat and in corn. All experiments were performed on flat terrain. The machine settings were the standard values for the crop type. The delays determined using correlation are shown in Table 6.1. All the relations shown in Fig. 6.1 are examined, except those containing sieve losses or return flow. These signals are too noisy to allow for an accurate calculation of the delay.

An ANOVA test (Neter et al., 1996) is performed to determine whether the difference in delay between both crops is significant or not. The 95% confidence interval around the difference of the wheat and corn delay is shown in Table 6.1, as well as the probability that the delay of both crops is equal. Note the difference in time delay between wheat and corn for the relation between ground speed on one hand and feed rate and engine load on the other hand. The delay on this relation is significantly larger in wheat than in corn. This can be explained by the presence of a chopper on the corn header. This chopper almost immediately makes contact with the crop, and thus leads to a very short delay at crop entry. The difference in delay on the relation between feed rate and engine load on one hand, and grain mass flow on the other hand is not statistically significant, but there is some trend visible. Note that in corn only the

cups enter the cleaning system, whereas in wheat also short straw will end up on the cleaning system. The more material-other-than-grain (MOG) that is on the cleaning system, the larger the time required for grain kernels to move through the cleaning system will become. The difference in delay is significant for the relation between ground speed and grain mass flow. After all this can be seen as the combination of for instance the ground speed-feed rate delay and the feed rate-grain mass flow delay.

The delay towards the torque of the threshing drum is also larger in corn than in wheat. This is probably related to the time needed for the header and the elevator to fill up with corn cups. In wheat this filling up the elevator is less an issue since the straw enters the machine as well, which results in a larger input volume, and thus a shorter transition phenomenon. Note that the total time needed for the crop to get from the front of the machine to the grain tank is also shorter in corn (delay between ground speed and grain mass flow) than in wheat.

In some conditions in corn there is also a very small delay of one or two samples (0.2-0.4 seconds) between pump setting and ground speed. Because this delay is so small - and condition specific - it is very hard to determine it. Therefore the delay is not included in this analysis, but in Chapter 7 the influence of this delay on the controller performance is evaluated.

6.4 Influence of disturbances

The above presented model structure is a simplification of reality, as most models of real-life systems are. By making the model adaptive, as will be discussed in section 6.5, the model is made to fit the process at all times. In this section the effects of unmodeled, measurable and unmeasurable inputs on the above mentioned process variables are discussed. The purpose is solely to give examples of important disturbances, because this gives an impression of the complexity of the process, and what kind of effects will have to be compensated by the time-varying model.

6.4.1 Effect of threshing speed and slope on the transport delay

The machine settings, as well as the slope of the field may have an important impact on the actual time needed for the crop to be transported throughout the machine. The rotation speed of the threshing drum is taken as an example of a machine setting with possible impact on the delay.

Question 6.1. *How large is the impact of the speed of the threshing drum (a machine setting), or the terrain slope on the delays inside the combine harvester process in corn?*

This dependency is examined to get an idea of the delay values in different conditions. It is very important for control purposes to know which aspects have a significant impact on the delay. Since the purpose is solely to illustrate the trends, the results are only shown for corn. The results for wheat are analogous. To investigate this dependency, the experiment of section 6.3 is redone under different conditions, and repeated several times for each condition.

The objective is to evaluate the effect of the different conditions, thus the grain mass flow model is not used to preprocess the ground speed, feed rate and engine

Table 6.2: Delays (in seconds) for corn in several conditions: flat, downhill and uphill, and normal threshing speed (normal rpm) and high threshing speed (high rpm) (F, 20051121b).

	FLAT normal rpm		FLAT high rpm		DOWNHILL normal rpm		UPHILL normal rpm	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
GS→FR	0.68	0.07	0.70	0.00	0.73	0.04	1.00	0.09
GS→EL	0.64	0.14	0.78	0.16	1.60	1.20	0.60	0.05
GS→TD	5.47	0.22	4.93	0.08	5.85	0.21	5.87	0.28
GS→GMF	15.93	0.47	15.10	0.13	17.95	0.14	14.88	0.10
FR→GMF	15.23	0.49	14.27	0.15	16.55	0.14	14.13	0.25
TD→GMF	9.96	0.19	10.00	0.30	11.15	1.27	9.45	0.35
EL→GMF	14.62	0.38	13.82	0.18	14.83	0.32	14.20	0.05
	5 runs		3 runs		2 runs		3 runs	

load signals. This is why the delay is considerably larger in Table 6.2 than in Table 6.1. After all, these dynamics do not influence the delay differences between the conditions.

Since this data set (F, 20051121b) consists of repeated measurements, it is possible to analyze the significance of the difference in delay between two settings. This difference is analyzed with a one-way ANOVA (Neter et al., 1996). The data presented in Table 6.2 can be seen as the result of two distinct experiments, namely:

- slope influence, three treatments: uphill, flat and downhill (all normal threshing speed)
- threshing speed influence, two treatments: normal speed and high speed of the threshing drum (all on flat terrain)

The issue at hand in each of the experiments is whether or not these treatments have the same average delay or not (taking into account all the measurement values). This can be formulated as the following hypothesis test:

$$\begin{aligned} H_0 : &\quad \mu_1 - \mu_2 = 0 \\ H_a : &\quad \mu_1 - \mu_2 \neq 0 \end{aligned} \tag{6.7}$$

ANOVA yields a so-called p-value, which is the probability that the measured data points all come from the same distribution. If this p-value is smaller than 0.05, the treatments are said to have different delays, with a significance level of 95%. Since it is also interesting to know how large the difference in average delay is, a Tukey-Kramer (Neter et al., 1996) test is performed to estimate the confidence intervals of the difference in delay between the treatments with 95% significance. If there are more than two treatments, as in the case of the slope influence, it is possible that the p-value is smaller than 0.05, and that some of the confidence intervals still contain 0. After all, ANOVA tests whether all the treatments have the same average.

Table 6.3: Confidence intervals (seconds) and p-values for all relations in Table 6.2. On the left the influence of the speed of the threshing drum is examined, on the right the slope influence is investigated. Bold means that the p-value is smaller than 5e-2 (5%). LB indicates the lower bound, and UB the upper bound of the confidence interval.

	RPM			SLOPE		
	normal-high		p	downhill-uphill		p
	LB	UB		LB	UB	
GS→FR	-0.19	0.12	6.3e-1	**-0.56**	**-0.11**	**2.5e-3**
GS→EL	-0.40	0.07	1.4e-1	-0.31	2.11	9.4e-2
GS→TD	**0.42**	**0.84**	**2.0e-4**	-0.57	0.44	**2.4e-2**
GS→GMF	**0.08**	**1.45**	**3.3e-2**	**2.10**	**4.10**	**7.0e-5**
FR→GMF	**0.30**	**1.64**	**1.1e-2**	**1.55**	**2.57**	**3.2e-4**
TD→GMF	-0.37	0.44	8.5e-1	**0.38**	**2.82**	**1.6e-2**
EL→GMF	**0.18**	**1.29**	**1.7e-2**	-0.32	1.25	1.8e-1

In Table 6.3 the confidence intervals and p-values are shown for all relations for each of the experiments. For the slope experiment only the difference between downhill and uphill is shown. For the relation between ground speed and torque drum, the difference between flat terrain and uphill is significantly different from zero. For most of the relations there is no significant difference between downhill and flat terrain. This may also be caused by the limited number of measurements downhill.

In the first two columns the delay when the combine operates at normal threshing speed is compared to the delay when the combine threshing drum operates at higher threshing speed. Thus positive values in the confidence interval imply that the delay is larger when the combine operates at normal threshing speed than at high threshing speed.

The confidence intervals marked in bold do not contain 0, and thus indicate that both treatments have a significantly different average. The results show a very logical behavior. Higher threshing speed leads to a faster response of the threshing drum torque. The time spent on the cleaning section is also reduced since the crop is thrown with a higher velocity onto the cleaning section.

A combine driving downhill has a larger capacity than a combine driving uphill (this is also illustrated in the following section), because the cleaning system is no longer horizontal on a slope. A larger capacity means that more material can be processed by the combine in the same time without a peak in sieve losses. Comparison of the delays between driving uphill and downhill illustrates this. The delay driving downhill is for almost all relations larger than the delay driving uphill, which implies the crop spends more time in the combine. In other words, the slope effect enlarges the available storage space in the combine, and thus the capacity.

The larger delay between ground speed and feed rate when driving uphill may come from the fact that it is very hard to position the combine exactly at the crop border on a slope. If there is some distance between the combine and the crop, this will look like a delay in the data processing. Another explanation for the larger delay

Table 6.4: Average losses uphill and downhill for two speeds (4 km/h and 6 km/h) (F, 20050810c).

	SLOW		FAST	
	downhill	uphill	downhill	uphill
mean losses	19.3	65.0	39.5	91.8
std losses	14.4	36.1	24.4	38.9
mean speed	4.0	4.1	5.0	5.0
mean inclination	-4.3	4.5	-4.5	4.2

between ground speed and feed rate when the combine is driving uphill may be that the combine slips when it accelerates (because of the slope). In other words, the combine started moving later than the data shows.

6.4.2 Effect of slope on losses

The previous section already showed that longitudinal inclination of the machine influences the transport delay of the grain kernels. Probably the time needed for the kernels to cover the distance between the threshing drum and the loss sensor across the upper sieve is also influenced by the slope. If the kernels spend less time on the cleaning section, the losses will increase. After all, less time on the cleaning section makes the cleaning section look shorter, since there is less time available to separate the grain kernels from the rest.

Question 6.2. *Does a combine harvester produce more losses uphill than downhill? And are those losses speed-dependent? Can in other words knowledge of the field profile in front of the combine be used to predict and avoid future losses?*

The machine was driven up a slope with constant inclination (of about 4 degrees) at constant speed whilst measuring the sieve losses. Then the machine was driven down the same slope at the same speed. The same test was repeated at a different speed (F, 20050810c).

At both speeds the average sieve losses are higher when driving uphill than when driving downhill. However, since the measurement of sieve losses is extremely noisy, the comparison of the average is not enough to draw conclusions.

Not every kernel that leaves the machine makes contact with the loss sensor, and straw tips (or other MOG) may be registered as kernels. These effects give rise to measurement noise. The mean value and standard deviation for the runs at both speeds (4km/h and 6km/h) are shown in Table 6.4. The loss value is uncalibrated, but proportional to the kg/s loss at the end of the cleaning system. The losses are expressed as number of kernel impacts per interval of 200 ms.

In order to verify the hypothesis that slope and speed have an effect on loss level, some data preprocessing is needed. Each of the four experiments (downhill/uphill, slow/fast) consists of 120 measurements (at 1 Hz sampling rate). Of course, these measurements cannot be seen as independent, since there is a high correlation between two subsequent measurements. Experiments have shown that the delay between

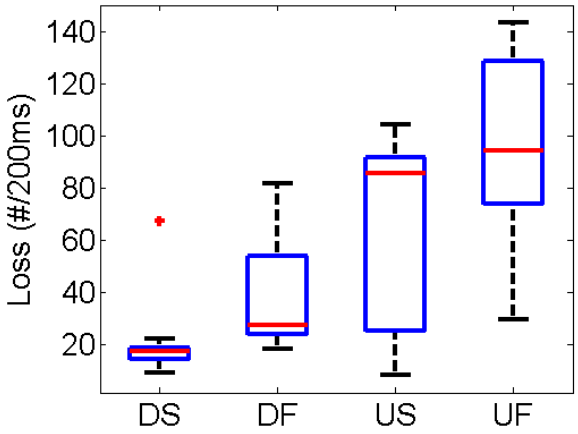


Figure 6.5: Boxplot for the loss level (expressed in number of kernels per 200 ms) for four different case: downhill/slow (DS), downhill/fast (DF), uphill/slow (US) and uphill/fast(UF).

ground speed and sieve losses is between 7 and 10 seconds (as mentioned earlier hard to determine exactly because of the high noise level). Thus, it is safe to assume that a measurement at time t_0 is independent from a measurement at $t_0 + \Delta t$, if Δt is larger than 5 seconds (which is half the delay). After all, in view of the delay of 7 to 10 seconds, kernels that exit 5 seconds later are separated by half the length of the transport system of the combine harvester. Since the loss measurement is highly susceptible to noise, it is best to average the loss signal over a time interval (of correlated measurements), before reducing the resolution. The loss time signal is averaged over 5 second time intervals. Every other average is removed from the data set to obtain independent data points. This means that the data set is reduced to 12 measurements for each experiment. The boxplot of the data is shown in Fig. 6.5.

The objective of this statistical study is two-fold:

- investigate the difference between downhill and uphill at both speeds (and obtain a confidence interval for that difference)

- examine the effect of speed on the loss level

The dataset is analyzed using two-way ANOVA. There are two factors, namely slope and speed, and for each factor there are two treatments, respectively downhill and uphill, and slow and fast. The ANOVA table is shown in Table 6.5.

This two-way ANOVA analysis yields 3 p-values:

- the p-value for the null-hypothesis that driving uphill or downhill has no effect on the losses (first row). The smaller the p-value, the less probable the null-hypothesis. The p-value is 1.4e-7, which rejects the hypothesis with a very high significance. Thus slope has a significant influence on the loss level.

Table 6.5: The ANOVA table for the two-way ANOVA analysis of slope and speed effect on losses (SS: sum of squares, dF: degrees of freedom, MS: mean squares, p: probability of the F-statistic).

	SS	dF	MS	F	p
Slope	3.2e4	1	3.2e4	39.1	1.4e-7
Speed	7.1e3	1	7.1e3	8.7	5.0e-3
Interaction	384.2	1	384.2	0.47	4.9e-1
Error	3.6e4	44	809.0		
Total	7.5e4	47			

Table 6.6: Difference between the treatments of each factor (slope and speed). The average difference as well as the confidence interval is given (LB: lower bound, UB: upper bound).

Difference	LB	value	UB
Uphill-downhill	34.79	51.34	67.89
Fast-slow	7.70	24.25	40.80

- the p-value for the null-hypothesis that driving fast or slow has no effect on the losses. The p-value is 0.005, which rejects the hypothesis with 99.5% significance. In other words, speed significantly influences the losses.

- the p-value for the null-hypothesis that there is no interaction between both factors (fast/slow and downhill/uphill). The p-value is 0.49, which does not reject this hypothesis. Thus both factors have an independent effect.

In a second step the difference between the treatments is estimated (with a 95% confidence interval) for each factor using Tukey-Kramer (Neter et al., 1996). If the confidence interval does not contain zero, both treatments are significantly different. The confidence bounds for the differences are given in Table 6.6. Note that the difference between the two treatments is significant for both factors.

So far we only discussed slopes along the machine axis, but side slopes also have an impact on machine performance. All the crop is thrown to one side of the machine, which results in a lower effective capacity (Quick and Buchele, 1978, Bichel and Cornisch, 1974). The dependency of sieve loss on the field slope causes considerable problems on hilly terrain. When the machine drives on a steep hill, a lot of kernels are thrown out of the machine. To avoid this, special combines have been designed which remain level in all circumstances. These are the so-called hillside combines. For some regions longitudinal compensation suffices (Fig. 6.6 (a)). For other regions full compensation (for longitudinal and transversal slopes) is required (Fig. 6.6 (b)). These combine harvesters can run at full capacity at side slopes up to 18%.



(a) (b)

Figure 6.6: A New Holland CX hillside combine harvester with longitudinal compensation (a) (W, Farmphoto (2008)) and longitudinal and transversal compensation (b) (W, New Holland Austria (2008)). These harvesters are specially designed to keep the cleaning system level horizontal, thus reducing the loss level when driving uphill or even on side slopes.

6.4.3 Effect of slope and grain bin mass on the propulsion system

Since the hydraulic system of the combine harvester is flow-driven (Chapter 10, section 10.3), the propulsion system of the combine harvester is not influenced by the grain bin mass or the slope as long as the diesel engine can provide sufficient power. In field conditions however, this is not always the case, since the threshing and cleaning mechanism also require a lot of power.

The mass of a combine harvester varies considerably during the harvesting process. An empty New Holland CR combine weighs about 14 tons and can maximally load about 8 tons of grain (the standard grain tank has a volume of 10 500 liters). Thus the total weight of the combine can increase by about 50% in a couple of minutes. In order to investigate the effect of the mass in the grain tank and the terrain slope on the propulsion system, the combine pump setting was excited with a sine signal. The pump setting varied between 30 and 50%, with a period of 2 seconds, whilst the combine was harvesting barley (thus the mass of the machine increases with time) (F, 20060711a, F, 20060713a). The normal operating range of the pump setting is between 30% and 50%. The period was taken at 2 seconds because this is about the fastest sine input the combine harvester can respond to. Each run is about 30 seconds. This way the mass in the grain bin does not vary too much during the experiment.

Question 6.3. *How much is the propulsion system influenced by variation in the terrain slope? Does the response of the combine harvester depend on the mass in the grain bin?*

In Table 6.7 the average combine speed is given. The rows mark the terrain type. The further to the right the value stands, the heavier the combine is during the measurement. Note that the average speed increases with the mass when the combine drives downhill, and that the average speed decreases with the mass in the grain bin if the combine drives uphill. This denotes the static system amplification. One can also conclude from this table that the combine speed can vary up to 20% between driving

Table 6.7: The average speed (left) and the speed range (right) during a sine excitation for different masses in the grain tank and different terrain types. The downhill and uphill slopes are approximately 4 degrees (F, 20060711a, F, 20060713a).

	Average speed Increasing mass ⟶			Speed range Increasing mass ⟶		
Downhill	4.7	5.1		2.8	2.3	
Flat		4.2			2.3	
Uphill	4.3		4.1	2.7		2.3

uphill and downhill if the combine is almost fully loaded with grain, even on this very moderate slope of 4 degrees.

Since the pump setting is a sine, the speed varies sinusoidal as well. The range of the sine is also shown in Table 6.7. Note that the range decreases with increasing mass. In other words the dynamic amplification at 0.5 Hz (2 second period) decreases with increasing mass. This implies that the cut-off frequency of the system shifts to lower values as the mass increases.

6.4.4 Effect of the machine load on the grain mass flow dynamics

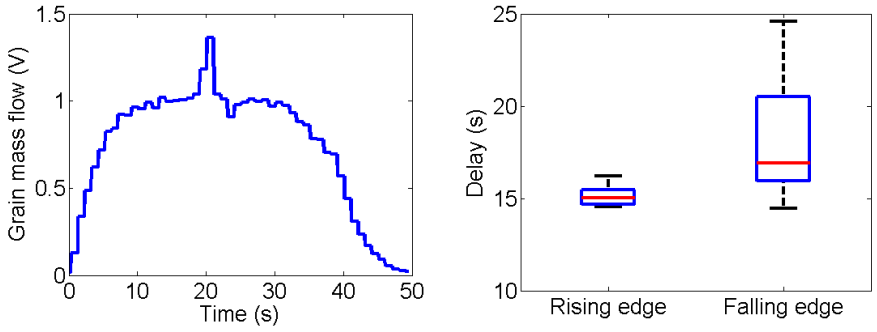
In section 6.2 was already mentioned that the rising and falling behavior of the step response of grain mass flow is not the same. This is illustrated in Fig. 6.7. In this experiment a block wave was applied to the pump setting. The peak in the grain mass flow halfway through the run is when the pump setting is returned to zero. Stopping the machine causes the grain on the sieves to slide forward in the grain bin collector, leading to a peak in the grain mass flow.

Question 6.4. *Is the grain mass flow influenced by the machine load? How can this be explained physically?*

Machine load should be interpreted as the instantaneous capacity utilization. Assuming constant crop density, the capacity utilization is proportional to the machine speed.

Up till the threshing drum the dynamics of the machine are quite fast. There is a swift response from ground speed towards process states such as engine load, feed rate and torque of the threshing drum. Since the dynamics towards grain mass flow are significantly slower (Hemming and Chaplin, 2005) (see also section 6.3), these dynamics must depend on the system beyond the threshing drum, namely the cleaning section at the end of the machine.

From a modeling point of view, the cleaning section can be seen as a nonlinear filter. As described in Chapter 3, the grain and chaff mixture enters the cleaning section on the upper sieve. Through the shaking motion of the upper sieve, the material is transported to the end of the machine. In the mean time, the material can fall through the upper sieve if the particles are small enough. The material that falls through the upper sieve falls on the lower sieve. Here the same behavior occurs as on the upper sieve. If the material falls through the lower sieve, it is transported to the grain tank. If



(a) (b)

Figure 6.7: (a) The step response of grain mass flow. Note the difference between the rising and the falling edge. The peak in the grain mass flow is caused by stopping the combine harvester, and marks the start of the falling edge (F, 20060711b).
(b) Boxplot of the delay of GMF for the rising and the falling edge (F, 20060711c, F, 20060713b).

it does not fall through the upper sieve, the material ends up in the return flow, which means the material is threshed again, and injected once more in the cleaning process. Material which does not fall through the upper sieve, exits the machine.

For grain mass flow, the delay and the dynamics are mainly determined by the average location (or more exactly the spatial distribution of the location) where the material falls through the upper and lower sieve. The characteristics of the material, and the settings of the sieves (the opening and the sieve type) will strongly influence this 'fall-through' location.

This implies that the grain mass flow dynamics vary with the machine load, i.e. the amount of crop processed in the combine per unit of time. This effect was experimentally examined during the barley season. If the delay is determined between ground speed and grain mass flow without taking the grain mass flow dynamics into account, this delay is a good measure of the dynamics as well. After all, the delay is calculated using correlation analysis. In section 6.3 is shown that this correlation also depends on the dynamics. Moreover, if it can be proven that the delay depends on the current machine load, it immediately follows that the delay varies during each transient (change in crop flow). During crop entry or exit, the machine load varies. A dynamically varying delay, leads to nonlinear dynamics as are seen in the data (Fig. 6.7 (a)). This dependency of the delay on the machine load may also explain the difference that is seen between a falling and a rising edge of the grain mass flow.

The delay between ground speed and grain mass flow is examined for the rising and the falling edge separately, without correcting for the dynamics of grain mass flow. The main objective of this experiment is to show that the delay of the falling edge and the rising edge of grain mass flow is different, and that it depends on the machine load. Step inputs of different sizes are applied to the pump setting of the machine (F, 20060711c, F, 20060713b). Each experiment consists of a rising edge and a falling edge. At the start and the end of the experiment the machine is stopped. The delay is determined using cross correlation (see section 6.3).

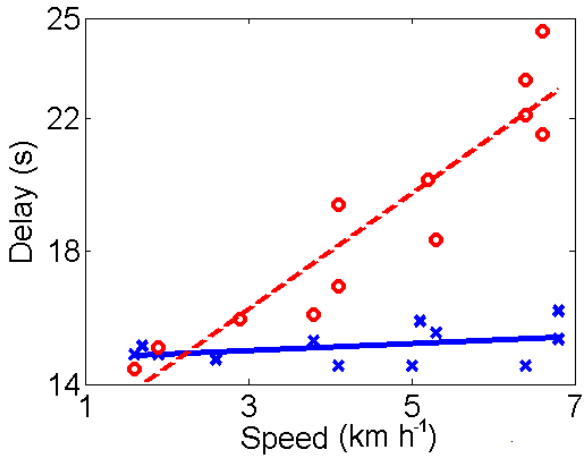


Figure 6.8: The delay between ground speed and grain mass flow as a function of the driving speed. 'x' marks rising edge delays, 'o' marks falling edge delays. The solid line gives a linear least squares approximation for the rising edge. The dashed line does the same for the falling edge.

The boxplot of the delays between ground speed and grain mass flow is shown for the rising edge and the falling edge in Fig. 6.7 (b). The average delay is considerably larger for the falling edge than for the rising edge. Also note that the variation of the delay of the rising edge is much smaller than the variation of the delay of the falling edge.

Since at constant crop density the machine load is proportional to the speed, the speed can be considered a measure of the machine load (capacity utilization). In order to check the dependency of the delay on the machine load, a linear relation of the form $a \cdot x + b$, with a, b parameters, will be estimated between the average ground speed and the delay. The data points as well as the estimated relation are shown in Fig. 6.8.

The plot shows a linear relation between ground speed (or machine load) and delay for the falling edge, with a R squared value of 0.868. The relation between the delay on the rising edge and the machine speed is very weak with a R squared value of 0.144. The quality of this relation can now be expressed by deriving a 95% confidence interval on the parameters of both relations (Table 6.8). Note that the confidence interval for the gain factor of the relation between ground speed and delay for the rising edge contains zero. This means that there is no 95% significant relation between ground speed and the delay of the rising edge. Between ground speed and the delay of the falling edge however, there is indisputably a relationship. The confidence interval of the gain factor remains far from zero.

The experiments show that the delay increases with higher machine load. This effect is only visible on the falling edge. Let us now try to explain this effect based on physical knowledge of the process. A sieve can be modeled as a series of interconnected elements (Hemming and Chaplin, 2005). The material enters the system at the first element. In the next time step, the material can go either one of three directions.

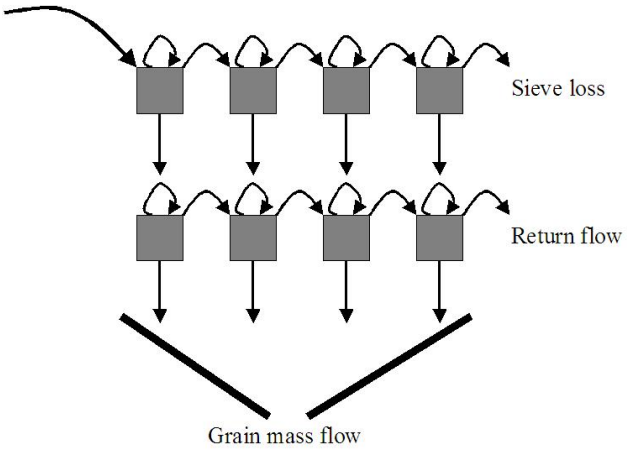


Figure 6.9: The proposed model for the cleaning section. The upper part represents the upper sieve, the lower part represents the lower sieve.

Table 6.8: The confidence interval for the parameters of the relation $a \cdot x + b$ between the average speed and the delay for rising and falling edge. The last column shows the R^2 value for these relations.

		Parameter value			R^2
		Minimum	Value	Maximum	
Rising edge	a	-0.0756	0.1057	0.287	0.144
	b	13.8442	14.6876	15.5309	
Falling edge	a	1.2619	1.7392	2.2165	0.868
	b	8.6928	11.0264	13.36	

It can remain in that element, it can be thrown onto the next element, or it can fall through the sieve. Each of these options has a certain probability of occurring. Two sieve systems can be connected by stating that whenever material falls through the upper sieve, enters the corresponding element on the lower sieve. These probabilities all depend on the cleaning settings (sieve opening, fan speed). The model is illustrated in Fig. 6.9.

Since a fixed percentage of the flow falls through the sieve at each element, the distribution of the amount of kernels which fall through the sieve is an exponential function of the sieve length. Such a distribution leads to a linear grain mass flow response, which is not consistent with the experimental data. Examining this model from a practical point of view however, it is logical that the flow through a given sieve opening is limited. The effect of limiting the throughput of each element of the upper sieve towards the lower sieve is shown in Fig. 6.10. Note that because of this constraint the grain mass flow indeed looks more like Fig. 6.7 (a). However, this model changes the rising edge as well as the falling edge. One could argue that the throughput constraint can be neglected for the rising edge, because the system is not

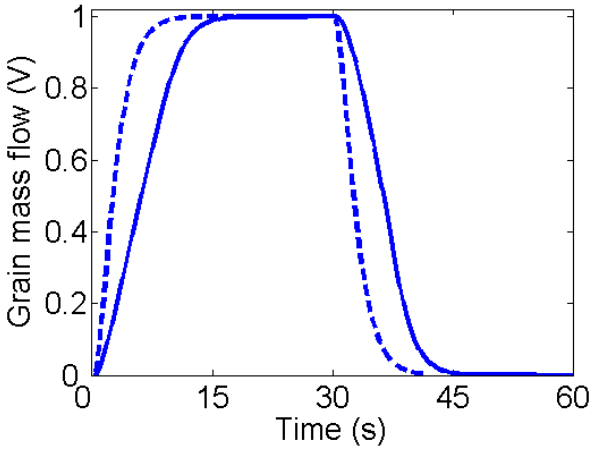


Figure 6.10: The effect of limiting the throughput of each sieve element on the simulated grain mass flow. The dotted line is the grain mass flow without limitation, the solid line is the grain mass flow with limitation.

stationary yet. The constraint only becomes active once a layer has been formed on the sieve.

Fig. 6.11 shows the effect of different loads, as well as the distribution of the crop across the upper and lower sieve for this model. The grain mass flow has been normalized to steady-state value 1 to make the comparison easier. Note that the dynamics become slower as the load becomes higher. The distribution of the crop across the sieves contains two sections: a linear part (where the throughput constraint is active) and an exponential part (where the throughput is smaller than the constraint).

This effect can be seen as a disturbance, since it also depends on the crop type and variety (as well as a moisture content). This makes modeling the effect, based on the machine load (in tons/hour), very hard. One can determine a model of the cleaning system based on this data, but the model would strongly depend on the crop conditions and the machine settings.

6.4.5 Conclusion

This section provides some insight into unmodeled effects. First of all is established that the delays between the process variables depend on the settings of the machine, such as the speed of the threshing drum, and on the terrain slope. However, it is too complex to model this delay. If necessary for the controller performance, the delay could be estimated online, since the combine harvester regularly exits and enters the crop.

The fact that the losses depend on the slope, render knowledge of the field profile useful. In the near future it may be possible to take the field profile in front of the machine into account by using for instance GPS maps.

The slope and the grain bin mass influence the propulsion system in field conditions, together with other effects such as the terrain condition. This experiment gives

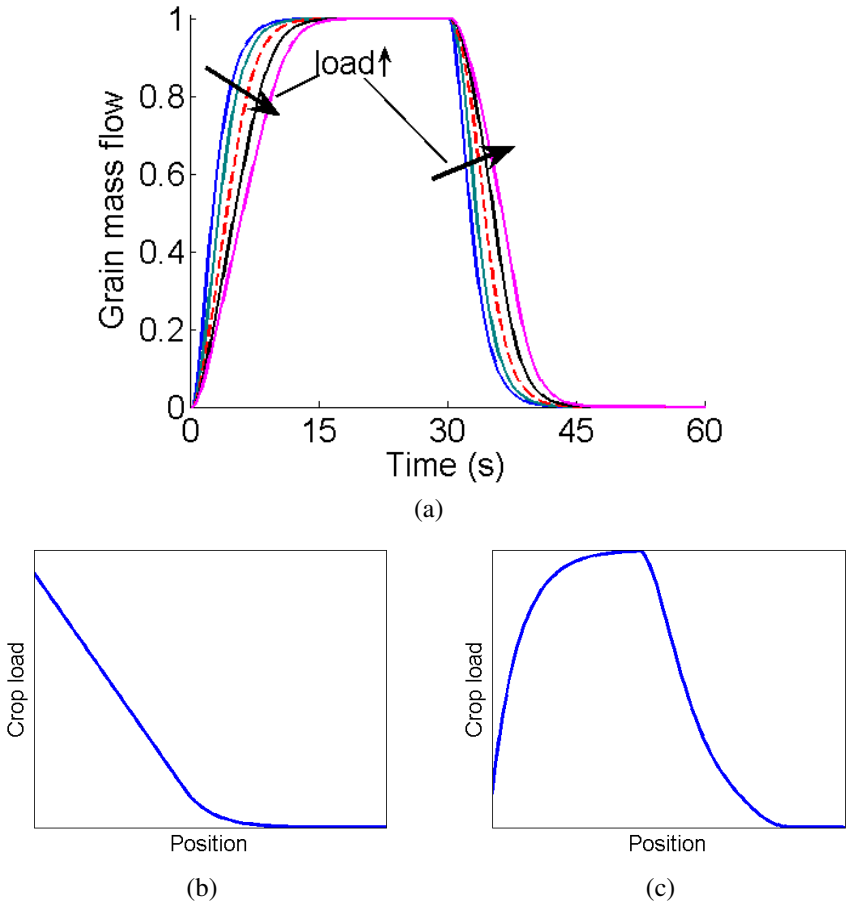


Figure 6.11: The simulated effect of the machine load on the grain mass flow dynamics. (a) The dotted line shows the simulated grain mass flow for an average load. The arrows indicate the change of the response for an increasing load. (b) The profile of the crop distribution across the upper sieve. (c) The profile of the crop distribution across the lower sieve.

Table 6.9: Variation of the gains throughout a field. For each gain the minimum (min), the mean, the median, the maximum (max) and the standard deviation (std) is listed (F, 20060710).

	min	mean	median	max	std
PS→GS	5.85E-02	1.03E-01	1.03E-01	1.41E-01	9.96E-03
GS→FR	5.08E-03	3.82E-02	3.65E-02	1.05E-01	1.70E-02
GS→EL	1.26E+00	5.39E+00	5.22E+00	1.26E+01	2.04E+00
GS→TD	1.25E-02	9.46E-02	8.51E-02	2.39E-01	3.99E-02
GS→GMF	5.30E-02	1.54E-01	1.60E-01	2.88E-01	3.50E-02
FR→GMF	1.23E+00	4.71E+00	4.23E+00	2.64E+01	2.23E+00
EL→GMF	1.22E-02	3.15E-02	3.07E-02	1.45E-01	9.55E-03

an idea of the variation of the propulsion system the controller has to deal with.

The shape of the rising edge of the grain mass flow dynamics is more or less constant, but the falling edge of grain mass flow considerably depends on the machine load. The larger the machine load, i.e. the larger the capacity utilization, the larger the delay between ground speed and grain mass flow becomes for the falling edge. Fortunately, the falling edge is not that important for capacity control.

6.5 Time-varying model

Gain variation

The presented model contains a lot of gain factors, which are all related to external factors, biological or otherwise:

pump setting→ground speed: this relation depends on the field slope, terrain conditions, mass in the grain bin,...

ground speed→feed rate: this gain is more or less proportional to the crop density. Since the feed rate signal in corn also includes the measurement of the torque used by the choppers, this gain is in corn much larger than in wheat or barley.

ground speed→engine load: this relation also depends on the crop density, but is also influenced by the same effects as the pump setting-ground speed relation, especially during acceleration. Unloading the grain bin will also influence this relation.

feed rate→grain mass flow: the most dominant effect here is the grain-straw ratio.

engine load→grain mass flow: analogous to the feed rate-grain mass flow relation, but also influenced by the variation of the ground speed-engine load relation.

Before calculating the gain factors, the signals are shifted over the delay determined in section 6.3. Especially for the gains of relations which involve grain mass flow, it is important to take the delay into account.

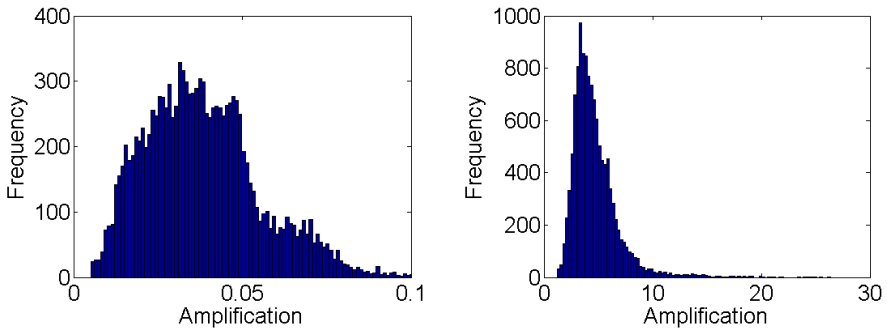


Figure 6.12: Histogram of (a) the *GS-FR* amplification over an entire barley field and (b) the *FR-GMF* amplification over the same field.

Table 6.10: The gains for different crop types: barley, wheat and corn.

	barley	wheat	corn
PS→GS	0.103	0.0905	0.1088
GS→FR	0.0382	0.0764	0.2362
GS→EL	5.39	8.6543	7.9127
GS→TD	0.0946	0.1097	0.0174
GS→GMF	0.154	0.1746	0.2372
FR→GMF	4.71	2.2868	1.0040
EL→GMF	0.0315	0.0202	0.0300

To illustrate the variation of the gains in practice, a barley field has been harvested in normal operation. The barley field is about 2 hectares, and shows significant variation because of its location next to a forest. The variation of the gains throughout the field (F, 20060710) is summarized in Table 6.9. The histograms of two relations, namely the ground speed-feed rate and the feed rate-grain mass flow relation, are shown in Fig. 6.12.

These gains strongly depend on the crop type. Example values of these gains for barley, wheat and corn are summarized in Table 6.10. These are the average values over comparable fields in wheat and corn. The barley field is the same as the one in Table 6.9. Table 6.10 only gives an indication of what the possible value for these gains are for these three crops. The variation of the gains over several fields has not been determined. There are some differences between the crops. All relations containing feed rate vary depending on the crop type. Note also the extremely low gain for the ground speed-torque drum relation in corn. This is because in corn only the ear enters the threshing system, whereas in small grain crops the straw also enters the threshing system.

Gain estimation

Several techniques are available to perform such estimation online. Two techniques are discussed and compared in this research, namely a Sliding Mode estimator and a Recursive Least Squares (RLS) estimator.

Because the process is time-varying, a gain factor will never be estimated exactly. Each gain factor can be seen as a stochastic variable with a mean and standard deviation. Later on in this thesis this stochastic information will be used in the controller. Therefore it is important that the parameter error, which is actually the uncertainty on the estimated parameter, is normally distributed. This is important to determine the probability of constraint violation in Chapter 8. It will be shown that the choice of the estimator has a considerable impact on the distribution of the parameter error.

All the estimated parameters are strongly linked to the crop and environment conditions. This means that one can assume that these parameters can only vary in a continuous manner within a field. However, since the harvesting operation is regularly interrupted to be resumed in another field, or in another part of the field, it is no longer guaranteed that these parameters vary continuously. Due to differences in moisture content for instance (the crop in the middle of the field versus the crop at the side of a field near trees), parameters such as the ground speed-feed rate gain may vary considerably.

Recursive Least Squares The Recursive Least Squares algorithm calculates an optimal parameter value by minimizing the square of the model error, taking into account the previous parameter value. More information about RLS and its applications can be found in Nelles (2001) and Vahidi et al. (2005).

RLS is basically described by three variables: amplification K, error e and covariance R. The relation between those variables is:

$$R(N) = (1-\gamma)R(N-1) + \gamma x^2(N) \tag{6.8}$$

$$e(N) = y(N) - K(N-1)x(N) \tag{6.9}$$

$$K(N) = K(N-1) + \gamma e(N)\frac{x(N)}{R(N)} \tag{6.10}$$

where γ is the update rate ($0 \leq \gamma \leq 1$), $x(N)$ is the input at time instant N and $y(N)$ is the output at time instant N. The update rate determines the filtering action of the estimator. The smaller the update rate, the longer it will take before a change in the process amplification reaches the estimated value $K(N)$.

The adaptive model will converge to the new gain factor without any bias. The settings of the estimator are, however, a trade-off between convergence speed and noise rejection. In other words, one cannot demand a swift response if the parameter makes a jump, and on the other hand expect a good noise rejection. In control theory there is an analogue problem, for instance in the tuning of a PID controller. For some systems one wants a fast response if the system is far from the set point, and a somewhat slower response when the system approaches the set point to avoid instability. This can be solved by using gain scheduling: change the controller gain depending on the controller error. This idea can also be used for the RLS estimator. The update rate can be varied as a function of the model error. If the error is larger than a certain threshold value, the update rate is set to a higher value.

The convergence of an RLS estimator normally yields a smooth gain estimate. When the update rate is simply abruptly switched from one value to another, the derivative of the gain estimate is no longer continuous during transition phases. This can be avoided by defining a transition region around the error threshold. The actual update rate is then determined based on the membership of the linguistic variables large error and small error. Of course this concept can be expanded to more than two different update rates. The update rate γ is then varied as a function of the model error by fuzzifying the model error. This approach will be called Fuzzy RLS further on. More information about fuzzy arithmetic can be found in (Passino and Yurkovich, 1998).

First of all two membership functions have to be introduced, M_{small} and M_{large}. These membership functions map the value of the error to the linguistic variables 'small' and 'large'.

$$M_{small}(e) = \begin{cases} 1 & e <= a \\ \frac{b-e}{b-a} & a < e <= b \\ 0 & e > b \end{cases} \tag{6.11}$$

$$M_{large}(e) = \begin{cases} 0 & e <= a \\ \frac{e-a}{b-a} & a < e <= b \\ 1 & e > b \end{cases} \tag{6.12}$$

where a and b are respectively the beginning and the end of the transition zone between full membership and no membership. The rule base below relates the error to the update factor:

Error	Update factor
small	γ_{small}
large	γ_{large}

where γ_{small} is a small update rate (e.g. 0.01), whereas γ_{large} is a large update rate (e.g. 0.2). The eventual update rate is calculated using the following formula:

$$\gamma_{FRLS} = \gamma_{small} M_{small} + \gamma_{large} M_{large} \tag{6.13}$$

The effect of this Fuzzy RLS approach will be illustrated further on.

Sliding mode More details about sliding mode control can be found in Utkin (1992) and Edwards and Spurgeon (1998). Sliding mode is especially suited to estimate amplifications of nonlinear systems. The concept is in fact quite simple:

1. Calculate the model error $e(k)$ at time step k based on the previous estimated parameter value $K(k-1)$:

$$e(k) = y(k) - K(k-1) \cdot x(k) \tag{6.14}$$

2. Depending on the sign of the error, define a new parameter value $K_d(k)$ as the previous gain $K(k-1)$ increased or decreased with a given fixed value K_{cor}:

$$K_d(k) = K(k-1) + \text{sign}(e(k)) \cdot K_{cor} \tag{6.15}$$

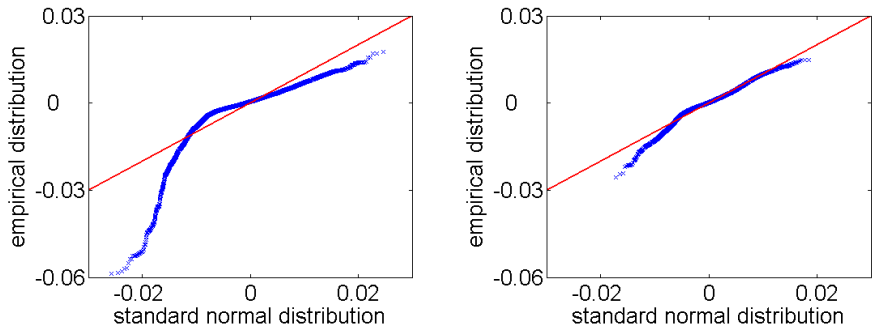


Figure 6.13: The qq-plot for RLS (a) and Fuzzy RLS (b) for the gain of the pump setting-ground speed relation. Fuzzy RLS indeed leads to an increased degree of normality of the parameter error.

3. Filter the sequence of discrete parameter values $K_d(k)$ with a lowpass filter, for instance a first order Butterworth filter. This filtered sequence is the estimated process gain $K(k)$:

$$K(k) \quad = \quad \alpha K(k-1) + \beta K_d(k) + \gamma K_d(k-1) \tag{6.16}$$

with α, β and γ the parameters of the first order Butterworth filter with a cut-off at for instance 0.5 Hz.

Statistics As mentioned earlier, the statistical properties of the remaining parameter errors are very important. Ideally, this error should have a normal distribution, since this will greatly simplify the calculations further on in the text. In order to evaluate the gain estimates of the different relations with RLS, Fuzzy RLS and Sliding Mode, these estimators are tested on field data (a barley field, F, 20060710). The complete data set will be fed to the estimator. The estimator will only update if both inputs (x and y) are larger than a given threshold. This precaution is implemented in the field software as well.

The degree of normality can be estimated by plotting the experimental percentiles of the parameter error against the percentiles expected if the parameter error had a normal distribution (a so-called qq-plot). If the experimental data indeed belong to a normal distribution, this plot will be a straight line. In other words, the degree of normality is proportional to the coefficient of correlation between the percentiles of the normal distribution and the experimental percentiles. In Fig. 6.13 this qq-plot is shown for both RLS and Fuzzy RLS for the gain of the pump setting-ground speed relation.

Another interesting performance measure of an estimator is the rms model error and the effort required to obtain that error. In this case the effort is the amount of noise on the parameter estimate. A nervous estimator will change the parameter estimate without lowering the rms model error. The input effort can be quantified by

filtering the parameter value, and calculating the rms value of the difference between the parameter value and the filtered valued. For this experiment a second order Butterworth filter with a cut-off frequency at 0.1 Hz is used. The filter is applied in forward and reverse time direction to avoid phase distortion. Since it is only used for offline evaluation, this is no problem.

In Table 6.11 the different estimators are compared. For each estimator the squared coefficient of correlation (called R squared) is given, as well as the relative noise (the rms value of the noise on the parameter expressed in % of the parameter value) and the rms value of the model error.

In terms of normality of the parameter error, Fuzzy RLS is clearly superior to the other algorithms (Fig. 6.13). This could be expected since Fuzzy RLS avoids that a large error needs a lot of time to converge by modifying the update rate. The relative noise value however is larger for Fuzzy RLS than for RLS. This is caused by the faster response (if the error is large) of the Fuzzy RLS algorithm. The SM relative noise value is similar to the RLS noise value. The model error (rms value) is also the smallest for the Fuzzy RLS algorithm. This is once more caused by the faster response to persisting large errors. The SM model error is in general larger than that of the RLS algorithm. Table 6.11 illustrates that the performance of an estimator is a trade-off between smoothing the noise on the estimated parameter and following parameter changes rapidly. This can be illustrated by considering two Fuzzy RLS experiments on the same data set as used above. The second experiment with the Fuzzy RLS algorithm (FRLS2) uses half the error threshold (different values for the transition zone between the 'small' and the 'large' membership function). This means that it will react even faster to large model errors. The results are shown in Table 6.12. Indeed the error is smaller for the FRLS2 algorithm, but the relative noise level on the parameter is larger as well. This perfectly illustrates the trade-off pointed out above. The Fuzzy RLS algorithm will be used for all the parameter estimations in this research. The parameters of the algorithm are determined by experimental tuning.

The histograms of the parameter error are compared in Fig. 6.14 for the gain of the relation between pump setting and ground speed. The boxplot in this figure clearly illustrates the benefits of Fuzzy RLS.

Note that all algorithms perform less well on the relation between feed rate and grain mass flow. The precautions listed in the next paragraph will considerably improve the estimator performance.

Special precautions Some gain factors strongly depend on external circumstances. The pump setting-ground speed relation for instance strongly depends on the slope of the field. Since the standard operation of a harvester is to drive up and down a field, a harvester will often drive up a slope, turn at the top, and drive down again. This means that the amplification between current and ground speed makes a jump from the 'downhill value' to the 'uphill value' and vice versa. If the amplification is estimated online, this implies that some transition time is needed to converge to the correct setting. This transition time is shortened by using Fuzzy RLS, but it would be better if the slope knowledge would be incorporated in the estimator.

The slope information (longitudinal inclination) can be converted into one of three fuzzy variables, namely uphill, flat or downhill terrain. The membership function is made trapezoidal. There are three independent estimators: downhill, uphill and

Table 6.11: Comparison of three different estimation techniques for the gains, namely Recursive Least Squares (RLS), Fuzzy Recursive Least Squares (FRLS) and Sliding Mode (SM).

	RLS			FRLS			SM		
	R^2	rel noise (%)	rms error	R^2	rel noise (%)	rms error	R^2	rel noise (%)	rms error
PS→GS	0.78	4.32E-03	2.39E-01	0.88	5.79E-02	2.16E-01	0.75	1.41E-02	2.89E-01
GS→FR	0.97	3.70E-02	3.87E-02	0.99	3.69E-01	3.41E-02	0.97	3.63E-02	5.67E-02
GS→EL	0.97	1.95E-02	4.37E+00	0.99	2.97E-01	3.61E+00	0.94	5.44E-02	5.56E+00
GS→TD	0.92	1.99E-02	8.42E-02	0.98	4.08E-01	6.07E-02	0.92	3.77E-02	1.29E-01
GS→GMF	0.96	9.11E-03	6.59E-02	0.97	1.22E-01	6.46E-02	0.97	2.01E-02	7.99E-02
FR→GMF	0.66	6.21E-02	1.41E-01	0.79	1.02E+00	1.16E-01	0.73	2.36E-02	2.31E-01
EL→GMF	0.79	2.72E-02	1.11E-01	0.9	3.35E-01	1.02E-01	0.85	2.34E-02	1.54E-01

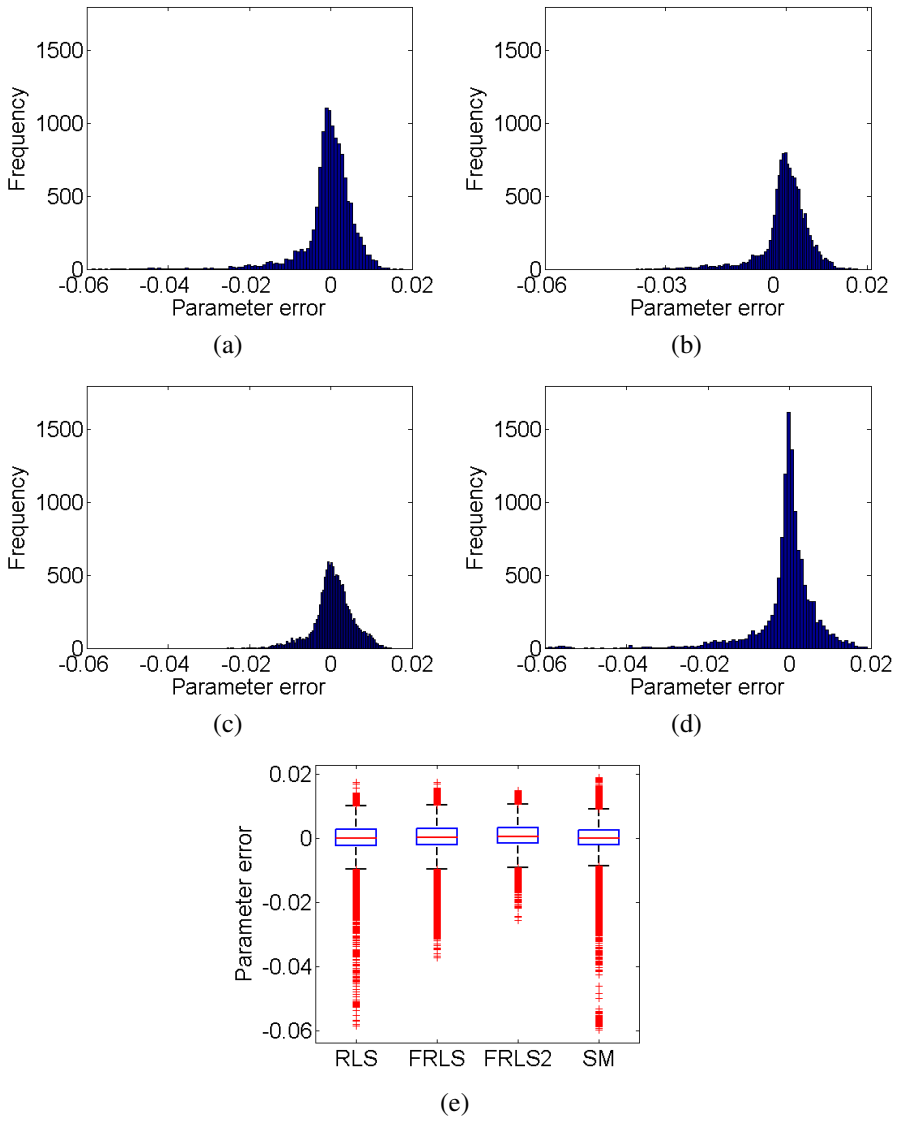


Figure 6.14: The histogram of the parameter error for different estimation techniques for the gain between pump setting and ground speed: RLS (a), FRLS (b), FRLS2 (c) and SM (d). The different histograms are summarized in the boxplot in (e). Note that there is a clear difference between RLS and SM on one hand and Fuzzy RLS on the other hand. It is not immediately clear however, whether the error threshold has a significant impact on the degree of normality.

Table 6.12: Comparison of two different settings of the Fuzzy RLS algorithm, namely the settings of Table 6.11 (FRLS) and a version with half the error threshold (FRLS2).

	FRLS			FRLS2		
	R^2	rel noise (%)	rms error	R^2	rel noise (%)	rms error
PS→GS	0.88	5.79E-02	2.16E-01	0.97	9.85E-02	1.84E-01
GS→FR	0.99	3.69E-01	3.41E-02	1.00	6.33E-01	2.61E-02
GS→EL	0.99	2.97E-01	3.61E+00	0.99	4.07E-01	2.76E+00
GS→TD	0.98	4.08E-01	6.07E-02	0.99	4.58E-01	4.83E-02
GS→GMF	0.97	1.22E-01	6.46E-02	1.00	1.72E-01	6.07E-02
FR→GMF	0.79	1.02E+00	1.16E-01	0.79	1.55E+00	8.99E-02
EL→GMF	0.90	3.35E-01	1.02E-01	0.91	5.53E-01	7.85E-02

flat. The degree of membership of each determines the update rate of each of these estimators. If the slope inclination (in degrees) lies within $[-1, 1]$, only the 'flat' estimator is updated. If the inclination lies in the interval $\pm\,(1, 4)$ the 'flat' as well as the 'downhill' or 'uphill' estimator is used. For other slope inclinations only the 'downhill' or 'uphill' estimator is used. The degree of membership is also used to combine the three gain estimates into one gain factor. For the comparison of the estimators on the barley field this is not relevant since the barley field was flat.

Since especially the gain in quasi-static conditions is of interest for the model, the parameter estimator will be disabled during crop entry and if the signals are not within their normal range. Crop entry is detected, and some time after crop entry the estimator is activated. Since not all the field loggings include the crop entry, this is not implemented for the above-described estimator comparison. The signal constraints however, have also been taken into account for the estimator comparison.

6.6 Model for capacity control on the combine harvester

The motivation behind capacity control has been extensively discussed in Chapter 3. The main objective is to run the combine harvester at maximal capacity all the time, even when the operator gets tired. The model for capacity control has been derived throughout this chapter. In this section the model which is used for the controller is summarized.

Model structure The general model structure has been defined in Fig. 6.1. For the capacity control system the relevant process variables are pump setting, ground speed, feed rate, engine load and grain mass flow. As described in section 6.2, only the dynamics of the relation between ground speed and engine load, and the dynamics of the relation between feed rate and engine load on one hand, and grain mass flow on the other hand are relevant for control purposes. Field experiments have also shown that the dynamics between pump setting and ground speed, and between ground speed

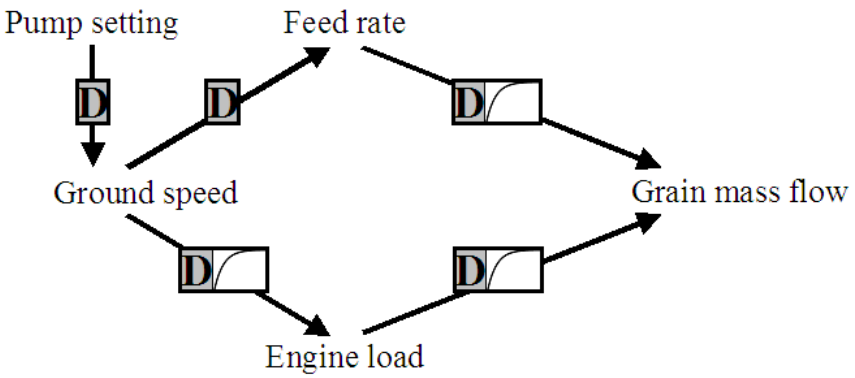


Figure 6.15: Schematic of the capacity control model. Blocks with 'D' contain a delay. The blocks with the step response also contain dynamics.

and feed rate are not important during normal field operation. On the road - at higher speeds - the dynamics between pump setting and ground speed however do become relevant (Chapter 10). An overview of the model used for capacity control is shown in Fig. 6.15.

Zero load Two process variables, namely feed rate and engine load, show an offset which drifts slowly in time. The drift in feed rate measurement is caused by the varying temperature of the belt whose tension is measured by the feed rate sensor. The engine load drifts because of the varying general machine condition (such as load or wear). These offsets can be corrected for by calibrating regularly. Each time the header is lifted, while the header as well as the threshing are on, a moving average is taken, starting from the previous zero load value. This guarantees a good approximation of the actual offset.

Time-varying behavior All the process gains are made time-varying, and are estimated online using Fuzzy RLS (section 6.5). The delays are kept constant, and not varied online. Different delays are used for the different crops (section 6.3).

Re-aligned model The model is realigned at every time step. This means that it only predicts the change of the model state based on the change of the input. This increases the robustness of the prediction when faced with model uncertainty (Maciejowski, 2002).

Model The results of the time domain identification in section 6.2 (Eqn (6.1), (6.2) and (6.3)), combined with the required delay states, lead to the following process model:

$$
\begin{aligned}
GS_{k+1}^{0} &= GS_k^0 + K_{GS}\Delta u_k \\
&\vdots \\
GS_{k+1}^{d_1} &= GS_k^{d_1-1} \\
FR_{k+1}^{0} &= FR_k^0 + K_{FR}\left(GS_k^{d_{FR}} - GS_k^{d_{FR}+1}\right) \\
&\vdots \\
FR_{k+1}^{d_2} &= FR_k^{d_2-1} \\
EL_{k+1}^{T} &= (1-\alpha_{EL})\left(EL_k^T + K_{EL}\left(GS_k^{d_{EL}} - GS_k^{d_{EL}+1}\right)\right) \\
EL_{k+1}^{0} &= EL_k^0 + \alpha_{EL}\left[K_{EL}\left(GS_k^{d_{EL}} - GS_k^{d_{EL+1}}\right) + EL_k^T\right] \\
&\vdots \\
EL_{k+1}^{d_3} &= EL_k^{d_3-1} \\
GMF_{k+1}^{T,FR} &= (1-\alpha_{FR-GMF})\left[GMF_k^{T,FR}\right. \\
&\quad \left.+K_{FR-GMF}\left(FR_k^{d_{FR-GMF}} - FR_k^{d_{FR-GMF}+1}\right)\right] \\
GMF_{k+1}^{T,EL} &= (1-\alpha_{EL-GMF})\left[GMF_k^{T,EL}\right. \\
&\quad \left.+K_{EL-GMF}\left(EL_k^{d_{EL-GMF}} - EL_k^{d_{EL-GMF}+1}\right)\right] \\
GMF_{k+1} &= GMF_k \\
&\quad +\gamma_1\alpha_{FR-GMF}\left[K_{FR-GMF}\left(FR_k^{d_{FR-GMF}} - FR_k^{d_{FR-GMF}+1}\right)\right. \\
&\quad \left.+GMF_k^{T,FR}\right] \\
&\quad +\gamma_2\alpha_{EL-GMF}\left[K_{EL-GMF}\left(EL_k^{d_{EL-GMF}} - EL_k^{d_{EL-GMF}+1}\right)\right. \\
&\quad \left.+GMF_k^{T,EL}\right]
\end{aligned}
\tag{6.17}
$$

with

$$
\begin{aligned}
d_1 &= \max\left(d_{EL}, d_{FR}\right) + 1 \\
d_2 &= d_{FR-GMF} + 1 \\
d_3 &= d_{EL-GMF} + 1 \\
1 &= \gamma_1 + \gamma_2
\end{aligned}
\tag{6.18}
$$

where d_{FR} is the delay between ground speed and feed rate, d_{EL} is the delay between ground speed and engine load, d_{FR-GMF} is the delay between feed rate and grain mass flow and d_{EL-GMF} is the delay between engine load and grain mass flow. K_{GS}, K_{FR}, K_{EL}, K_{FR-GMF} and K_{EL-GMF} are adaptive process gains which are estimated online (section 6.5). γ_1 and γ_2 are the weighting factors of the grain mass flow models. These parameters are also determined online based on the uncertainty

of each model. FR^0 and EL^0 are the actual feed rate and engine load states. The superscript indicates how many samples it has been delayed. The matrix notation of this model is shown in Appendix F. The sampling time of the model is 5 Hz.

6.7 Conclusion

In this chapter it is shown that, although the harvesting process contains nonlinearities, the process can be modeled with linear dynamics and a simple model structure, by making the relevant gain parameters adaptive. The delays in the system vary depending on the crop type, and are also influenced by the terrain slope and the threshing speed.

In order to be able to establish the quality of the model, the importance of unmodeled effects is examined. The slope for instance influences the loss level, and in field conditions the slope and the mass in the grain bin influence the propulsion system. The falling edge of the grain mass flow even depends on the machine load. In general however, most of these effects can be coped with by the adaptive model.

The technique used to estimate the adaptive gains is very important. As will be shown later on, the controller assumes that the parameter error is normally distributed. The distribution of the error heavily depends on the used estimation technique. Fuzzy RLS is selected as the best technique. This is a form of Recursive Least Squares (RLS) where the update rate depends on the error. For this technique the parameter error can be assumed to be normally distributed.

Now the model is available, we can move to control the capacity of the combine harvester using MPC. This is tackled in the next chapter.

Chapter 7

Model-based Predictive Control

The more unpredictable the world is,
the more we rely on predictions.
-Steve Rivkin-



The previous chapter discussed the system modeling. Now that a model is available, we can move on to control, the next pillar to arrive at real-life applications. This chapter discusses the implementation of a capacity control system using MPC. First of all the controller specifications are listed. Then the important tuning parameters of the objective function are discussed. The influence of the model definition is also illustrated. At the end of this chapter the MPC is benchmarked with two classic controllers, namely a PID and a fuzzy feed rate controller.

7.1 Controller specification

First of all, the controller specifications need to be established, based on the requirements listed in Chapter 4. As shown before in Chapter 2, each MPC consists of three parts, namely:

- the objective function which defines the relation between the input effort and the control error (see Appendix D).
- the model which defines the relation between the process variables and the inputs.
- the constraints which define the allowed operating region of the process and input variables.

The definition of the objective function, leads to several questions, such as:

- which variables need to be controlled?
- how much freedom to control is accorded to the controller?
- which relations exist between the variables?
- which constraints need to be respected?

Table 7.1: The range of the four controlled variables in two crop types: wheat (F, 20050817 and F, 20060905b) and corn (F, 20061017 and F, 20061030b) over several measurement runs.

	Range			
	Wheat		Corn	
	Min	Max	Min	Max
Speed (km/h)	0	6	0	12
Feed rate (V)	0.8	1.6	1.5	3.5
Engine load (%)	30	100	30	110
Grain mass flow (V)	0	1.4	0	2.25

7.1.1 Variables

This chapter presents the procedure to define a MPC, which is immediately applied to a real-life application, namely capacity control on a combine harvester. As illustrated in Chapter 3, there are many possible definitions of capacity of a combine harvester. In this research feed rate, engine load and grain mass flow are considered as capacity measures. Of course only one of these signals is controlled at the same time. The other measurements may be used as constraints. Which variable is controlled depends on the circumstances as described in Chapter 4.

For any controller, the range of the controlled variables is very important. The four controlled variables (speed, feed rate, engine load and grain mass flow) have however very different characteristics, which may even depend on the crop type. In Table 7.1 the measured range for the four variables is shown in wheat and corn.

The zero load value (the measurement with all components running without crop entering the machine) may also vary for feed rate and engine load depending on the field conditions and the crop type (the header depends on the crop type). The zero load value will therefore be updated regularly during operation.

7.1.2 Control freedom

Controlling is not free. Controlling a process leads to several costs, such as:

operator comfort: since the operator has to drive the combine for 8 hours or more each day, it is very important not to make aggressive speed adjustments unless it is necessary.

fuel consumption: speed variations will lead to a higher fuel consumption, especially in muddy field conditions.

machine wear: excessive speed changes will cause machine wear in the belts and other components of the system.

product quality: in crops where the straw enters the combine harvester, a swath exits the machine at the back. A speed increase will lead to a temporarily thinner swath, a speed decrease will lead to a temporarily thicker swath. Swath thickness variations can lead to problems for the baler afterwards.

The trade-off between control cost and control error is thus very important. In view of the variable characteristics of the controlled variables (Table 7.1), it will be difficult to define a set of fixed objective weights which performs well in all circumstances, even if these parameters are made dependent on the crop type.

Input effort is closely related to the change of current, rather than to current itself. Thus, although current is the actual system input, the input effort term in the objective function should be specified in terms of the change of current.

7.1.3 Model

One of the key components of a successful MPC implementation is a good process model. The process model for capacity control is presented in Chapter 6. The better the model, the better the controller will be, at least in theory. One should however, not lose the practical aspects out of sight. Depending on the model and the objective structure, a different optimization method will be required. Thus, there exists a trade-off between two extreme choices, which each have their pros and cons. One model type delivers computational simplicity, the other type yields higher accuracy.

a simple, linear model: the major advantage of this type of model is that it can very easily be used by MPC solvers. If the objective function is quadratic, which it usually is, this leads to a computationally tractable QP problem. On the other hand, most practical processes are not linear, often not even close.

a more complex, nonlinear model: many applications, like a combine harvester, contain considerable nonlinearities. Thus, from a modeling point of view, a nonlinear model would surely be more suited. MPC-problems with a nonlinear model are, however, harder to solve online.

As the combine harvester process is time-varying (it depends on the field properties), the model will also have to be adaptive. One of the main concerns of adaptive models is persistent excitation. Parameters can only be estimated online if there is sufficient excitation. Since a combine harvester operates most of the time in steady-state, it is hard to estimate more than one parameter per measurement. For the identification excitation signals can be applied. This not possible for all applications. In a chemical plant for instance, it is often not allowed to excite the process. One can combine an adaptive process gain with a nonlinear model, assuming that the nonlinear effect is not time-varying.

Since the reference input is more or less constant in normal operation, the main consequence of the nonlinearities is that the model depends on the working point. This type of effect can be dealt with by the adaptive model.

The input of the used model is the change of the pump setting, which determines the speed of the combine. The process model is realigned, the change of the states is modeled based on the change of input. This renders the model much more robust to disturbances.

7.1.4 Constraints

The last aspect of the controller specification consists of defining the constraints. The ability to deal with constraints is one of the major advantages of MPC. There are

three main types of constraints, namely input constraints, state constraints and output constraints. For all practical purposes state and output constraints are analogous.

Input constraints on the other hand, are something completely different. If one calculates the optimal input of a system, without considering the constraints, this controller output can easily be constrained to satisfy the input constraints, at least for a single input system. In other words, if the sole constraints are input constraints, an optimization method that takes the constraints into account is not required. Thus, input constraints can be implemented with little or no computational cost for a single input system. State and output constraints on the other hand do imply a significant cost, since taking these constraints into account requires special optimization methods, such as an Interior-Point Method (IPM) or an Active Set Method (ASM) (see Chapter 9).

7.2 Objective function configuration

The objective function contains several parameters that have to be tuned, namely:

- cost terms and their relative weight
  - control error (in speed, feed rate, engine load or grain mass flow)
  - input effort
- horizon
  - prediction horizon
  - control horizon

For the capacity control application two main cost terms are present in the objective function, namely control error and input effort. As mentioned earlier, input effort is specified in terms of change of current.

7.2.1 Horizon effect

As explained in Chapter 2, the horizon length is tightly linked to controller stability. It is however, not always clear how long the horizon should be to guarantee stability. A theoretical result about the optimal horizon length is given by Primbs and Nevistic (2000).

There are in fact two horizons, namely the control horizon (N_C) and the prediction horizon (N_P). The control horizon defines how many future control values (controller outputs) are calculated by the optimization. The prediction horizon determines how far the model predicts the system response in the future. The control horizon can never be larger than the prediction horizon.

The horizon length determines the number of terms in the objective function. In order to be able to vary the horizon length without disturbing the balance between control error and input effort, even in the case that the control and the prediction horizon are not equal, the weight of the control error and the input effort is in this research always divided by the control and the prediction horizon, respectively.

Table 7.2: The input effort (I_L and I_H) and the control error (E_L and E_H) (expressed in V) as a function of the prediction horizon $N_P = N_C$ (F, 20051123a).

ID	N_P	Input effort		Control error	
		I_L	I_H	E_L	E_H
1	20	0.117	0.040	0.111	0.027
2	40	0.103	0.036	0.097	0.024
3	60	0.080	0.040	0.079	0.026
4	80	0.101	0.045	0.094	0.030
5	100	0.099	0.041	0.092	0.027

Prediction horizon Let us first examine the impact of varying the prediction horizon. Two effects are important:

- lengthening the horizon will make the controller more stable. As the horizon moves to infinite length, the controller will get an increasingly better view of the effect of its actions.
- for real-life applications, increasing the horizon also has a second effect:

  constant N_C: increasing the prediction horizon beyond the time that the system enters steady state, does not add any useful information. Assuming the prediction horizon is already sufficiently long (Primbs and Nevistic, 2000), further increasing the prediction horizon will not increase the performance, but will increase the computational cost of the controller.

  $N_C = N_P$: another effect comes into play. As the control horizon increases, the number of optimization variables increases as well. This creates more control freedom. Once the control horizon becomes larger than the order of the system, the solution is overparameterized. More variables have to be estimated, but the solution can no longer improve. In theory this is no problem. In practice however, the numerical condition of the problem may lead to a decrease in the quality of the solution. Moreover, inputs in the distant future are not important, since the system state will have changed by then due to disturbances.

A field experiment was performed in which different horizon lengths were compared for feed rate control (F, 20051123a). The results are listed in Table 7.2. More information about the used performance measures can be found in Appendix D. First note that the high frequency input effort (I_H) and the high frequency control error (E_H) both remain more or less constant at respectively 0.04 and 0.03. This, together with the fact that the low-frequency input effort also remains more or less constant, means that, as can be expected, changing the horizon length has no effect on the input change (which is a measure for operator comfort). The table also shows that the low frequency control error (E_L) decreases with increasing horizon. The experiment with horizon 60 samples (12 seconds) has the best performance, but this large difference in control error is most likely caused by slightly different field conditions. The lower value for

low-frequency input effort implies that the crop density showed less variation in this experiment.

It should come as no surprise that increasing the horizon to more than 20 samples has no huge impact on the system (considering the system dynamics). The horizon could not be increased further than 100 samples because of computational limitations.

Control horizon The above discussion already showed that the control horizon is better set independently from the prediction horizon. This decision depends on the system at hand. The combine harvester application contains many delays, which implies that the system output lags behind on the input. In other words, the effect of the last control values is not visible in the prediction horizon when both horizons are kept equal. If the effect of these control values has no impact on the output objective, which depends on the prediction horizon, there is no benefit in calculating these control values. After all, the control values are also the optimization values. This means that the computational complexity directly depends on the control horizon. The dependency of the computational complexity on the chosen prediction and control horizon is further elaborated in Chapter 9 and in Appendix C.

However, the solution to the optimization problem is not sensitive to the number of optimization variables. In other words, the first 5 control values will not change because the control horizon is increased from 5 to 6. There are two distinct cases:

- the speed error term is much larger than the input cost term (because there is a large error or a large error weight) and cannot be corrected in one time step (because of the input constraint). In this case the first element of the optimal input sequence will be the input constraint. In the capacity control application the input change would be limited to 5%.
- the speed error term and the input cost term are in the same order of magnitude. In this case a trade-off occurs between the error and the input effort. The required input correction will then be spread over several time steps.

In the first case the horizon does not matter. Even a control horizon $N_C = 1$ would yield the same result. In the second case the first element of the optimal sequence - which is the only element that matters - would be influenced if the next few elements are put to zero by limiting the control horizon. By using a control horizon of 5 elements, there is hardly any difference (as far as the first element is concerned) with a horizon of 10 or more elements. Of course, the control horizon should be balanced with the prediction horizon. There is not much use in predicting 10 seconds ahead if the system enters steady-state within 5 seconds.

7.2.2 Objective function weight factors

As mentioned earlier, it is not straightforward to find good values for the weight parameters of the different terms in the objective function. The order of magnitude of these weight factors is determined based on the variable characteristics listed in Table 7.1. During field test, the performance of several values in this order of magnitude is evaluated.

The range of the change of current is about 10 (due to the input constraints of the controller), and the range of feed rate in corn is (Table 7.1) about 2. In view of the

Table 7.3: The input effort (I_L and I_H) and the control error (E_L and E_H) (expressed in V) as a function of the weight of the control error (on feed rate) in the objective function. The weight of the input effort is kept constant (at 5) (F, 20051122a).

ID	Weight	Input effort		Control error	
	Control Error	I_L	I_H	E_L	E_H
1	30	0.107	0.031	0.068	0.036
2	50	0.101	0.055	0.065	0.023
3	70	0.064	0.031	0.067	0.025
4	100	0.142	0.061	0.099	0.026
5	150	0.177	0.061	0.127	0.027
6	200	0.120	0.052	0.110	0.031
7	300	1.232	0.261	0.219	0.038

factor 5 between both ranges, a logical starting point would be to put the change of current weight equal to 5 and the feed rate error weight to 25 (=5×5). However, it is clear that the feed rate signal is much noisier than the change of current signal since the latter is an input, and the former is a measurement.

During field tests (F, 20051122a) experiments were performed with different feed rate error weights (and fixed change of current weight). The results are listed in Table 7.3.

First of all, one should note that the field conditions (especially variation in crop density) are probably somewhat different in run 3 and run 6. The trend however is clear. All performance parameters reach an optimum in the region between weight 30 and weight 100.

It might seem strange that the control error increases if too much weight is put on the control error. This can be explained as follows: if the controller reacts too strongly to errors, it will overshoot. This will cause another controller intervention, and another overshoot. Even if those overshoots do not lead to instability, the control error as well as the input effort will go up. After all, if the controller is responding to the signal noise, it will always lag behind, causing the error to be larger than when the controller just followed the average signal.

7.2.3 Error dependent objective function

The purpose of any control engineer is to design a controller with the best possible performance. Nonlinearities cause the system amplification to vary depending on the working point. This creates an unpleasant dilemma:

- good disturbance rejection without instabilities (by using a conservative controller)

OR

- good transient response at crop entry (by using an aggressive controller)

For many process types this choice would be extremely easy. Chemical plants for instance are practically always in regime conditions. If transients occur, the controller can easily be switched to one with good transient response if this is required. A combine harvester capacity control system, however, continuously encounters both situations.

This section illustrates how disturbance rejection can be improved without jeopardizing the transient response. A straightforward solution is to make the objective function dependent on the control error. Use an aggressive controller far from the set point, and a more gentle controller near the set point. For instance, switch to the gentle controller if the feed rate is within 0.3 (Volt) of the set point (in corn). This is about 10% of the set point. Possible objective functions for both controllers are:

aggressive controller:

$$1.67 \sum_{k=1}^{k=30} (FR_k - REF_k)^2 + \sum_{j=0}^{4} (\Delta HF_j)^2 \tag{7.1}$$

gentle controller:

$$\sum_{k=1}^{k=30} (FR_k - REF_k)^2 + \sum_{j=0}^{4} (\Delta HF_j)^2 \tag{7.2}$$

with FR_k the feed rate level at time k and REF_k the feed rate reference at time k. ΔHF_j represents the change of the hydrostatic flow (the system input) at time j. The prediction and control horizon respectively are 30 and 5 samples. Note that the first controller puts more weight on the set point error than the second controller.

Both controllers were tested in the field (F, 20061107a). The system response as well as the system input are shown in Fig. 7.1 for one of the field tests. Note that the input swiftly increases until the process is in the neighborhood of the set point (here within a distance of 0.3V). At that moment a different, slower controller takes over. Note that one sees a mild deceleration at the moment the slower controller takes over. Due to the fact that the system contains many delay elements, actions of the previous, fast controller are still waiting to be executed. The MPC takes this into account and takes countermeasures according to its new, slower objective function. This deceleration is only visible in the data, but cannot be sensed on the machine. It could be avoided by gradually switching between both controllers.

If one tunes a controller for disturbance rejection (small deviations from the set point) and one uses the same controller for large errors (enter the crop for instance), this will lead to a too slow response. If one would tune the controller for a swift transient, this would lead to instability. This is caused by the uncertainties on the process model (the model gains as well as the delays), which make it impossible to have an equally fast controller near the set point, since this would lead to continuous oscillations. In the next chapter, Chapter 8, a better way will be presented to deal with this process uncertainty than switching the objective function as is presented in this section.

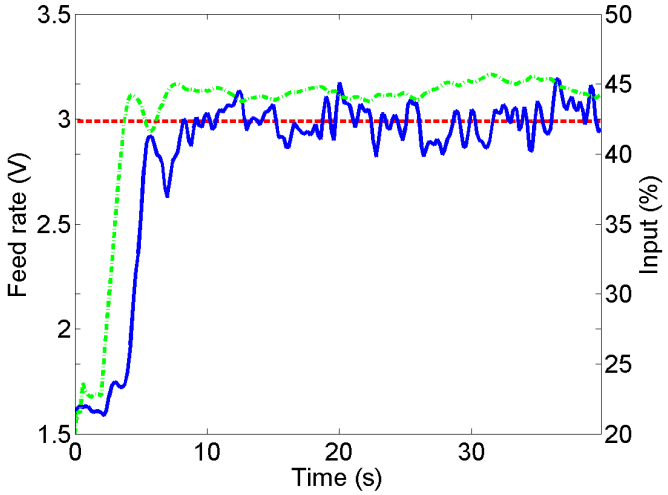


Figure 7.1: The behavior of the controller with an error dependent objective function. The solid line is feed rate, the dashed line is the feed rate set point (both on the left axis) and the dash-dot line is the input (on the right axis). Once the feed rate is within 0.3V of the set point, the controller clearly slows down because it switches to a different objective function (F, 20061107a).

7.2.4 Control rate

The control rate is the number of outputs per second produced by the controller. Note that this does not necessarily mean that the optimization problem needs to be solved at the same rate. Often the control rate will equal the data acquisition rate.

Since each MPC optimization problem leads to a solution vector, which contains an output for each time instant along the control horizon, one could take a single optimization solution to generate several control outputs.

There are actually two effects which determine the control rate:

- the control rate is often linked to the model sampling rate (discussed below) because the data acquisition rate is often the same as the control rate. In our case both rates can be chosen independently.

- the control rate needs to be sufficiently fast to correct the system without aggressively modifying the control values.

In these applications the control rate will be 5 Hz, except for some specific cases. 5 Hz is chosen because it is the standard sampling rate on the machine's CANbus system, and it is sufficiently fast to control the machine at such low speeds. To control the machine speed on the road (where the speed can of course be much higher), 20 Hz is compared to 5 Hz. The results for this application can be found in section 10.4.

7.3 Model configuration

In this section the different aspects of the successful configuration of a process model are discussed. Note that dynamical models, process delays and model parameters all influence the regularization of the controller if not chosen correctly. For instance, if a model gain is set to a larger gain than the real value, this will lead to a slower controller response. Thus it is very important to determine these configuration parameters in a rigorous manner.

7.3.1 Model sampling rate

The sampling rate of the model will influence the total number of states in the model, which on its turn will influence the calculation time needed to solve the optimization problem (Chapter 9). The computational complexity of the model update also depends on the number of states in the model (see section 7.3.5 and Appendix C).

It thus seems to be good practice to keep the model sampling rate as low as possible. The model sampling rate is determined by the fastest dynamics in the system. Note that it may very well be that some system component requires a 20 Hz sampling rate to be modeled, but that the system as a whole can be controlled at a much slower rate. This may lead to a 20 Hz model, which is then used in a controller with a 5 Hz control rate.

Three different scenarios for the model sampling rate were considered:

1 Hz sampling rate: the entire system is approximated by a 1 Hz model. 1 Hz is the optimal sampling rate for slow variables such as grain mass flow, but is actually a bit too slow for fast variables such as the speed. The slower model is however compensated by the control rate which is kept constant at 5 Hz. If the computational cost needs to be reduced, it is better to lower the model sampling rate than to lower the control rate.

5 Hz sampling rate: the entire system is modeled at 5 Hz. This is the optimal sampling frequency for the fast variables. The downside is that this leads to large matrices if grain mass flow is taken into account (this variables has a large delay). This model does not allow online model updates for grain mass flow control (see Appendix C) because of these large matrices. The model can however be used for feed rate or engine load control.

Mixed sampling rate: this is the best of both worlds. The slow variables are modeled at 1 Hz, and the fast variables are modeled at 5 Hz. Some adjustments to the model initialization are needed to enable such a split sampling rate (see Appendix C).

A controller with a 1 Hz model and one with a 5 Hz model are compared for feed rate control. In steady state there is no significant difference in performance for feed rate control between a 1 Hz and a 5 Hz model. However, if the disturbance rejection is examined, there is a clear difference, which is illustrated in Fig. 7.2 (F, 20061107b).

As noted in Chapter 3 the four-wheel drive of the combine can be engaged on-the-go, with a button in the cab. Engaging the four-wheel drive changes the system significantly because the oil of the hydrostatic system is suddenly distributed over the

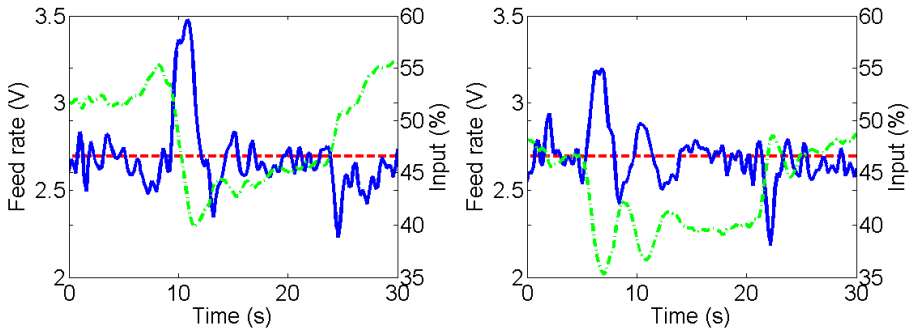


Figure 7.2: The solid line is feed rate, the dashed line is the feed rate set point (both on the left axis) and the dash-dot line is the input (on the right axis).
(a) Disturbance recovery of the controller with a 5 Hz model.
(b) Disturbance recovery of the controller with a 1 Hz model. Note that there is a disturbance after about 10 seconds (4x4 disengaged) and after approximately 25 seconds (4x4 re-engaged).

front and rear wheels. This causes the model gains to change instantaneously. Engaging and disengaging the four-wheel drive is a convenient way to induce a disturbance in the system.

In Fig. 7.2 the four-wheel drive is first disengaged (after approximately 10 seconds), which causes a sudden speed increase. After about 25 seconds the four-wheel drive is engaged again, causing the machine to slow down suddenly. Note that although the controller with the 5 Hz model remains perfectly stable, the controller with the 1 Hz model tends to oscillate mildly. The standard deviation of the control input the first 10 seconds after the disturbance is an interesting measure for controller performance:

- 5 Hz model: 0.811 (% input)
- 1 Hz model: 1.747 (% input)

As was already clear from the plot, the 5 Hz model leads to a better disturbance rejection than the 1 Hz model. The peak in feed rate is however larger for the controller with 5 Hz model. This is probably caused by the dip in feed rate just before the disturbance. Because of this, the controller was accelerating at the moment of the disturbance. The operator experienced the 5 Hz model controller as smooth during the disturbance recovery, whereas the oscillations of the 1 Hz model controller were clearly felt.

The mixed frequency approach allows to have the benefits of a 5 Hz sampling frequency (more robust to disturbances), without the computational cost. In this approach the delay towards grain mass flow is assumed a factor 5 smaller than it really is. Since there is no feedback included in the model, the delay towards grain mass flow control is not that relevant. The calculation of the initial state has to be changed to take the lower frequency of the grain mass flow model into account.

Table 7.4: The control error and the input effort for engine load control with three different parameters for the first order model (F, 20060906).

ID	α	Input effort		Control error	
		I_L	I_H	E_L	E_H
1	0.1	0.078	0.018	1.403	0.238
2	0.01	0.019	0.004	2.468	0.322
3	0.001	0.009	0.002	2.965	0.412

7.3.2 Effect of dynamical models

As described in the previous chapter (Chapter 6), the model of the combine contains some dynamics. The purpose of this section is to illustrate the effect of dynamic models on the performance of MPC.

To illustrate the effect of the dynamical model, let us consider engine load control. As shown before, there are dynamics between feed rate and engine load (a first order model):

$$EL_{k+1} = (1-\alpha)\, EL_k + \alpha u_k \tag{7.3}$$

where EL_k is the model value for engine load at time instant k, and u_k is the model input. α is the parameter of the first order model. The empirical value of this parameter is 0.1 (see Chapter 6). During field test the effect of the parameter of the model was examined (F, 20060906).

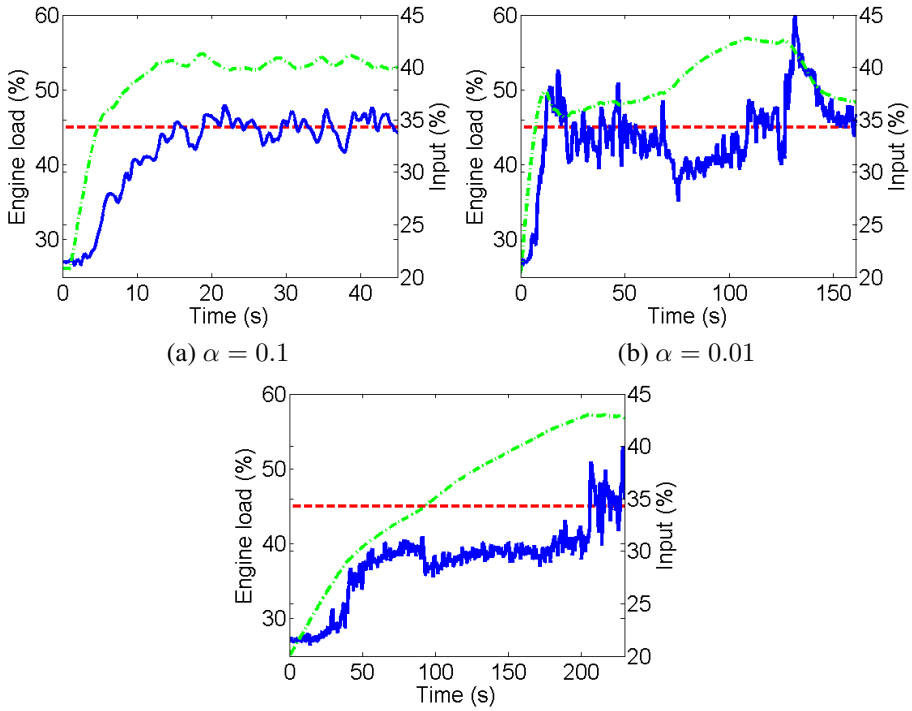
This parameter α can actually be explained as a regularization parameter. The smaller the value α is, the longer it will take before the controller responds. Such a model can in other words also be used to avoid instability on systems with variable time delays. Including a model with low-pass characteristics allows less conservative tuning of the controller.

In Table 7.4 the control error and the control effort (both during steady state) are shown for three values of α, namely 0.1, 0.01 and 0.001. As expected the input effort decreases with decreasing α, and the control error increases with decreasing α. Note that this effect is visible in both the low frequencies and the high frequencies. An example of a step response for each of the three controllers is shown in Fig. 7.3. As discussed in Chapter 6, the actual engine load parameter α is about 0.1. This value will be used in further tests.

7.3.3 Delays

The combine contains many delays, which also vary depending on the conditions. It is thus very important to examine the robustness of the MPC relative to changing delays. This is done by varying the delay parameters of the model in the field.

As discussed in section 6.3 of Chapter 6, there is a delay between pump setting and ground speed in some conditions. During field test (F, 20061030a) this delay was included in the model and varied to examine the effect of this delay on the controller performance.



(a) $\alpha = 0.1$ (b) $\alpha = 0.01$

(c) $\alpha = 0.001$

Figure 7.3: The solid line is engine load, the dashed line is the engine load set point (45%) (both on the left axis) and the dash-dot line is the input (on the right axis). The step response is given for three different values of α.

Table 7.5: This table shows the rise time (seconds), the input effort (I_L and I_H) and the control error (E_L and E_H) for different model delays between hydrostatic flow and ground speed (F, 20061030a).

ID	Delay (samples)	Rise time (seconds)	Input effort		Control error	
			I_L	I_H	E_L	E_H
1	0	1.2	1.664	0.302	0.307	0.036
2	1	1.4	0.799	0.193	0.153	0.043
3	2	1.6	0.486	0.149	0.161	0.030
4	3	3.6	0.322	0.137	0.139	0.029
5	4	4.2	0.255	0.136	0.106	0.029

Effect of delays In Table 7.5 the impact of variation of the delay on the controller performance is illustrated. It is clear that introducing this delay is a way of adding extra regularization. As one can see, I_L, the slow input effort, decreases with increasing delays, which is consistent with increasing regularization. The high-frequency control error decreases till 0.03, and then remains constant. In previous experiments, such as for example in Table 7.3, it already became clear that the noise level of feed rate (the lower bound of the high frequency control error) is about 0.03 V.

The most important measures to quantify controller performance are the rise time, the high frequency input effort (I_H, which is uncomfortable for the operator) and the low frequency error (E_L). As can be seen in Table 7.5, the rise time increases (the controller becomes slower) and the input effort and the control error decrease as the delay is increased. If even higher delays would be examined, the control error would increase again.

Because of the increasing rise time, a delay of 3 samples is regarded as the best choice for this application. It is clear that this is a trade-off between the different measures. One could also tune the controller more conservatively instead of including this delay.

Robustness to varying delays Often delays are very hard to determine, depend on the plant state, or just drift in time (depending on input material,...). One of the challenges of a real-life application with many delays is to deal with this uncertainty. This will be further elaborated in Chapter 8.

7.3.4 Importance of the adaptive model

Correct values for the system parameters are important for the controller performance. The objective of this section is to quantify the impact of varying model parameters, assuming that the controller model is constant. This will give an idea of the performance gain realized by making the model adaptive.

Since the system parameters cannot be controlled during field tests, the effect is demonstrated on simulation. For this simulation the same model as for the controller was used. Noise was added to all measurements to better approximate reality. Of course, this simulation does not include the effects of biological variability.

Three 'crop entries' were simulated with different gains for the ground speed-feed rate relation. The controller gain α is set to 0.04, because this is a value which often occurs in practice. The response of the controller is simulated for model gains β equal to 0.5α, α and 2α (F, 20080613). This is a realistic situation since these gains also occur in field conditions (Chapter 6).

The simulated response is shown in Fig. 7.4. The difference between the controller parameter α and the simulation parameter β can be summarized as follows:

$\beta < \alpha$**:** the controller becomes significantly slower, but remains stable.

$\beta > \alpha$**:** the controller becomes faster, which often lead to oscillations in the step response. In some cases the controller can even become unstable.

It is clear that if the purpose is to make the controller performance independent of the field conditions, it is necessary to use an adaptive model. This also renders

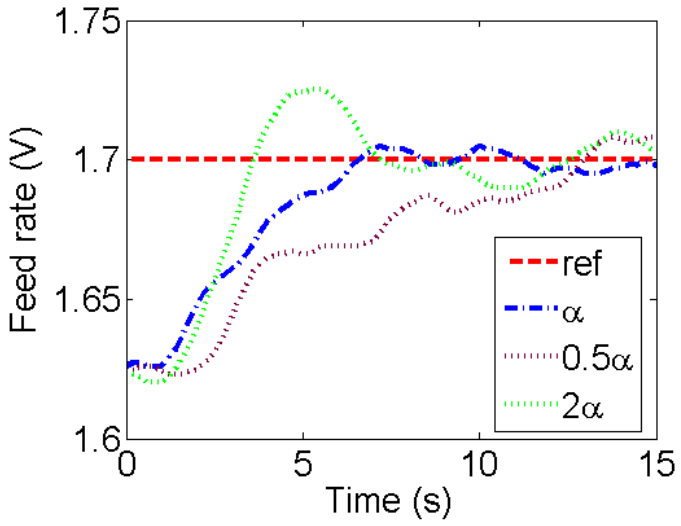


Figure 7.4: Simulated crop entry response for 3 model gains β: α, 0.5α and 2α.

the controller independent of the crop type (canola, wheat, corn...) and the machine specifications.

7.3.5 Model update rate

As discussed in Chapter 6 the model is adaptive, and thus the model matrices need to be modified online. Updating these model matrices is a significant computational cost. The calculation time depends on the control horizon, the prediction horizon and the number of constraints. Without special precautions, it takes about 1.5 seconds to update a model for feed rate control (prediction horizon 30 and control horizon 5). By using a lower sampling rate for the variables with large delays (i.e. grain mass flow), the calculation time to update the model can be lowered to less than 200ms. Details on the computational complexity of the model update can be found in Appendix C.

There are several ways to determine when to update the model:

fixed update rate: the model can be updated at a fixed rate, for instance once a second.

significant variation: the model can be updated each time the model parameters vary more than a given threshold, possibly with a fixed minimal update rate.

In this research the first method was used, because this best ensures repeatability of the results. For a commercial implementation, the second option would be preferred, since this allows to lower the update rate even further.

7.4 Constraints

7.4.1 Constraint definition

A capacity control system needs to take many constraints into account. The most important constraints are:

- input constraints
  - change of input current, e.g. a maximum change of $\pm$ 5% current
- state/output constraints
  - pump setting, between 0 and 100%
  - ground speed (limitation on the operator and the cutting mechanism of the header), e.g. between 1 and 10 km/h, depending on the type of header
  - feed rate (header and threshing process limitation), e.g. between 0.2 and 2V, depending on the header
  - engine load (threshing limitation), lower than 100%
  - grain mass flow (cleaning limitation), typically lower than 2V
  - losses (economical and cleaning limitation), e.g. less than 100 impacts per 200ms

All these constraints have been tested in the field, except for losses. As mentioned in Chapter 3 the sieve losses are very hard to measure. The standard loss sensor in fact only gives a qualitative indication of the actual loss level. Once better sensors are available to measure the losses, this constraint can also be included in the capacity control system.

7.4.2 Feasibility

Adding constraints implies that feasibility needs to be taken into account. Due to the presence of disturbances it cannot be guaranteed that the constraints are satisfied at each time instant. Since an ASM is used to solve the optimization problem, a feasible starting point for the optimization - which may not exist - is required at the start of each optimization (see Chapter 9).

This implies that the optimization problem has to be redefined such that the existence of a feasible starting point can be guaranteed. This implies that the constraints will have to be modified. The solution method then requires the solution of two optimization problems:

1. *Determine the initial value and the constraints:* a new optimization problem is defined to calculate a feasible initial value with minimal modifications to the inequality constraints.

2. *Determine the optimal solution:* the optimization problem is solved as before, starting from the initial value and the constraints calculated in the first step.

Assume that the constraints of the optimization problem can be written as follows:

$$\begin{cases} \mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{C}\mathbf{x}_0 + \mathbf{D}\mathbf{u} \leq \mathbf{b} \end{cases} \tag{7.4}$$

where $\mathbf{x}$ and $\mathbf{u}$ are the following stacked vectors:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_{N_P} \end{bmatrix} \qquad \mathbf{u} = \begin{bmatrix} \mathbf{u}_0 \\ \vdots \\ \mathbf{u}_{N_C} \end{bmatrix} \tag{7.5}$$

with N_P is the prediction horizon and N_C is the control horizon. $\mathbf{x}_0$ is the measured (or estimated) state. All the other vectors $\mathbf{x}_k$ and $\mathbf{u}_k$ are optimization variables.
The stage-1 optimization problem then becomes:

$$\begin{gathered} \min_{\mathbf{u}} f(\mathbf{p}) \\ \begin{cases} \mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{C}\mathbf{x}_0 + \mathbf{D}\mathbf{u} \leq \mathbf{b} + \mathbf{T}\mathbf{p} \end{cases} \end{gathered} \tag{7.6}$$

with $\mathbf{p}$ the slack variables and $\mathbf{T}$ a weight matrix. This weight matrix makes the trade-off between the different constraints. This is especially important if not all the variables have the same range. $f(\mathbf{p})$ expresses the cost of a constraint violation. $\mathbf{p}$ can take one of two forms:

- **Notation 1:** different amount of slack on each constraint

$$\mathbf{p} = \begin{bmatrix} \alpha \\ \beta \\ \vdots \\ \xi \end{bmatrix} \tag{7.7}$$

  where each Greek letter represents an independent variable.

- **Notation 2:** the same amount of slack on each constraint

$$\mathbf{p} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \alpha \tag{7.8}$$

  with α the amount of slack.

The function $f(\mathbf{p})$ can be any function of the vector of slack variables $\mathbf{p}$. Often it is one of the following norms:

1-norm: the sum of the absolute values of the slacks is minimized ($\mathbf{p}$ in notation 1)

2-norm: the sum of the squares of the slacks is minimized ($\mathbf{p}$ in notation 1)

∞-norm: the maximal value of the slacks is minimized ($\mathbf{p}$ in notation 2)

Field tests have shown that the system parameters are too uncertain to be able to impose constraints without paying attention to the parameter variability. In Chapter 8, constraints will be added to the Stochastic MPC controller.

7.5 Benchmark

7.5.1 Basic control

For some capacity control systems, basic control methodologies are valid alternatives. For speed control and feed rate control there is no reason why basic control techniques such as PID would be by definition unsuitable.

During this research several feed rate control structures were tested, and one of those controllers was eventually implemented in the commercial combine harvester as IntelliCruise (launched at Agritechnica 2007, Hannover, Germany W, Agritechnica (2007)). Two controllers that were tested on the combine harvester, but not implemented in the commercial software are presented in this section as benchmark.

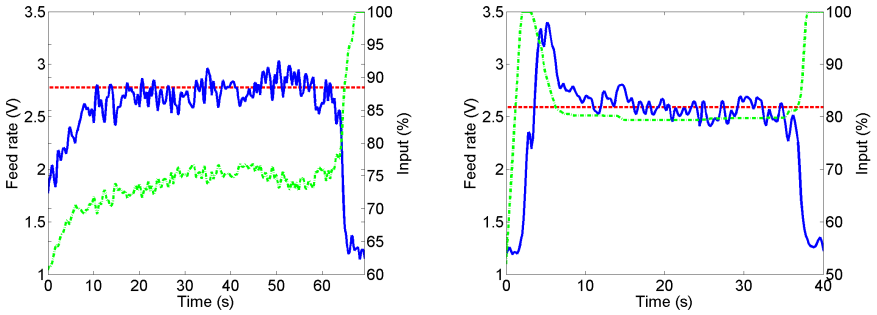
Grain mass flow control, and to a lesser extent engine load control, struggle with considerable time delays between the control signals (pump setting) and the process response. As discussed in Chapter 6 the delay between speed changes and grain mass flow is more than 10 seconds. A system with such large delays is very hard to control with basic control theory. There are some more advanced techniques, such as Smith predictors (Franklin et al., 2006), which can deal with delays. All these techniques, however, have one thing in common: they require a high quality model and a good estimate of the delay. These are two things which are not available due to the biological variability, such as the variation of the crop density, grain-straw ratio and other crop properties.

In the next section some results of two feed rate controllers which have been tested on the combine harvester will be shown, namely a PID controller and a fuzzy controller. These controllers can serve as a benchmark for the performance of the MPC controllers to be developed. Note though that some requirements listed in Chapter 4, such as a flexible objective and dealing with constraints, are very hard to fulfill by this type of controllers.

PID feed rate control The performance of the PID controller is assessed by applying a step to the controller reference. The performance measures are shown in Table 7.6. The performance in terms of input effort and control error is more or less constant across the 6 runs. The PID controller is tuned to have the lowest possible control error in steady state and a minimal rise and settling time. Ziegler-Nichols (Franklin et al., 2006) is used to provide an initial guess of the controller parameters. The controller performance is then further improved by modifying the parameters by trial and error.

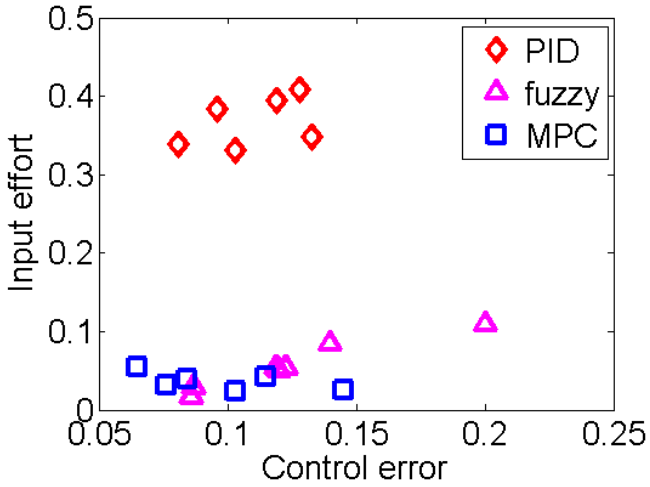
There is a trade-off between control error and control effort. The tuning of the PID controller (and of the fuzzy controller in the next paragraph) focused on the steady state situation, and did not consider crop entry or large set point changes. An example of crop entry with the PID controller is shown in Fig. 7.5 (a).

Fuzzy feed rate control The fuzzy controller was tuned according to the same specifications as the PID controller. This way the performance of both controllers can easier be evaluated in terms of operator comfort. Just as with the PID controller, the focus lays on steady state error. The rule table of this fuzzy controller is shown in Table 7.7. The controller consists of a proportional and derivative part. The linguistic variables error and change of error have a trapezoid membership function. The different rules



(a) PID controller

(b) Fuzzy controller



(c) Performance comparison

Figure 7.5: The response of the PID controller (a) and the fuzzy controller (b) at crop entry. The solid line is the feed rate measurement, the dashed line is the set point (both on the left axis) and the grey dash-dot line is the input (right axis).
(c) The input effort and the control error of Table 7.6, Table 7.8 and Table 7.9 for PID control, fuzzy control and MPC.

Table 7.6: The input effort and control error for tests with a PID feed rate controller (F, 20051103a and F, 20051102).

ID	Input effort		Control error	
	I_L	I_H	E_L	E_H
1	0.262	0.339	0.081	0.029
2	0.362	0.408	0.128	0.034
3	0.338	0.394	0.119	0.034
4	0.322	0.383	0.096	0.032
5	0.328	0.347	0.133	0.029
6	0.343	0.331	0.103	0.030
Average	0.326	0.367	0.110	0.031
Median	0.333	0.365	0.111	0.031

Table 7.7: Rule table of the fuzzy controller. Δ error means change of error (the derivative). '- -' indicates a large negative (change of) error, '-' a negative (change of) error, '0' zero (change of) error, '+' a positive (change of) error and '++' a large (change of) error. The table lists the output for each input combination. Analogous to the error, the symbols represent a (large) positive/negative change of the output. '0' means that the output remains constant.

		error				
		–	-	0	+	+
Δ error	-	++	++	+	0	-
	0	++	+	0	-	- -
	+	+	0	-	- -	- -

are aggregated based on the rule membership. For instance if the variable is for 60% in rule 1 and 40% in rule 2, the output of the controller is 60% of the output of rule 1 and 40% the output of rule 2 (Passino and Yurkovich, 1998).

A small output correction is an increase/decrease of the pump setting with 1%. A large correction is a 5% step. Crop entry for the fuzzy controller is shown in Fig. 7.5 (b). The fuzzy controller shows a quite large overshoot at entry, but this feels actually very smooth and comfortable to the operator. This controller enables the machine to enter very swiftly, without becoming unstable.

The performance measures for the fuzzy controller are shown in Table 7.8. These performance measures only take the steady state part into account. The control error is indeed similar to the control error of the PID controller (slightly larger than for the PID), but the input effort is significantly smaller. Both control structures, and the MPC, are compared in terms of input effort and control error in Fig. 7.5 (c). Special precautions, such as a dead zone, could be added to the PID controller to reduce the input effort without influencing the control error.

Table 7.8: The input effort and control error for tests with a fuzzy feed rate controller (F, 20051103b).

ID	Input effort		Control error	
	I_L	I_H	E_L	E_H
1	0.102	0.051	0.120	0.028
2	0.160	0.084	0.140	0.033
3	0.112	0.053	0.119	0.031
4	0.050	0.029	0.087	0.028
5	0.323	0.109	0.200	0.041
6	0.034	0.017	0.086	0.028
7	0.091	0.053	0.123	0.028
Average	0.125	0.057	0.125	0.031
Median	0.102	0.053	0.120	0.028

Table 7.9: The input effort and control error for tests with a Model-based Predictive Controller (F, 20051122b and F, 20051123b).

ID	Input effort		Control error	
	I_L	I_H	E_L	E_H
1	0.132	0.042	0.115	0.023
2	0.078	0.026	0.145	0.032
3	0.101	0.055	0.065	0.023
4	0.071	0.032	0.076	0.026
5	0.071	0.024	0.103	0.025
6	0.075	0.039	0.084	0.030
Average	0.089	0.036	0.099	0.027
Median	0.076	0.035	0.094	0.026

Conclusion These controllers have been evaluated by professional operators with satisfactory results. However, quite a few requirements of Chapter 4 have not yet been met, such as a flexible objective, and dealing with constraints. It will become clear that, apart from the performance, MPC has some advantages over these basic controllers.

7.5.2 MPC

As pointed out above, the main purpose of using MPC is not to increase performance as such, but to satisfy additional requirements, such as allowing a flexible objective, and dealing with constraints. In this section the performance of MPC is benchmarked against the performance of the PID and fuzzy controllers of the previous section. Field experiments were performed with the feed rate error weight equal to 50 and the input weight equal to 5. This is one of the best configurations of Table 7.3. The prediction horizon N_P is 20 samples. ($N_C = N_P$)

The results of six similar field tests (F, 20051122b and F, 20051123b) are listed in Table 7.9. The performance of the MPC is slightly better than the performance of the PID and the fuzzy controller, since the average input effort as well as the average control error are lower. This is illustrated in Fig. 7.5.

7.6 Conclusions

This chapter discusses the important tuning parameters of a Model-based Predictive Controller. First of all the influence of the control and the prediction horizon is established. Next it is shown how the configuration of the objective function defines the trade-off between a conservative, 'careful' controller and a more aggressive controller. The benefits of making this trade-off dependent on the circumstances, is illustrated with field tests of an error-dependent objective function. In the next chapter the trade-off between control error and input effort will be made dependent on the model uncertainty.

The sampling rate of the model has a considerable impact on the computational complexity of the optimization. Field tests show that the sampling rate has to be kept at 5 Hz for the fast variables such as feed rate, but can be lowered without performance loss for a slow variable such as grain mass flow.

It is shown that model parameter errors influence the regularization, or in other words, the relative weight on the error. The objective to design a controller which has the best possible performance in all circumstances can thus only be reached by making the model adaptive, thus avoiding inaccurate model parameters. Otherwise the regularization of the system would vary depending on the parameter error. For this system adding constraints is only possible by making the controller very slow and conservative. A solution may be to modify the controller when it comes near the constraints. This will be tackled in the next chapter.

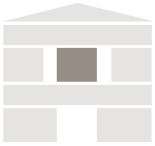
MPC is benchmarked with two other controllers (PID and fuzzy). Next to the clear MPC benefits such as variable objective function and constraint handling (Chapter 8), MPC has a lower input effort and a lower control error than the PID and fuzzy controllers.

The next chapter will present a methodology to correct the objective function automatically for the varying model uncertainty. The constraints will also be modified to take the model uncertainty into account.

Chapter 8

Stochastic MPC

Maturity is the capacity to endure uncertainty
-John Finley-



MPC is a model-based control strategy and like most model-based strategies, the quality of the controller depends on the quality of the model. For MPC especially the prediction behavior of the model is important. The model of the combine harvester process can never be exact due to the natural variability of the process input. Model uncertainty thus plays an important role. Several ways to deal with uncertainty were described in section 2.3 of Chapter 2. All these approaches assume, however, that the plant model is confined within some bounds, without taking the probabilistic aspect of the uncertainty into account.

In most cases all the possible plant models are not equally probable. The probability distribution of the estimated model parameters can be used to calculate a probability distribution for each of the model states. Each state thus becomes a stochastic variable. The expected value of this stochastic variable is defined by the process model, whereas the state uncertainty depends on the parameter uncertainty through an uncertainty model.

The objective of this chapter is to include the available stochastic information of the model parameters in the control methodology, by applying and extending the Stochastic MPC framework. This approach is then validated on the combine harvester for capacity control. Results of this chapter are published as Coen et al. (2009).

8.1 State of the art

Until recently, very little attention had been paid to this case of stochastic uncertainty. Yet it arises naturally in many control applications, for example when models are identified using least squares procedures. Several authors have proposed formulations to include the randomness of processes in MPC (Kouvaritakis, 2000a, Van Hessem and Bosgra, 2002, Couchman et al., 2005, Kendrick, 2005, Batina, 2004). All the formulations listed below, as well as the approach taken in this research, assume that the state uncertainty is normally distributed.

Kouvaritakis et al. One of the first authors to venture in the field of Stochastic MPC was Kouvaritakis (2000a) (Kouvaritakis et al., 2004, 2006). Stochastic uncertainty was used to describe a key area of human endeavor, namely sustainable development. This was one of the first research fields where Stochastic MPC received attention. Sustainability refers to the strategy of encouraging development at the current time without compromising the potential for development in the future. A very considerable random element is involved when modeling the effects of a policy over a horizon of 30 years. These random effects have to be taken into account when assessing the optimality of any proposed policy.

MPC is ideally suited for generating constrained optimal solutions and as such would be an ideal tool for policy assessment. However, this first calls for suitable extensions to deal with the probabilistic case. After all, although early development of MPC was specifically targeted to handle random processes (Astrom and Wittenmark, 1973, Clarke and Gawthrop, 1975), the formulation of MPC has been predominantly deterministic.

In order to be able to assess different policies, Kouvaritakis (2000a) developed an econometric model called Prometheus[1]. It models the impact of budget allocation to different technologies on eight indicators (such as CO_2 emission). $\psi_i, i = 1 \dots 8$ represent the cumulative values of these indicators over a 30 year horizon. On the basis of thousands of Prometheus simulations (excluding all outliers and realizations of the random variables that violate economic theory), it was possible to model the dependence of these indicators on instruments $u_j, j = 1 \dots 15$ as

$$\psi_i \quad = \quad \mathbf{r}_i^T \mathbf{u}, \qquad \mathbf{u} = \begin{bmatrix} u_1 & u_2 & \dots & u_{15} \end{bmatrix}^T \tag{8.1}$$

where

$$\mathbf{r}_i \quad \propto \quad \mathcal{N}\left(\boldsymbol{\rho}_i, \mathbf{V}_i\right), \qquad i = 1, \dots, 8 \tag{8.2}$$

with $\mathbf{r}_i$ a stochastic parameter vector which is normally distributed with mean vector $\boldsymbol{\rho}_i$ and covariance matrix $\mathbf{V}_i$.

A deterministic optimization problem can only be posed in terms of the mean response of the model described by Eqns (8.1) and (8.2), which would strip the problem of its probabilistic nature. Instead, in Kouvaritakis (2000b) the following optimization is proposed:

$$\max_{\mathbf{u}} \Pr\left\{\psi_1 > A_1\right\} \tag{8.3}$$

subject to

$$\Pr\left\{\psi_i > A_i\right\} \quad \geq \quad p_i \qquad i = 2, \dots, 8 \tag{8.4}$$

[1] Prometheus is a full econometric model, developed by Kouvaritakis (2000a), which models the effect of shocks (the discrete time equivalent of impulse changes to the common instruments superimposed on a baseline) applied to the budget allocations for each of 15 alternative technologies (wind turbines, solar thermal power,...) on eight measurable indicators (CO_2 emissions all over the world, total energy cost,...). Monte Carlo simulations, which used Prometheus to predict the effect of impacts on indicators at the end of a 30-year horizon, established the validity of a linear model in which the impact coefficients were random variables.

The optimal input sequence $\mathbf{u}$ is thus calculated by maximizing the probability that the indicator ψ_1 is larger than a given value A_1, while respecting the constraint that all other indicator values ψ_i have to remain larger than a given minimal value A_i with a probability larger than p_i. The optimization problem of Eqns (8.3) and (8.4) is equivalent to (Kouvaritakis et al., 2004):

$$\min_{\mathbf{u}} s \tag{8.5}$$

subject to

$$\begin{array}{rcll} \frac{A_1-\boldsymbol{\rho}_1^T\mathbf{u}}{\sqrt{\mathbf{u}^T\mathbf{V}_1\mathbf{u}}} & \leq & s & \\ \frac{A_i-\boldsymbol{\rho}_i^T\mathbf{u}}{\sqrt{\mathbf{u}^T\mathbf{V}_i\mathbf{u}}} & \leq & a_i & i=2,\ldots,8 \end{array} \tag{8.6}$$

Note that this is not a QP, and requires more advanced and computationally more expensive solution methods.

Van Hessem et al. Van Hessem and Bosgra (2002) proposed a new methodology to include the influence of stochastic disturbances. The key aspect of their approach is the simultaneous optimization over a reference trajectory (steering task) and a feedback map (controlling task) while dynamically sizing back-off to the state and the input constraints. This approach differs from classic MPC by explicitly predicting the closed-loop behavior of the plant under the influence of stochastic disturbances.

Van Hessem points out that given an uncertain model, MPC will always drive the process through the constraints. A receding horizon strategy cannot prevent this phenomenon since it is always one step too late. The single method to avoid this behavior is to take future disturbance realizations into account by guarding back-off to the constraints.

The objective is to optimize a function $f(\cdot)$ of the performance output $\mathbf{z}$, where $\mathbf{z}$ must be satisfy the following constraint:

$$\Pr\{\mathbf{z}\in\mathcal{P}\} \quad \geq \quad \alpha \tag{8.7}$$

with α a user chosen level, and $\mathcal{P}$ defined as:

$$\mathcal{P} \quad = \quad \left\{\boldsymbol{\xi} : \mathbf{H}^T\boldsymbol{\xi} \leq g\right\} \tag{8.8}$$

This means that the probability that the performance output $\mathbf{z}$ is contained within the region $\mathcal{P}$ needs to be larger than α. Assuming that $\mathbf{z}$ is normally distributed with mean $\bar{\mathbf{z}}$ and covariance matrix $\mathbf{Z}$, this probability constraint is equivalent to the following integral constraint:

$$\frac{1}{\sqrt{(2\pi)^n \det(\mathbf{Z})}} \int_{\mathcal{P}} e^{-\frac{1}{2}(\boldsymbol{\xi}-\bar{\mathbf{z}})^T\mathbf{Z}^{-1}(\boldsymbol{\xi}-\bar{\mathbf{z}})} \mathrm{d}\boldsymbol{\xi} \quad \geq \quad \alpha \tag{8.9}$$

with $\bar{\mathbf{z}}$ and $\mathbf{Z}$ respectively the expected value and the covariance matrix of the performance output $\mathbf{z}$, and $\mathcal{P}$ the allowed region as defined in Eqn (8.8). Note that the

covariance matrix $\mathbf{Z}$ depends on the feedback map which is calculated online. Van Hessem then defines the following ellipsoid $\mathcal{E}_r$:

$$\mathcal{E}_r = \left\{ \boldsymbol{\xi} : \boldsymbol{\xi}^T \mathbf{Z}^{-1} \boldsymbol{\xi} \leq r^2 \right\} \tag{8.10}$$

with r chosen such that:

$$\alpha = \frac{1}{2^{n_z/2}\Gamma\left(n_z/2\right)} \int_0^{r^2} s^{\frac{n_z}{2}-1} e^{-\frac{s}{2}} \mathrm{d}s \tag{8.11}$$

with s the integration variable and $\Gamma\left(n_z/2\right)$ the gamma distribution function (Nguyen and Rogers, 1989). The shape of this ellipsoid depends on the feedback map (controlling task), and the position of the ellipsoid (see below) depends on the reference trajectory (steering task). If we can now make sure that

$$\bar{\mathbf{z}} + \mathcal{E}_r \subset \mathcal{P}, \tag{8.12}$$

Eqn (8.7) is implicitly satisfied. The ellipsoid represents the back-off from the constraints the controller has to respect. This back-off depends on the variability (i.e. the covariance) of the performance vector. An application of this methodology is shown in Van Hessem and Bosgra (2006).

Couchman et al. Couchman assumes a system model where the state equations are deterministic, and the matrices of the output equations are stochastic (Couchman et al., 2005, 2006):

$$\left\{ \begin{array}{lcl} \mathbf{x}_{k+1} & = & \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \\ y_{1,k} & = & \mathbf{c}_{1,k}\mathbf{x}_k \\ y_{2,k} & = & \mathbf{c}_{2,k}\mathbf{x}_k \end{array} \right. \tag{8.13}$$

with

$$\mathbf{c}_{1,k} \propto \mathcal{N}\left(\bar{\mathbf{c}}_1, \boldsymbol{\Theta}_{\mathbf{c}_1}\right) \tag{8.14}$$

$$\mathbf{c}_{2,k} \propto \mathcal{N}\left(\bar{\mathbf{c}}_2, \boldsymbol{\Theta}_{\mathbf{c}_2}\right) \tag{8.15}$$

Note that y_1 and y_2 are scalar outputs. Scalars are used to keep the notation simple. Extra outputs can be defined in the same way as y_1 and y_2. Even in steady state, the output is still a random variable. The randomness is thus not related to the input.

In deterministic tracking the cost is simply defined as the distance between the current value and the target value, in order to drive the output y_1 to the reference r. Since y_1 is a stochastic variable, the aim will be to drive, in a probabilistic sense, $y_{1,k}$ to y_1^{ss}. y_1^{ss} is the steady-state value with $y_1^{ss} = \mathbf{c}_{1,k}\mathbf{x}^{ss}$. This objective can be achieved through enveloping:

$$\begin{array}{c} \Pr\left\{y_{1,k} - y_1^{ss} \geq -t_k\right\} \geq p_1 \\ \Pr\left\{y_{1,k} - y_1^{ss} \leq t_k\right\} \geq p_1 \end{array} \tag{8.16}$$

Satisfaction of Eqn (8.16) ensures that the random variable $y_{1,k} - y_1^{ss}$ lies with a probability of at least p_1 in each of the regions $[-\infty, t_k]$ and $[-t_k, \infty]$, and hence with probability of at least $2p_1 - 1$ in the interval $[-t_k, t_k]$.

In this setting the best result that can be achieved is to drive the probabilistic bounds t_k, and thus the zone around the set point $[-t_k, t_k]$, to zero. A good choice for the MPC performance cost thus is $J = \sum_k [t_k]^2$. $\mathbf{v}_k$ is substituted according to the following relation:

$$\mathbf{v}_k = \mathbf{x}_k - \mathbf{x}^{ss} \tag{8.17}$$

The bounds t_k are then given by (Couchman et al., 2006):

$$t_k \quad = \quad |\bar{\mathbf{c}}_1^T \mathbf{v}_k| + k_1 \sqrt{\mathbf{v}_k^T \mathbf{\Theta}_{\mathbf{c}_1} \mathbf{v}_k} \tag{8.18}$$

where $\mathfrak{N}(k_1) = p_1$[2].

The parameter p_1 affects the importance placed on the mean cost relative to the variation around the mean cost. The same approach can be taken to implement constraints. Often probabilistic constraints are placed on the secondary output y_2:

$$\Pr\{y_{2,k} - y_2^{ss} \leq L\} \quad \geq \quad p_2 \tag{8.19}$$

where L is the secondary constraint threshold. Define k_2 such that $\mathfrak{N}(k_2) = p_2$. The constraint of Eqn (8.19) can then be rewritten as:

$$L \quad \geq \quad \bar{\mathbf{c}}_2^T \mathbf{v}_k + k_2 \sqrt{\mathbf{v}_k^T \mathbf{\Theta}_{\mathbf{c}_2} \mathbf{v}_k} \tag{8.20}$$

Note that this approach does not lead to a QP, but to a SOCP (Second Order Cone Program). The computational cost to solve a SOCP is considerably higher than the cost to solve a QP.

Conclusion The previous paragraphs illustrate the current state of the art in the field of Stochastic MPC. The leitmotiv is the same for all authors, but there are some differences which depend on the background and/or the application at hand. In the next paragraphs the approach taken in this research is presented. It leans towards the approach by Couchman et al., but is not entirely the same. One of the main differences is the place in the system where the uncertainty is introduced. Couchman places the uncertainty in the output matrix $\mathbf{C}$, whereas in this research the uncertainty is actually located in the input matrix $\mathbf{B}$, and to a lesser extent in the state matrix $\mathbf{A}$. This implies that the uncertainty does not only affect the output, but the internal states as well.

8.2 New method

8.2.1 Assumptions

Several assumptions are made about the system for which the control system is being designed:

- the model uncertainty is solely caused by input changes. In other words, the influence of disturbances on model uncertainty is not included.

[2] $\mathfrak{N}(\cdot)$ is the cumulative distribution function of the normal distribution, thus $\mathfrak{N}(x) = \Pr\{z \leq x\}$ with $z \propto \mathcal{N}(0, 1)$, i.e. z standard normally distributed (mean 0 and variance 1)

- the parameter error is assumed to be normally distributed. This does not mean that the parameter itself needs to be normally distributed, but it requires that the correct estimation technique is used to ensure that the error is normally distributed.
- the errors on the uncertain model parameters are statistically independent.
- the parameter uncertainty is only present in the forward path. In other words, none of the state equations has a term in the product of an uncertain parameter and a state which depends on that same uncertain parameter. This limitation is no problem for the applications considered in this thesis.
- the state equations are linear in the uncertain model parameters.

8.2.2 System definition

Assume the following general system definition:

$$\mathbf{x}_{k+1} \quad = \quad f\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{p}\right) \tag{8.21}$$

where $\mathbf{x}_k$ is the state vector at time step k, $\mathbf{u}_k$ is the input vector at time step k and $\mathbf{p}$ is the parameter vector at time step k. In this work the parameter vector is assumed constant across the prediction horizon, and can thus be written as $\mathbf{p}$. Assume that all states are measurable and that the output vector $\mathbf{y}$ is equal to the state vector:

$$\mathbf{y}_k \equiv \mathbf{x}_k \tag{8.22}$$

The parameter vector $\mathbf{p}$ contains the different model parameters, and is normally distributed with mean $\bar{\mathbf{p}}$ and covariance matrix $\mathbf{\Sigma}_p$. $\bar{\mathbf{p}}$ is calculated by the gain estimator described in section 6.5 of Chapter 6. In order to guarantee that the modeling error is normally distributed, Fuzzy RLS is used as gain estimation technique. The procedure to estimate the covariance matrix $\mathbf{\Sigma}_p$ is discussed in section 8.4.

$$\begin{aligned} \mathbf{p} \propto \mathcal{N}\left(\bar{\mathbf{p}}, \mathbf{\Sigma}_p\right) \qquad \mathbf{p} &= \begin{bmatrix} K_1 \\ K_2 \\ \vdots \\ K_r \end{bmatrix} \\ \bar{\mathbf{p}} = \begin{bmatrix} \bar{K}_1 \\ \bar{K}_2 \\ \vdots \\ \bar{K}_r \end{bmatrix} \qquad \mathbf{\Sigma}_p &= \begin{bmatrix} \sigma^2_{K_1} & 0 & \dots & 0 \\ 0 & \sigma^2_{K_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma^2_{K_r} \end{bmatrix} \end{aligned} \tag{8.23}$$

with r the number of model parameters, K_1, K_2,...,K_r the model parameters and $\mathbf{\Sigma}_p$ the covariance matrix. All the off-diagonal elements in the covariance matrix are zero because of the assumption that the model parameters are statistically independent. In practice the expected value of each model parameter is itself time-variable. However, we can assume that the online calculated model parameter is a good estimate of the real model parameter value. Moreover, the model parameters can be assumed to be

more or less constant over the prediction horizon. The system of Eqn (8.21) can be written as:

$$\mathbf{x}_{k+1} = \mathbf{A}(\mathbf{p})\,\mathbf{x}_k + \mathbf{B}(\mathbf{p})\,\mathbf{u}_k \tag{8.24}$$

with $\mathbf{A}(\mathbf{p})$ the state matrix which depends on the parameter $\mathbf{p}$, and $\mathbf{B}(\mathbf{p})$ the input matrix which also depends on the parameter $\mathbf{p}$. Note that we also assume that the uncertainty is only present in the forward path. This means that a state which depends on parameter a is never multiplied with parameter a. In other words, the uncertainty is not allowed to accumulate.

Since we are dealing with stochastic variables, the objective function is an expected value function. An example of an objective function is:

$$V(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \mathrm{E}\left[\sum_{k=1}^{N} \mathbf{x}_k^T \mathbf{x}_k\right] \tag{8.25}$$

with N the prediction horizon, and $\mathbf{x}_k$ a stochastic variable. More complex objective functions are possible, this example is only chosen for the ease of notation.

8.2.3 Uncertain parameter in B

Let us start with a simple case where only $\mathbf{B}$ depends on the model parameter $\mathbf{p}$. This means that the uncertain model parameters are localized in the input matrix $\mathbf{B}$. This case is also considered by Couchman et al. (2005). Eqn (8.24) then becomes:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}(\mathbf{p})\,\mathbf{u}_k \tag{8.26}$$

Since the state equations are linear in the model parameters, the matrix $\mathbf{B}$ can be rewritten as follows:

$$\mathbf{B}(\mathbf{p}) = \mathbf{B}_0 + \sum_{i=1}^{r} \mathbf{B}_i K_i \tag{8.27}$$

with K_i the elements contained in $\mathbf{p}$ and r the number of uncertain model parameters in $\mathbf{B}$. $\mathbf{B}$ can then be split in a deterministic part and a stochastic part:

$$\mathbf{B}(\mathbf{p}) = \bar{\mathbf{B}}(\mathbf{p}) + \tilde{\mathbf{B}}(\mathbf{p}) \tag{8.28}$$

with $\bar{\mathbf{B}}$ the expected value of $\mathbf{B}$ and $\tilde{\mathbf{B}}$ the stochastic part of $\mathbf{B}$:

$$\bar{\mathbf{B}}(\mathbf{p}) = \mathbf{B}_0 + \sum_{i=1}^{r} \mathbf{B}_i \bar{K}_i \tag{8.29}$$

$$\tilde{\mathbf{B}}(\mathbf{p}) = \sum_{i=1}^{r} \mathbf{B}_i \tilde{K}_i \tag{8.30}$$

with

$$K_i = \bar{K}_i + \tilde{K}_i \qquad \forall i = 1, \ldots, r \tag{8.31}$$

Eqn (8.26) can then be rewritten as follows:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \left(\bar{\mathbf{B}}(\mathbf{p}) + \tilde{\mathbf{B}}(\mathbf{p})\right)\mathbf{u}_k \tag{8.32}$$

Let us now also split the state $\mathbf{x}_k$ in a deterministic and a stochastic part:

$$\mathbf{x}_k = \bar{\mathbf{x}}_k + \tilde{\mathbf{x}}_k \tag{8.33}$$

with $\bar{\mathbf{x}}_k$ the expected value of $\mathbf{x}_k$, and $\tilde{\mathbf{x}}_k$ the stochastic part of $\mathbf{x}_k$. Eqn (8.32) can then also be separated in an equation for the deterministic and the stochastic part of the state $\mathbf{x}_k$:

$$\bar{\mathbf{x}}_{k+1} = \mathbf{A}\bar{\mathbf{x}}_k + \bar{\mathbf{B}}(\mathbf{p})\mathbf{u}_k \tag{8.34}$$
$$\tilde{\mathbf{x}}_{k+1} = \mathbf{A}\tilde{\mathbf{x}}_k + \tilde{\mathbf{B}}(\mathbf{p})\mathbf{u}_k \tag{8.35}$$

with

$$\bar{\mathbf{x}}_0 = \mathbf{x}_{start} \tag{8.36}$$
$$\tilde{\mathbf{x}}_0 = 0 \tag{8.37}$$

with $\mathbf{x}_{start}$ the state of the system at the start of the horizon. The objective function of Eqn (8.25) can now be further elaborated as follows:

$$\begin{aligned} V(\mathbf{x}_1, \ldots, \mathbf{x}_N) &= \mathrm{E}\left[\sum_{k=1}^{N} \left(\bar{\mathbf{x}}_k + \tilde{\mathbf{x}}_k\right)^T \left(\bar{\mathbf{x}}_k + \tilde{\mathbf{x}}_k\right)\right] \\ &= \sum_{k=1}^{N} \bar{\mathbf{x}}_k^T \bar{\mathbf{x}}_k + \mathrm{E}\left[\sum_{k=1}^{N} \tilde{\mathbf{x}}_k^T \tilde{\mathbf{x}}_k\right] + 2\mathrm{E}\left[\sum_{k=1}^{N} \bar{\mathbf{x}}_k^T \tilde{\mathbf{x}}_k\right] \end{aligned} \tag{8.38}$$

Let us now try to get rid of the expected value operator in this objective function. First we calculate the expected value of the stochastic part of the state $\tilde{\mathbf{x}}_k$. Taking into account Eqn (8.23), Eqn (8.30), Eqn (8.31) and Eqn (8.35):

$$\begin{aligned} \mathrm{E}[\tilde{\mathbf{x}}_{k+1}] &= \mathbf{A}\mathrm{E}[\tilde{\mathbf{x}}_k] + \mathrm{E}\left[\tilde{\mathbf{B}}(\mathbf{p})\right]\mathbf{u}_k \\ &= \mathbf{A}\mathrm{E}[\tilde{\mathbf{x}}_k] + \sum_{i=1}^{r} \mathbf{B}_i \mathrm{E}\left[\tilde{K}_i\right]\mathbf{u}_k \\ &= \mathbf{A}\mathrm{E}[\tilde{\mathbf{x}}_k] + 0 \\ &\Updownarrow \\ \mathrm{E}[\tilde{\mathbf{x}}_k] &= 0 \end{aligned} \tag{8.39}$$

The last term of Eqn (8.38) then becomes:

$$\begin{aligned} \mathrm{E}\left[\sum_{k=1}^{N} \bar{\mathbf{x}}_k^T \tilde{\mathbf{x}}_k\right] &= \sum_{k=1}^{N} \bar{\mathbf{x}}_k^T \mathrm{E}[\tilde{\mathbf{x}}_k] \\ &= 0 \end{aligned} \tag{8.40}$$

The state can be expressed in terms of the input $\mathbf{u}_k$ and the initial state. For Eqn (8.35), taking into account Eqn (8.37), this yields:

$$\begin{aligned}\tilde{\mathbf{x}}_k &= \mathbf{A}^k\tilde{\mathbf{x}}_0 + \sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\tilde{\mathbf{B}}\left(\mathbf{p}\right)\mathbf{u}_i \\ &= \sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\tilde{\mathbf{B}}\left(\mathbf{p}\right)\mathbf{u}_i \end{aligned} \tag{8.41}$$

$\tilde{\mathbf{B}}\left(\mathbf{p}\right)$ can be substituted in Eqn (8.41) using Eqn (8.30):

$$\begin{aligned}\tilde{\mathbf{x}}_k &= \sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\left(\sum_{j=1}^{r}\mathbf{B}_j\tilde{K}_j\right)\mathbf{u}_i \\ &= \sum_{j=1}^{r}\tilde{K}_j\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_j\mathbf{u}_i\right) \end{aligned} \tag{8.42}$$

This can be plugged into the second term of Eqn (8.38) as follows:

$$\begin{aligned}\mathrm{E}\left[\sum_{k=1}^{N}\tilde{\mathbf{x}}_k^T\tilde{\mathbf{x}}_k\right] &= \sum_{k=1}^{N}\mathrm{E}\left[\left(\sum_{j=1}^{r}\tilde{K}_j\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_j\mathbf{u}_i\right)\right)^T\right. \\ &\qquad \left.\left(\sum_{j=1}^{r}\tilde{K}_j\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_j\mathbf{u}_i\right)\right)\right] \\ &= \sum_{k=1}^{N}\mathrm{E}\left[\sum_{j=1}^{r}\sum_{m=1}^{r}\tilde{K}_j\tilde{K}_m\right. \\ &\qquad \left.\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_j\mathbf{u}_i\right)^T\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_m\mathbf{u}_i\right)\right] \\ &= \sum_{j=1}^{r}\sum_{m=1}^{r}\mathrm{E}\left[\tilde{K}_j\tilde{K}_m\right] \\ &\qquad \sum_{k=1}^{N}\left(\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_j\mathbf{u}_i\right)^T\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_m\mathbf{u}_i\right)\right) \end{aligned} \tag{8.43}$$

The objective function of Eqn (8.38) now becomes, taking into account Eqn (8.40)

and Eqn (8.44):

$$\begin{aligned} V\left(\mathbf{x}_1,\ldots,\mathbf{x}_N\right) &= \sum_{k=1}^{N}\bar{\mathbf{x}}_k^T\bar{\mathbf{x}}_k \\ &\quad +\sum_{j=1}^{r}\sum_{m=1}^{r}\mathrm{E}\left[\tilde{K}_j\tilde{K}_m\right] \\ &\quad \sum_{k=1}^{N}\left(\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_j\mathbf{u}_i\right)^T\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_m\mathbf{u}_i\right)\right) \end{aligned} \tag{8.44}$$

Thus the expected value of the objective function reduces to the variance and the covariance of the model parameters. This means that we can get rid of the expected value operator by substituting the variance and covariance of the model parameters. As stated by Eqn (8.23), the covariance is zero, thus Eqn (8.45) becomes:

$$\begin{aligned} V\left(\mathbf{x}_1,\ldots,\mathbf{x}_N\right) &= \sum_{k=1}^{N}\bar{\mathbf{x}}_k^T\bar{\mathbf{x}}_k \\ &\quad +\sum_{j=1}^{r}\sigma_{K_j}^2\sum_{k=1}^{N}\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_j\mathbf{u}_i\right)^T\left(\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}_j\mathbf{u}_i\right) \end{aligned} \tag{8.45}$$

This implies that in order to implement the stochastic objective function of Eqn (8.25), an estimator of the standard deviation of each of the model parameters is required. If there is only one uncertain parameter K_1 in $\mathbf{B}\left(\mathbf{p}\right)$, the objective function of Eqn (8.25) can be written as follows:

$$V\left(\mathbf{x}_1,\ldots,\mathbf{x}_N\right) = \sum_{k=1}^{N}\bar{\mathbf{x}}_k^T\bar{\mathbf{x}}_k+\sum_{k=1}^{N}\boldsymbol{\epsilon}_{x,k}^T\boldsymbol{\epsilon}_{x,k} \tag{8.46}$$

with $\boldsymbol{\epsilon}_{x,k}$ the estimated state uncertainty of $\mathbf{x}_k$, which is defined as:

$$\boldsymbol{\epsilon}_{x,k+1} = \mathbf{A}\boldsymbol{\epsilon}_{x,k}+\hat{\boldsymbol{\sigma}}_{\mathbf{B}}\mathbf{u}_k \tag{8.47}$$

with

$$\begin{aligned} \boldsymbol{\epsilon}_{x,0} &= 0 \\ \hat{\boldsymbol{\sigma}}_{\mathbf{B}} &= \mathbf{B}_1\hat{\sigma}_{K_1} \end{aligned} \tag{8.48}$$

with $\hat{\sigma}_{K_1}$ the estimated standard deviation of the model parameter K_1. Note that $\boldsymbol{\epsilon}_{x,k}$ may be negative. This is however not a problem, since the objective function only uses the square of this variable. An estimator of the standard deviation of the state is:

$$\hat{\boldsymbol{\sigma}}_{x,k} = \left|\boldsymbol{\epsilon}_{x,k}\right| \tag{8.49}$$

The state $\mathbf{x}_k$ thus is normally distributed as follows:

$$\mathbf{x}_k \propto \mathcal{N}\left(\bar{\mathbf{x}}_k,\hat{\boldsymbol{\sigma}}_{x,k}^2\right) \tag{8.50}$$

Because of the assumption that model uncertainty is created by input changes, the standard deviation of the state $\mathbf{x}$ depends on the system inputs. In other words, the variation $\hat{\boldsymbol{\sigma}}_{x,k}^2$ is time-varying, and even varies across the prediction horizon.

8.2.4 Uncertain parameters in $\mathbf{A}$ and $\mathbf{B}$

It is not straightforward to extend the above presented approach for uncertain parameters in $\mathbf{B}$, to the case where there are uncertain parameters in both $\mathbf{A}$ and $\mathbf{B}$. In this case the output in general becomes a nonlinear function of the parameters, which severely complicates the statistics.

Taking into account the assumption that the uncertain parameters are only situated in a so-called forward path, a pragmatic solution is possible. According to Eqn (8.45) the uncertainty caused by a parameter in $\mathbf{B}$ is linearly dependent on the standard deviation of that parameter. If that uncertainty is multiplied with an uncertain parameter in $\mathbf{A}$, this can be treated as if the parameters were multiplied at the source.

Let us illustrate this with an example. Assume an uncertain parameter K_1 in the input matrix $\mathbf{B}$. This parameter induces a state uncertainty on for instance state x_1. Assume a state equation which contains a multiplication of that state x_1 and a second uncertain parameter K_2. If the parameter were not uncertain, the state uncertainty would simply scale with the value of K_2. If the parameter were uncertain however, we could rescale the state uncertainty of x_1 as if it was caused by the product of both uncertain parameters K_1 and K_2, instead of just by one uncertain parameter K_1. Assuming that:

$$K_1 \propto \mathcal{N}\left(\bar{K}_1, \sigma^2_{K_1}\right) \tag{8.51}$$

$$K_2 \propto \mathcal{N}\left(\bar{K}_2, \sigma^2_{K_2}\right) \tag{8.52}$$

the state uncertainty needs to be rescaled with the following factor $\hat{\sigma}_{K_1,K_2}$:

$$\hat{\sigma}_{K_1,K_2} = \frac{\sqrt{\left(\hat{K}_1\right)^2 \left(\hat{\sigma}_{K_2}\right)^2 + \left(\hat{K}_2\right)^2 \left(\hat{\sigma}_{K_1}\right)^2 + \left(\hat{\sigma}_{K_1}\right)^2 \left(\hat{\sigma}_{K_2}\right)^2}}{\hat{\sigma}_{K_1}} \tag{8.53}$$

The state uncertainty is first divided by the estimated standard deviation of K_1, and then multiplied with the standard deviation of $K_1 \times K_2$. Note that the product of two normally distributed variables is not normally distributed. In this work we will assume that the product is also normally distributed. This approximation is only relevant for the calculation of the probability that constraints are violated (see section 8.2.5). The calculation of $\hat{\sigma}_{K_1,K_2}$ is detailed in section E.1 of Appendix E.

This approach can easily be extended for multiple uncertain parameters in $\mathbf{A}$, as long as the uncertain parameters are only present in the forward path.

8.2.5 Probabilistic constraints

For any real-life application, constraint handling is an important aspect. As explained in section 2.2.4 of Chapter 2, the optimization problem consists of two steps:

Phase 1: slack variables are added to the right hand side of all state constraints, and the input vector is determined in such a way that the slack variables are minimized.

Phase 2: the values of the slack variables which are calculated in Phase 1 are added to the right hand side of the constraints. Then the search for the optimal input vector begins, starting from the input vector resulting from the previous phase.

This approach ensures that the problem remains feasible. However, if the model is uncertain, oscillations around the constraint may occur. After all, the process may reach the constraints sooner than expected.

Assume that there is a constraint on the state variable x:

$$x \quad \leq \quad x_{max} \tag{8.54}$$

with x_{max} the upper limit of the variable x. This constraint can now be recast, taking into account the stochastic nature of the variable x:

$$x + K_{\sigma,x}\sigma_x \quad \leq \quad x_{max} \tag{8.55}$$

with $K_{\sigma,x}$ a weighing factor. $K_{\sigma,x} = 2$ for instance means that the constraint holds in 98% of all cases. This is again very analogous to the approach by Couchman in Eqn (8.20).

8.3 Analysis

The objective function of Stochastic MPC is shown in Eqn (8.45):

$$\begin{aligned} V(\mathbf{x}_1, \ldots, \mathbf{x}_N) \quad = \quad & \sum_{k=1}^{N} \bar{\mathbf{x}}_k^T \bar{\mathbf{x}}_k \\ & + \sum_{j=1}^{r} \sigma_{K_j}^2 \sum_{k=1}^{N} \left(\sum_{i=0}^{k-1} \mathbf{A}^{k-i-1} \mathbf{B}_j \mathbf{u}_i \right)^T \left(\sum_{i=0}^{k-1} \mathbf{A}^{k-i-1} \mathbf{B}_j \mathbf{u}_i \right) \end{aligned} \tag{8.56}$$

This objective function contains two terms, which can be interpreted as a bias and a variance term. The first term, the bias term, is the control error. The second term, the variance, is the state uncertainty. Note that, because of our assumption that state uncertainty is only caused by input changes, this second term depends on the parameter uncertainty and the input vector. This term can thus also be seen as a regularization term. The larger the parameter uncertainty becomes, the larger this regularization term grows, slowing down the controller response.

Stochastic MPC can thus be seen as an automatic tuning technique. The most important tuning parameter of MPC, the trade-off between control error and input effort, is modified online in function of the instantaneous model uncertainty.

One might introduce a gain a to vary the relative importance of the uncertainty in Eqn (8.56) compared to the the mean cost. The objective function of Eqn (8.18) in the approach by Couchman shows some similarity with Eqn (8.56). However, the square of Eqn (8.18) is *not* equal to Eqn (8.56). In fact the approach by Couchman gives rise to a SOCP, while this approach leads to an easier to solve QP.

The above analysis can be illustrated with an example:

Example 8.1. *Assume the following MPC problem:*

$$\min_{u_0,\ldots,u_{N-1},z_1,\ldots,z_N} V(u_0, \ldots, u_{N-1}, z_1, \ldots, z_N)$$

$$V(u_0, \ldots, u_{N-1}, z_1, \ldots, z_N) = \sum_{k=1}^{N} E[z_k Q z_k + u_{k-1} R u_{k-1}] \quad (8.57)$$

subject to

$$\begin{array}{rcll} x_{k+1} & = & x_k + K_1 u_k & \\ z_{k+1} & = & K_2 x_k, & \forall k = 0, \ldots, N-1 \\ K_1 & \propto & \mathcal{N}\left(\bar{K}_1, \sigma^2_{K_1}\right) & \\ K_2 & \propto & \mathcal{N}\left(\bar{K}_2, \sigma^2_{K_2}\right) & \end{array} \quad (8.58)$$

The above described probabilistic description can then be divided into a model for the expected values $\bar{x}_k$, $\bar{z}_k$:

$$\begin{array}{rcl} \bar{x}_{k+1} & = & \bar{x}_k + \hat{K}_1 u_k \\ \bar{z}_{k+1} & = & \hat{K}_2 \bar{x}_k \end{array} \quad (8.59)$$

and a model for the state uncertainty where $\epsilon_{x,k}$ and $\epsilon_{z,k}$ are the predicted state uncertainties at time step k of the states x and z. The uncertainty model can then be seen as follows:

$$\begin{array}{rcl} \epsilon_{x,k+1} & = & \epsilon_{x,k} + \hat{\sigma}_{K_1} u_k \\ \epsilon_{z,k+1} & = & \hat{\sigma}_{K_1,K_2} \epsilon_{x,k} \end{array} \quad (8.60)$$

with $\epsilon_{x,0} = 0$, $\epsilon_{z,0} = 0$ and $\hat{\sigma}_{K_1,K_2}$ as defined in Eqn (8.53). Let us now further elaborate Eqn (8.57), by substituting $z_k = \bar{z}_k + \tilde{z}_k$ with $\bar{z}_k$ the expected value of the stochastic variable z:

$$\begin{array}{rcll} W(u_k, z_k) & = & E[z_k Q z_k + u_{k-1} R u_{k-1}] & (8.61) \\ & \Updownarrow & & \\ & = & E[(\bar{z}_k + \tilde{z}_k) Q (\bar{z}_k + \tilde{z}_k)] & \\ & & + E[u_{k-1} R u_{k-1}] & (8.62) \\ & \Updownarrow & & \\ & = & \bar{z}_k Q \bar{z}_k + E[\tilde{z}_k Q \tilde{z}_k] + 2[\bar{z}_k Q \tilde{z}_k] & \\ & & + u_{k-1} R u_{k-1} & (8.63) \\ & \Updownarrow & \tilde{z}_k \propto \mathcal{N}\left(0, \sigma^2_{z,k}\right), |\epsilon_{z,k}| \approx \sigma_{z,k} & \\ & = & \bar{z}_k Q \bar{z}_k + \epsilon_{z,k} Q \epsilon_{z,k} + u_{k-1} R u_{k-1} & (8.64) \end{array}$$

Note in Eqn (8.64) that the sign of the predicted value $\epsilon_{z,k}$ (prediction of $\sigma_{z,k}$) is irrelevant, since this variable only occurs squared. Let us now combine Eqn (8.60) with Eqn (8.64) for different values of k (details can be found in Appendix E, section

E.2):

k=1:

$$W(u_0, z_1) = \bar{z}_1 Q \bar{z}_1 + u_0 R u_0 \tag{8.65}$$

k=2:

$$\begin{aligned} W(u_1, z_2) &= \bar{z}_2 Q \bar{z}_2 + \epsilon_{z,2} Q \epsilon_{z,2} + u_1 R u_1 \\ &\Updownarrow \\ &= \bar{z}_2 Q \bar{z}_2 + u_0 \left(\hat{\sigma}_{K_1} \hat{\sigma}_2\right)^2 Q u_0 + u_1 R u_1 \end{aligned} \tag{8.66}$$

k=3:

$$\begin{aligned} W(u_2, z_3) &= \bar{z}_3 Q \bar{z}_3 + \epsilon_{z,3} Q \epsilon_{z,3} + u_2 R u_2 \\ &\Updownarrow \\ &= \bar{z}_3 Q \bar{z}_3 + (u_0 + u_1) \left(\hat{\sigma}_{K_1} \hat{\sigma}_2\right)^2 Q (u_0 + u_1) \\ &\quad + u_2 R u_2 \end{aligned} \tag{8.67}$$

k=4:

$$\begin{aligned} W(u_3, z_4) &= \bar{z}_4 Q \bar{z}_4 + \epsilon_{z,4} Q \epsilon_{z,4} + u_3 R u_3 \\ &\Updownarrow \\ &= \bar{z}_4 Q \bar{z}_4 + (u_0 + u_1 + u_2) \left(\hat{\sigma}_{K_1} \hat{\sigma}_2\right)^2 Q (u_0 + u_1 + u_3) \\ &\quad + u_3 R u_3 \end{aligned} \tag{8.68}$$

□

This example once more illustrates that using Stochastic MPC actually comes down to online modifying the input weights in the objective function, taking into account the model structure and the current parameter uncertainty. In this example only one input and one state are considered in the objective. One can easily show that in the case of a system with multiple inputs, the uncertainty can influence the trade-off between the inputs. The solution will be biased in favor of the input with the most certain process model.

8.4 Parameter uncertainty

The parameter uncertainty is closely related to the model error. There are, however, several different formulations of the parameter uncertainty possible. In this section the approach taken for capacity control on the combine harvester is described. Note that although for this application the parameter uncertainty is constant across the horizon, the uncertainty may also be made time-varying.

On the combine harvester the estimator error is used to calculate the standard deviation on the parameter error. As discussed in Chapter 6, the gain between two process variables (for instance speed and feed rate) is estimated using Fuzzy RLS.

Remember Eqn (6.8), (6.9) and (6.10):

$$R(N) = (1-\gamma)R(N-1) + \gamma x^2(N) \tag{8.69}$$
$$e(N) = y(N) - K_{N-1}x(N) \tag{8.70}$$
$$K(N) = K(N-1) + \gamma e(N)\frac{x(N)}{R(N)} \tag{8.71}$$

where the update rate γ is made dependent on the error. The error can also be written as:

$$y(N) - e(N) = \left(K_{N-1}^C + \tilde{K}_{N-1}\right)x(N) \tag{8.72}$$

with K_{N-1}^C the correct parameter value and $\tilde{K}_{N-1}$ the parameter error. K_{N-1}^C is defined as:

$$K_{N-1}^C = \frac{y(N)}{x(N)} \tag{8.73}$$

which implies that:

$$-e(N) = \tilde{K}_{N-1}x(N) \tag{8.74}$$

The parameter error can thus be written as:

$$\tilde{K}_{N-1} = \frac{-e(N)}{x(N)} \tag{8.75}$$

The standard deviation of the parameter can then be approximated by $\hat{\sigma}_K$ as follows:

$$\hat{\sigma}_K(N) = \sqrt{(1-\nu)\left(\hat{\sigma}_K(N-1)\right)^2 + \nu\left(\frac{e(N)}{x(N)}\right)^2} \tag{8.76}$$

with ν the update rate ($0 \leq \nu \leq 1$) and $e(N)$ as defined in Eqn (8.70).

The downside of this formulation is that it does not take into account the assumption that parameter errors originate from input changes. The same model error will lead to a smaller parameter error at larger input levels, even though the input change may be the same. The procedure to estimate the parameter uncertainty proposed in Chapter 11 deals with this issue.

Reconfiguration The uncertainty can also be used to switch between models. This switch can be discrete (when a given condition is satisfied a different model is selected) or continuous (a weighted average of the models is taken). Grain mass flow can for instance be modeled based on feed rate or based on engine load. Which model has the best performance will depend on the field and crop circumstances. In the combine harvester application, a continuous approach is used. The model parameters are weighted with the parameter uncertainty.

Example 8.2. *The grain mass flow model consists of a static model, followed by a dynamic model (see Chapter 6). In this example we will only focus on the steady state*

(static) value of Grain Mass Flow (GMF_{ss}). This steady state value depends on the engine load (EL) and the feed rate (FR) as follows:

$$GMF_{ss} = \gamma_1 \hat{K}_{EL} EL + \gamma_2 \hat{K}_{FR} FR \tag{8.77}$$

with γ_1 and γ_2 defined as:

$$\gamma_1 = \frac{\frac{1}{\hat{\sigma}_{K_{EL}}}}{\frac{1}{\hat{\sigma}_{K_{EL}}} + \frac{1}{\hat{\sigma}_{K_{FR}}}} \tag{8.78}$$

$$\gamma_2 = \frac{\frac{1}{\hat{\sigma}_{K_{FR}}}}{\frac{1}{\hat{\sigma}_{K_{EL}}} + \frac{1}{\hat{\sigma}_{K_{FR}}}} \tag{8.79}$$

with $\hat{\sigma}_{K_{EL}}$, $\hat{\sigma}_{K_{FR}}$ the parameter error estimated as calculated in Eqn (8.76). □

8.5 Simulation results

The main benefits of Stochastic MPC, namely robustness to varying conditions, only become clear across different fields and conditions. Since the field condition is not controllable, the effect of the parameter uncertainty is first illustrated on a simulation case.

Stochastic objective

First of all, let us examine the effect of the stochastic objective function. For this purpose the following simple problem is used:

Example 8.3. *Assume a system with the following state space equation:*

$$x_{k+1} = x_k + K u_k \tag{8.80}$$

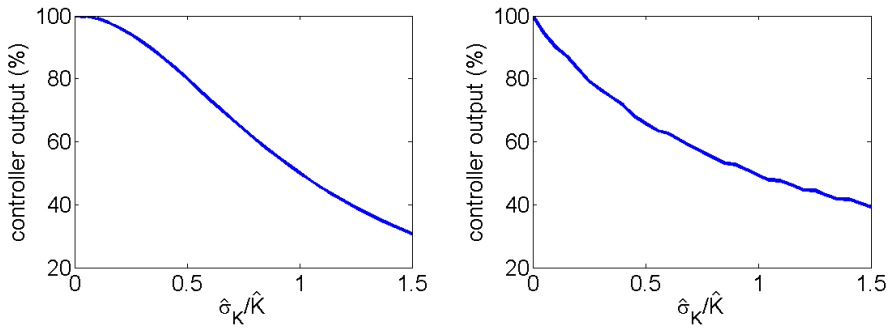
where K is a stochastic variable with standard deviation σ_K and expected value $\bar{K}$. The online estimates of the expected value and the standard deviation are $\hat{K}$ and $\hat{\sigma}_K$, respectively. The above described system can then be divided into a model for the expected value $\bar{x}_k$ and a model for the state uncertainty $\epsilon_{x,k}$. The objective is to control the state x_k to a given set point r_k. This leads to the following optimization problem:

$$\min_{u_k} \sum_{k=1}^{N} \left[(x_k - r_k)\, Q\, (x_k - r_k) + u_{k-1} R u_{k-1} \right] \tag{8.81}$$

subject to

$$\begin{aligned} x_k &= \bar{x}_k + \epsilon x, k \\ \bar{x}_{k+1} &= \bar{x}_k + \hat{K} u_k \\ \epsilon_{x,k+1} &= \epsilon_{x,k} + \hat{\sigma}_K u_k, \qquad \forall k = 0, \ldots, N-1 \end{aligned} \tag{8.82}$$

□



(a) (b)

Figure 8.1: Illustration of the controller output (relative to the output of classic MPC, %) as a function of the relative parameter uncertainty $\hat{\sigma}_K/\hat{K}$, with $\hat{\sigma}_K$ the estimated parameter uncertainty, and $\hat{K}$ the estimated parameter.
(a) when there are no constraints active.
(b) when the controller bumps into a constraint in this iteration (objective is to maximize the controlled variable).

Assume the variable x_1 is considerably smaller than the set point r_1. The controller output of a Stochastic MPC will depend on the parameter uncertainty $\hat{\sigma}_K$. In Fig. 8.1 (a) the controller output (relative to the classic MPC output, as a percentage) is shown as a function of the ratio of the parameter uncertainty $\hat{\sigma}_K$ and the parameter $\hat{K}$. Note that if the parameter uncertainty $\hat{\sigma}_K$ is half the size of the parameter $\hat{K}$, the controller output is decreased by about 20%.

Probabilistic constraints

To illustrate the effect of probabilistic constraints, the controller definition of Example 8.3 needs to be slightly modified:

Example 8.4. *Assume a system with the following state space equation:*

$$x_{k+1} \quad = \quad x_k + K u_k \tag{8.83}$$

where K is a stochastic variable with standard deviation σ_K and expected value $\bar{K}$. The online estimates of the expected value and the standard deviation are $\hat{K}$ and $\hat{\sigma}_K$, respectively. The above described system can then be divided into a model for the expected value $\bar{x}_k$ and a model for the state uncertainty $\epsilon_{x,k}$. The objective is to maximize the state x_k, while respecting a constraint on x_k. This leads to the following optimization problem:

$$\min_{u_k} \sum_{k=1}^{N} [-x_k Q x_k + u_{k-1} R u_{k-1}] \tag{8.84}$$

subject to

$$\begin{aligned} x_k &= \bar{x}_k + \epsilon x, k \\ \bar{x}_{k+1} &= \bar{x}_k + \hat{K} u_k \\ \epsilon_{x,k+1} &= \epsilon_{x,k} + \hat{\sigma}_K u_k \\ \bar{x}_{k+1} + \epsilon x, k+1 &\leq x_{max}, \qquad \forall k = 0, \ldots, N-1 \end{aligned} \tag{8.85}$$

□

This means that the state x_k needs to be maximized, respecting the constraint on the state x_k. Assume x_k is slightly below the constraint x_{max}, such that the constraint will be reached in this iteration. Fig. 8.1 (b) illustrates the effect of the parameter uncertainty on the controller output. In this case, a parameter uncertainty $\hat{\sigma}_K$ which is half the size of the parameter $\hat{K}$, decreases the controller output with more than 30%.

8.6 Field test results

The Stochastic MPC methodology has been tested on the combine harvester, and has been used to implement a capacity control system. This application was already used in Chapter 7. The model of the system is extensively described in Chapter 6. The objective is to control a throughput variable (feed rate, engine load or grain mass flow), respecting the constraints on for instance engine load, feed rate and/or ground speed. Respecting constraints was only possible with the classic MPC of Chapter 7 after detuning the controller (i.e. making the controller respond very conservatively).

As explained earlier, for Stochastic MPC an uncertainty model is required. Furthermore, as explained in section 8.2, the objective function needs to be expanded to a stochastic objective function. Constraints are converted into probabilistic constraints (section 8.2.5).

Uncertainty model

This application is a SIMO system (single input, multiple output), where there are also uncertain parameters in $\mathbf{A}$. This means that the uncertain parameters are multiplied together, as described in section 8.2.4. The uncertainty model is based on the process model of Eqn (6.17), and can be written as follows:

$$\begin{aligned} GS^0_{unc,k+1} &= GS^0_{unc,k} + K_{unc,GS} \Delta u_k \\ &\vdots \\ GS^{d_1}_{unc,k+1} &= GS^{d_1-1}_{unc,k} \\ FR^0_{unc,k+1} &= FR^0_{unc,k} + K_{unc,FR} \left(GS^{d_{FR}}_{unc,k} - GS^{d_{FR}+1}_{unc,k} \right) \\ &\vdots \\ FR^{d_2}_{unc,k+1} &= FR^{d_2-1}_{unc,k} \end{aligned} \tag{8.86}$$

$$
\begin{aligned}
EL^{T}_{unc,k+1} &= (1-\alpha_{EL})\left(EL^{T}_{unc,k} + K_{unc,EL}\left(GS^{d_{EL}}_{unc,k} - GS^{d_{EL}+1}_{unc,k}\right)\right)\\
EL^{0}_{unc,k+1} &= EL^{0}_{k} + \alpha_{EL}\left[K_{unc,EL}\left(GS^{d_{EL}}_{unc,k} - GS^{d_{EL+1}}_{unc,k}\right) + EL^{T}_{unc,k}\right]\\
&\vdots\\
EL^{d_3}_{unc,k+1} &= EL^{d_3-1}_{unc,k}\\
GMF^{T,FR}_{unc,k+1} &= (1-\alpha_{FR-GMF})\left[GMF^{T,FR}_{unc,k}\right.\\
&\quad \left. + K_{unc,FR-GMF}\left(FR^{d_{FR-GMF}}_{unc,k} - FR^{d_{FR-GMF}+1}_{unc,k}\right)\right]\\
GMF^{T,EL}_{unc,k+1} &= (1-\alpha_{EL-GMF})\left[GMF^{T,EL}_{unc,k}\right.\\
&\quad \left. + K_{unc,EL-GMF}\left(EL^{d_{EL-GMF}}_{unc,k} - EL^{d_{EL-GMF}+1}_{unc,k}\right)\right]\\
GMF_{unc,k+1} &= GMF_{unc,k}\\
&\quad + \gamma_1\alpha_{FR-GMF}\left[GMF^{T,FR}_{unc,k}\right.\\
&\quad \left. + K_{unc,FR-GMF}\left(FR^{d_{FR-GMF}}_{unc,k} - FR^{d_{FR-GMF}+1}_{unc,k}\right)\right]\\
&\quad + \gamma_2\alpha_{EL-GMF}\left[GMF^{T,EL}_{unc,k}\right.\\
&\quad \left. + K_{unc,EL-GMF}\left(EL^{d_{EL-GMF}}_{unc,k} - EL^{d_{EL-GMF}+1}_{unc,k}\right)\right]
\end{aligned}
$$

with d_1, d_2, d_3, γ_1 and γ_2, as specified in Eqn (6.18):

$$
\begin{aligned}
d_1 &= \max\left(d_{EL}, d_{FR}\right) + 1\\
d_2 &= d_{FR-GMF} + 1\\
d_3 &= d_{EL-GMF} + 1\\
1 &= \gamma_1 + \gamma_2
\end{aligned}
\tag{8.87}
$$

where the $_{unc}$ subscript denotes the state uncertainty or an uncertainty parameter. $GS_{unc,k}$ for instance denotes the uncertainty on the state GS_k on time step k. $K_{unc,GS}$ denotes the uncertainty parameter of the gain parameter K_{GS}. Define the function f based on Eqn (8.53):

$$
f\left(K_1, \sigma_1, K_2, \sigma_2\right) = \frac{\sqrt{K_1^2\sigma_2^2 + K_2^2\sigma_1^2 + \sigma_1^2\sigma_2^2}}{\sigma_1} \tag{8.88}
$$

The uncertainty parameters can then be calculated as:

$$
\begin{aligned}
K_{unc,GS} &= \hat{\sigma}_{K_{GS}}\\
K_{unc,FR} &= f\left(\hat{K}_{GS}, \hat{\sigma}_{K_{GS}}, \hat{K}_{FR}, \hat{\sigma}_{K_{FR}}\right)\\
K_{unc,EL} &= f\left(\hat{K}_{GS}, \hat{\sigma}_{K_{GS}}, \hat{K}_{EL}, \hat{\sigma}_{K_{EL}}\right)\\
K_{unc,FR-GMF} &= f\left(\hat{K}_{GS}\cdot\hat{K}_{FR}, K_{unc,FR}, \hat{K}_{FR-GMF}, \hat{\sigma}_{K_{FR-GMF}}\right)\\
K_{unc,EL-GMF} &= f\left(\hat{K}_{GS}\cdot\hat{K}_{EL}, K_{unc,EL}, \hat{K}_{EL-GMF}, \hat{\sigma}_{K_{EL-GMF}}\right)
\end{aligned}
\tag{8.89}
$$

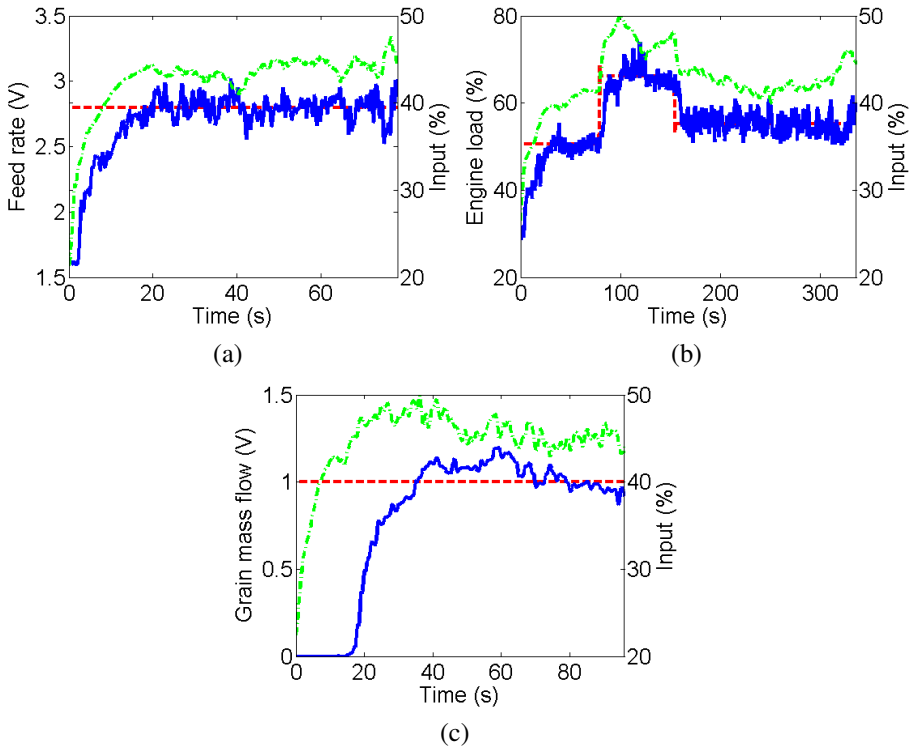
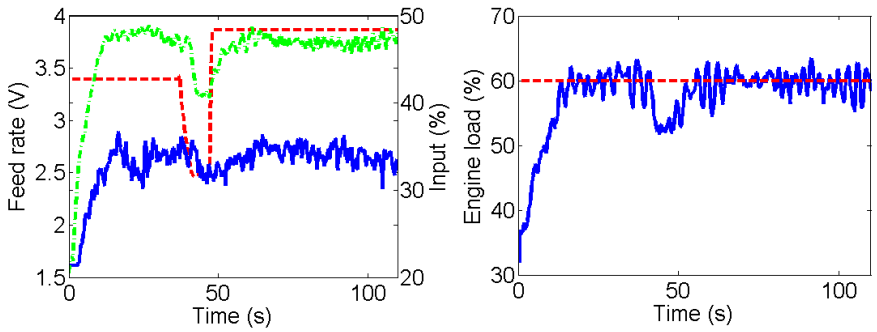


Figure 8.2: The step response for (a) feed rate control (F, 20061107c), (b) engine load control (F, 20060905a) and (c) grain mass flow control (F, 20061109b). The solid line is respectively the feed rate, engine load and grain mass flow, the dashed line is the respective set point (both on the left axis) and the dash-dot line is the input (on the right axis).

Stochastic Objective

Feed rate control The objective here is to maintain the feed rate measurement as close as possible to the set point. During the field test (F, 20061107c), no constraints are active. The controller performance is very similar to the performance of the classic MPC controller of Chapter 7 (Fig. 8.2 (a)). The response shown is a crop entry in corn. The combine is first put in front of the crop, and then the controller is engaged. Since the threshing and feeder mechanism are engaged in advance, the feed rate signal is already at the zero load level (about 1.6V) before the machine enters the crop. Note that there is no steady-state error, and that the input is varied smoothly, without any oscillation.

Engine load control The engine load is first controlled towards 50% load, after which the set point is raised to 70%, and lowered back to 50% (F, 20060905a). The controller responds swiftly (Fig. 8.2 (b)). Although the engine load measurement is quite noisy, and also depends on the propulsion system, the controller response is very



(a) (b)

Figure 8.3: (a) The step response of feed rate control with engine load constraint (F, 20061109c). At first the engine load constraint is active, then the set point is lowered which allows the controller to return to normal feed rate control. At the end of the experiment the feed rate set point is raised again and the engine load limitation becomes once more active. The solid line is feed rate, the dashed line is the feed rate set point (both on the left axis) and the dash-dot line is the input (on the right axis). (b) The engine load (solid line) and the engine load constraint (dashed line) for the same experiment.

smooth. Note that there is a small overshoot when the reference is raised from 50 to 70%. The controller however, intervenes quickly, without compromising the operator comfort.

This type of control can be used in situations where engine load is the only limiting factor. This situation for instance occurs in corn on hilly terrain.

Grain mass flow control Due to the time-dependent nature of the grain mass flow model (very uncertain parameters), it is very hard to control grain mass flow with classic MPC. The step response with Stochastic MPC is shown in Fig. 8.2 (c) (F, 20061109b). Note that even though there is a large delay between ground speed and grain mass flow, the controller can still be tuned quite aggressively, resulting in a swift step response. The control input is a bit noisy in this case, but is still acceptable, since the operator does not sense it.

It is very hard to estimate the parameter uncertainty for this relation because of the large time delay and the slow dynamics. It takes 15 to 20 seconds before the grain mass flow enters regime conditions, whereas the inputs of the grain mass flow models, namely engine load and feed rate respond almost immediately. In this research no special efforts were made to improve the parameter uncertainty estimate for grain mass flow. Undoubtedly, an improved uncertainty estimate would considerably improve the controller performance.

Probabilistic constraints

Feed rate control with engine load constraint Experiments were also performed with classic MPC (no probabilistic constraints) for this application. In order to get

the controller stable in the neighborhood of a constraint, a lot of regularization was required, which led to a very slow response. With probabilistic constraints, the controller only becomes more careful in the vicinity of the constraints, not across the entire feasible range. This results in a good performance, without compromising on stability.

Fig. 8.3 shows a step response of the feed rate controller with an engine load constraint (F, 20061109c). At the beginning and at the end of the run the constraint is active. In the middle of the run the set point is lowered, and the controller returns to normal, unconstrained operation. Note that when the set point is increased again, the controller moves very swiftly towards the constraint without oscillating.

8.7 Conclusions

This chapter presents a novel approach to Stochastic MPC, which is illustrated on the capacity control application. For this application a more advanced control methodology was required to be able to respect constraints without having to tune the controller conservatively. Moreover, in order to have the best performance in all circumstances, the objective function should take the model uncertainty into account.

The state uncertainty is modeled based on the process model and the parameter uncertainty. This implies that the state uncertainty is assumed to depend solely on the input change. If the system is in steady-state, the state uncertainty thus is zero. It is also assumed that the state uncertainty is never multiplied by the uncertain parameter that caused it. The parameter uncertainty can thus never be situated in the feedback path. This restriction poses no problem for the applications treated in this research. The state uncertainty is an estimator of the standard deviation of the state.

In Stochastic MPC the objective function is replaced by the expected value of the objective function, and constraints are redefined as probabilistic constraints. Taking the expected value of the objective function implies that an extra term is added to the objective function. Next to quadratic terms in the control error and the input effort (input change), the objective function now also contains a quadratic term in the state uncertainty. Since the state uncertainty is a function of the input change, this state uncertainty term is actually an extra regularization term. Stochastic MPC thus can be seen as an automatic tuning method. The objective function is modified online based on the instantaneous quality of the model.

For the case where the uncertainty parameters only occur in the input matrix, the objective function can be calculated exactly. If the state matrix also contains uncertain parameters, the objective function is approximated.

A probabilistic constraint means that the constraint is satisfied with a given probability. Instead of requiring a state to be smaller than its upper bound, the state increased with γ times the state uncertainty needs to be smaller than the upper bound. Assuming that the parameter error, and thus the state error is normally distributed, a constraint is satisfied with 98% probability if $\gamma = 2$.

Benefits

With Stochastic MPC the tuning effort can be considerably reduced, since the regularization is now varied online as a function of the model uncertainty. The input

effort weight in the objective function thus becomes less important. This also renders the controller more robust to crop variations. Another benefit of Stochastic MPC is that it stabilizes constraints, without compromising on controller performance. The controller can be tuned aggressively and still approach constraints carefully.

Drawbacks

Defining an exact uncertainty model to predict the standard deviation of the states is only possible if the uncertain parameters only occur in the input matrix. In this chapter a pragmatic solution is proposed to deal with uncertainty in the state matrix, but this is only an approximation, and subject to a number of assumptions.

Chapter 9

MPC implementation

640K ought to be enough for anybody
-Bill Gates-



This chapter forms the last pillar to arrive at real-life applications, namely optimization methods. MPC requires the solution of an optimization problem in every time step. In most cases, and also for the applications considered in this research, a quadratic objective function needs to be optimized subject to linear equality and inequality constraints.

The earlier applications in process industry all typically operated at a slow update rate. An update rate of once a minute or even less was no exception. The first applications in other sectors were also often slow processes or simulations, which do not call for a fast calculation. The capacity control system (Chapter 7, Chapter 8) however, calls for a sampling rate of 5 Hz, and cruise control on the combine harvester (Chapter 10) even requires a sampling rate of 20 Hz.

A lot of methodologies exist to solve these optimization problems very quickly. When building a prototype or a commercial implementation with a low volume, not only computational efficiency (number of operations necessary to solve the problem), but also the time required to implement the method is important. After all, often the fastest methods described in literature are not yet available on the implementation platform of choice. For instance, LabView (W, National Instruments (2008)), a leading software platform for real-time prototype implementation, only offers an Active Set Method since mid-2006.

Computational efficiency determines which hardware is needed for the application. The more computing power is needed, the higher the cost. On the other hand, for low volume applications, the engineering cost to implement the method is also important. For a lot of applications, the fastest, most complicated method thus is not the most economical solution.

This chapter first gives some background on Quadratic Programming (QP). Subsequently a variation on the general Active Set Method is derived, and contrasted with the standard implementation. This method is benchmarked with the Interior-Point Method available in LabView on a problem from literature. The derived Active Set Method is extended to solve Linear Programs (LPs) as well. Finally both methods are

compared on an optimization problem which occurs in the cruise control application of Chapter 10. The results of this chapter have been published as Coen et al. (2008a).

9.1 Theory of Quadratic Programming

An optimization problem with a quadratic objective function and linear constraints is called a quadratic program (Nocedal and Wright, 1989). The general quadratic program (QP) can be stated as:

$$\min f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{G}\mathbf{x} + \mathbf{x}^T\mathbf{d} \tag{9.1}$$

subject to

$$\begin{array}{ll} \mathbf{a}_i^T\mathbf{x} = b_i, & i \in \mathcal{E} \\ \mathbf{a}_i^T\mathbf{x} \geq b_i, & i \in \mathcal{I} \end{array} \tag{9.2}$$

where $\mathbf{G}$ is a symmetric $n \times n$ matrix. $\mathbf{d}$, $\mathbf{x}$ and $\{\mathbf{a}_i\}, i \in \mathcal{E} \cup \mathcal{I}$ are vectors with n elements. $\mathcal{I}$ is the set of inequality constraints, and $\mathcal{E}$ is the set of equality constraints. Note that by using the matrix definition of Eqn (2.6), the MPC optimization problem of Eqn (2.4) can be written as Eqn (9.1):

$$\min W(\mathbf{y}_s, \mathbf{x}_s, \mathbf{u}_s) \tag{9.3}$$

subject to

$$\begin{array}{rcl} \mathbf{b} & = & \mathbf{A}_s\mathbf{x}_s + \mathbf{B}\mathbf{u}_s \\ \mathbf{y}_s & = & \mathbf{C}_s\mathbf{x}_s \\ 0 & \geq & \mathbf{D}_s\mathbf{x}_s + \mathbf{F}_s\mathbf{u}_s \end{array}$$

with

$$\begin{array}{rcl} W(\mathbf{y}_s, \mathbf{x}_s, \mathbf{u}_s) & = & \left(\mathbf{y}_s - \mathbf{y}_{s,ref}\right)^T \mathbf{S}_s \left(\mathbf{y}_s - \mathbf{y}_{s,ref}\right) \\ & & + \left(\mathbf{x}_s - \mathbf{x}_{s,ref}\right)^T \mathbf{Q}_s \left(\mathbf{x}_s - \mathbf{x}_{s,ref}\right) \\ & & + \mathbf{u}_s^T\mathbf{R}_s\mathbf{u}_s \end{array} \tag{9.4}$$

where the subscript s denotes that the matrix or vector belongs to this formulation with stacked vectors. The stacked vectors $\mathbf{u}_s$, $\mathbf{x}_s$, $\mathbf{y}_s$, $\mathbf{x}_{s,ref}$ and $\mathbf{y}_{s,ref}$ are described in Eqn (2.5). The optimization problem of Eqn (9.1) can be proven to be equivalent to:

$$\min_{\mathbf{x}} \max_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i (\mathbf{a_i}^T\mathbf{x} - b_i) \tag{9.5}$$

where $\mathcal{L}$ is called the Lagrangian, and λ_i are so-called Lagrangian multipliers. In the following sections some methods for solving equality-constrained and inequality-constrained problems are treated. More details can be found in Nocedal and Wright (1989) and Boyd and Vandenberghe (2004).

9.1.1 Equality-constrained quadratic programs

Denote the number of constraints m, the number of optimization variables n, assume that the constraints are independent, and that $m \leq n$. After all, if the number of constraints is larger than the number of optimization variables, there is no exact solution possible. Based on Eqn (9.1), the quadratic program can be written as:

$$\min_{\mathbf{x}} g(\mathbf{x}) \tag{9.6}$$

subject to

$$\mathbf{Ax} = \mathbf{b} \ , \tag{9.7}$$

with

$$g(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{G}\mathbf{x} + \mathbf{x}^T\mathbf{d}, \tag{9.8}$$

and $\mathbf{A}$ the Jacobian of the constraints defined by

$$\mathbf{A} = [\mathbf{a}_i]^T_{i\in\mathcal{E}} \, , \tag{9.9}$$

where $\mathbf{a}_i$ is the first derivative of the ith constraint to $\mathbf{x}$. Note that since the constraints are independent, $\mathbf{A}$ is of full row rank. Based on Eqn (9.5), the Lagrangian for this problem can be formulated:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = g(\mathbf{x}) - \sum_{i\in\mathcal{E}} \lambda_i(\mathbf{a_i}^T\mathbf{x} - b_i) \tag{9.10}$$

As mentioned before, the solution to the problem of Eqn (9.6) can be found by maximizing Eqn (9.10) in terms of $\boldsymbol{\lambda}$ and minimizing it in terms of $\mathbf{x}$. The first-order optimality conditions for this problem are called the Karush-Kuhn-Tucker (KKT) conditions (Nocedal and Wright, 1989). They state that for a given solution $\mathbf{x}^*$, a vector $\boldsymbol{\lambda}^*$ must exist such that the following system of equations is satisfied:

$$\begin{bmatrix} \mathbf{G} & -\mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}^* \\ \boldsymbol{\lambda}^* \end{bmatrix} = \begin{bmatrix} -\mathbf{d} \\ \mathbf{b} \end{bmatrix} \tag{9.11}$$

where the first equation sets the derivative of Eqn (9.10) to $\mathbf{x}$ equal to zero, and the second equation puts the derivative of Eqn (9.10) to $\boldsymbol{\lambda}$ equal to zero.

The system (9.11) can be rewritten in a form that is more suited for computation by expressing $\mathbf{x}^*$ as $\mathbf{x}^* = \mathbf{x}_k + \mathbf{p}$, where $\mathbf{x}_k$ is an intermediate solution and $\mathbf{p}$ is the desired step. By introducing this notation and rearranging the equations, we obtain:

$$\begin{bmatrix} \mathbf{G} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} -\mathbf{p} \\ \boldsymbol{\lambda}^* \end{bmatrix} = \begin{bmatrix} \mathbf{g} \\ \mathbf{c} \end{bmatrix} \tag{9.12}$$

where

$$\mathbf{c} = \mathbf{A}\mathbf{x}_k - \mathbf{b}, \quad \mathbf{g} = \mathbf{d} + \mathbf{G}\mathbf{x}_k, \quad \mathbf{p} = \mathbf{x}^* - \mathbf{x}_k \tag{9.13}$$

The left matrix in (9.12) is called the KKT matrix. Note that this matrix is always indefinite due to its special structure. This severely complicates solving (9.12). Several

methods to obtain $\mathbf{p}$ and $\boldsymbol{\lambda}^*$ exist, such as the range-space and the null-space method (Nocedal and Wright, 1989). These methods are all based on a decomposition of the KKT matrix. The structure of the matrix can be exploited to reduce the computational complexity. This requires however implementing the correct numerical solvers for the decomposition. These numerical aspects render implementing the decomposition a tedious task.

9.1.2 Inequality-constrained quadratic programs

The methods presented in this section can easily be extended to the general case of a quadratic program with equalities and inequalities. For notational simplicity, only inequalities are taken into account. First an Interior-Point Method is discussed. Then a short introduction to Active Set Methods is given.

Assume the following inequality-constrained problem:

$$\min_{\mathbf{x}} \frac{1}{2}\mathbf{x}^T\mathbf{G}\mathbf{x} + \mathbf{x}^T\mathbf{d} \tag{9.14}$$

subject to

$$\mathbf{A}\mathbf{x} \geq \mathbf{b}\ , \tag{9.15}$$

The Lagrangian for this problem is:

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) &= g(\mathbf{x}) - \sum_{i \in \mathcal{I}} \lambda_i (\mathbf{a_i}^T\mathbf{x} - b_i) \\ \lambda_i &\geq 0 \end{aligned} \tag{9.16}$$

Note that the main difference with the equality-constrained problem is the extra constraint on λ_i.

Interior-Point Method (IPM)

An IPM is basically an iterative method to solve the nonlinear KKT conditions. Based on the Lagrangian formulation of Eqn (9.16), the KKT conditions are:

$$\begin{cases} \nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = 0, & \\ \mathbf{a}_i^T\mathbf{x}^* \geq b_i, & \forall i \in \mathcal{I}, \\ \lambda_i^* \geq 0, & \forall i \in \mathcal{I}, \\ \lambda_i^* \left(\mathbf{a}_i^T\mathbf{x}^* - b_i\right) = 0, & \forall i \in \mathcal{I}. \end{cases} \tag{9.17}$$

The first equation sets the first derivative of the Lagrangian to $\mathbf{x}$ to zero. The second equation is the inequality constraint. The third equation is the positivity constraint on the Lagrangian multiplier. If the left-hand and the right-hand side of an inequality constraint are equal, the constraint is called active. The last equation stipulates that the Lagrangian multiplier can only be different from zero if the inequality constraint is active. By introducing a positive slack variable $\mathbf{y}$, the KKT conditions can be rewritten

as:

$$\begin{array}{rcll} \mathbf{G}\mathbf{x} - \mathbf{A}^{*T}\boldsymbol{\lambda} + \mathbf{d} & = & 0, & \\ \mathbf{A}^{*}\mathbf{x} + \mathbf{y} - \mathbf{b} & = & 0, & \\ y_i\lambda_i & = & 0, & i \in \mathcal{I}, \\ \mathbf{y} & \geq & 0, & \\ \boldsymbol{\lambda} & \geq & 0, & \end{array} \tag{9.18}$$

with

$$\mathbf{A}^{*} = [\mathbf{a_i}]^T_{i\in\mathcal{I}}. \tag{9.19}$$

This problem is solved in an iterative fashion, and starts from a randomly chosen point $(\mathbf{x}^0, \mathbf{y}^0, \boldsymbol{\lambda}^0)$ that satisfies all equations of Eqn (9.18), except one, namely $y_i\lambda_i = 0, i \in \mathcal{I}$. The error on this set of equations is called the duality gap. A duality measure μ can be defined as:

$$\mu \quad = \quad \frac{1}{s}\sum_{i=1}^{s} y_i\lambda_i = \frac{\mathbf{y}^T\boldsymbol{\lambda}}{s} \tag{9.20}$$

with s the number of constraints. The algorithm now calculates the optimal steps $(\Delta\mathbf{x}, \Delta\mathbf{y}, \Delta\boldsymbol{\lambda})$ to reduce the duality μ to zero starting from the starting value $(\mathbf{x}^0, \mathbf{y}^0, \boldsymbol{\lambda}^0)$. By eliminating $\Delta\mathbf{y}$ from Eqn (9.18), the first two equations reduce to:

$$\begin{bmatrix} \mathbf{G} & -\mathbf{A}^T \\ \mathbf{A} & \boldsymbol{\Lambda}^{-1}\mathbf{Y} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} \quad = \quad \begin{bmatrix} -\mathbf{r_d} \\ -\mathbf{r_b} + \left(-\mathbf{y} + \sigma\mu\boldsymbol{\Lambda}^{-1}\mathbf{e}\right) \end{bmatrix} \tag{9.21}$$

where

$$\mathbf{Y} = \text{diag}(y_1, y_2, \ldots, y_m) \quad \boldsymbol{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m) \quad \mathbf{e} = (1, 1, \ldots, 1)^T$$

$$\mathbf{r_d} = \mathbf{G}\mathbf{x} - \mathbf{A}^T\boldsymbol{\lambda} + \mathbf{d} \qquad \mathbf{r_b} = \mathbf{A}\mathbf{x} - \mathbf{y} - \mathbf{b} \tag{9.22}$$

μ is the duality measure, $\sigma \in (0, 1)$ is a tuning parameter which determines the convergence speed of the algorithm. Note that the system (9.21) may not have a solution if the convergence speed is set too high (small σ). The step size α is defined such that $\mathbf{y} > 0$ and $\boldsymbol{\lambda} > 0$ remain satisfied. The next intermediate solution is then defined as:

$$(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\lambda}^*) \quad = \quad (\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}) + \alpha(\Delta\mathbf{x}, \Delta\mathbf{y}, \Delta\boldsymbol{\lambda}) \tag{9.23}$$

where α is the step size and $(\Delta\mathbf{x}, \Delta\mathbf{y}, \Delta\boldsymbol{\lambda})$ is the solution of system (9.21). More details about IPMs can be found in Nocedal and Wright (1989) and in Boyd and Vandenberghe (2004). Note that choosing an initial value is very easy for an IPM. As long as $\mathbf{y} > 0$ and $\boldsymbol{\lambda} > 0$ the starting point is allowed.

Active Set Method (ASM)

The scheme of an ASM is presented in Fig. 9.1. Only a brief outline of the method is given here. More details can be found in section 9.3. First of all, a feasible starting

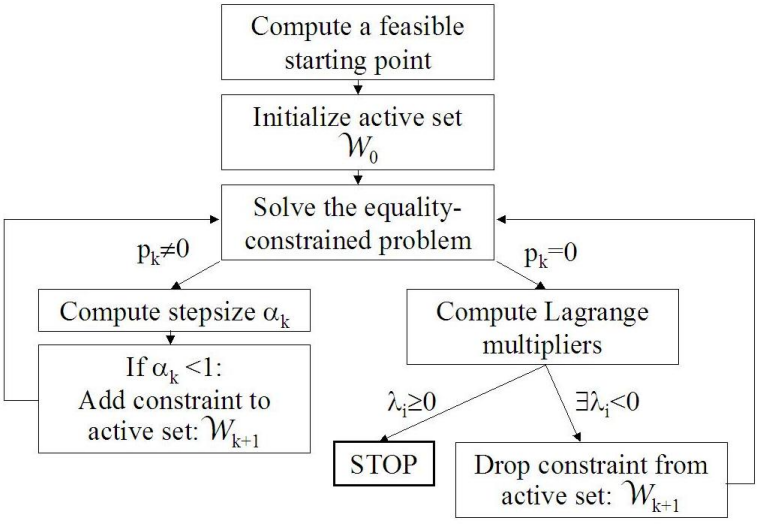


Figure 9.1: The flow chart of an ASM.

point $\mathbf{x}^0$ is calculated. A feasible point is a point that satisfies all the constraints. In every iteration k an intermediate solution $\mathbf{x}_k$ is available. In the first iteration this is the starting point $\mathbf{x}^0$. The active set is defined as the set of constraints that would be violated if the method proceeds from the current solution $\mathbf{x}_k$ to the unconstrained solution $\mathbf{x}^*$. At the start of the method the active set is usually initialized as an empty set.

In order to avoid violating the inequality constraints, an equality-constrained problem is solved instead of the unconstrained problem, with the equations in the active set as equality constraints. The next solution $\mathbf{x}_{k+1}$ is defined as $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$, where $\mathbf{p}_k$ is the optimal step from the current solution calculated by solving the equality-constrained problem. If the step $\mathbf{p}_k$ is different from zero, a step size α_k is calculated such that no constraints are violated. In the case that α_k is smaller than 1, there is a blocking constraint, which is then added to the active set. The blocking constraint is defined as the constraint that would be violated (first) if the step $\mathbf{p}_k$ is taken. The procedure is then repeated with the new active set.

If the step $\mathbf{p}_k$ is zero, the gap between the unconstrained solution $\mathbf{x}^*$ and the current solution $\mathbf{x}_k$ cannot be further decreased without leaving one or more of the active set constraints. The Lagrange multipliers are then used to check whether all constraints are necessary. If the active set contains a constraint that would not be violated by a step towards the unconstrained optimum $\mathbf{x}^*$, this constraint is dropped from the active set. If all constraints are necessary, the current solution is the optimal solution.

Final remarks

It has been shown that IPMs scale better with increasing system sizes than ASMs. The number of iterations of ASMs increases with the system dimension, whereas the num-

ber of iterations of IPMs is known to increase with the root of the system dimension (Boyd and Vandenberghe, 2004). This however does not mean that the calculation time of ASMs scale linearly with the system dimension, since the calculations inside each iteration also scale with the problem size. For small to mid-size systems ASMs are generally faster.

Both methods come down to solving an equality-constrained problem, which requires factorizations that are difficult to implement. However, since the matrix structure (location of zeros) remains constant for IPMs, standard sparse factorization techniques can be used. For ASMs, the structure of the matrix depends on the current active set. This renders an efficient implementation very difficult. In the following an ASM algorithm will be presented that avoids solving an equality-constrained problem.

9.2 Explicit MPC

One of the major drawbacks of MPC is its computational complexity. MPC requires the solution of a QP problem in every time step. In this section an algorithm is examined which allows to reduce the online computational effort.

The solution of a QP problem depends on which constraints are so-called 'active'. An active constraint is a constraint where there is no margin left. This means that an equality constraint will always be active, whereas an inequality constraint will most of the time be inactive. The active set is the collection of active constraints. The solution of a QP problem with a fixed set of active constraints is a linear function of the states. In other words, if one could describe the relation between the process state and the active set, one could map the process states to the right linear state function.

Based on this concept, an explicit formulation of MPC was presented by Bemporad et al. (2002) and Bemporad et al. (2003), which moves the online calculations as much as possible to an offline phase. Multi-parametric Quadratic Programming is used to split up the state space in regions with the same linear solution. The computational effort to calculate these cells will depend on the number of states, the degrees of freedom (input dimension times the control horizon) and the number of constraints.

This new technique is a valid alternative to online optimization for smaller applications, which have to be executed at a relatively fast rate, such as in the automotive or aerospace industry. The decision between online and offline computations is a trade-off between CPU speed and memory. After all, the explicit solution needs to be stored in memory. Explicit MPC divides the state space in a number of cells, and defines a linear solution of the states for each cell. This number of cells will grow superlinear with the problem size.

The downside of this approach is that it requires a lot of memory, and that it thus becomes practically infeasible for larger problems. Moreover, the offline computational effort steeply increases as the problem grows larger. Current research is focused on implementation techniques which do not require the storage of the polyhedral cells (Borelli et al., 2001), and on developing suboptimal methods that can make a trade-off between performance loss and controller complexity (Bemporad and Filippi, 2001). The controller complexity can be assumed proportional to the number of polyhedral cells. At this time the capacity control application is too complex to consider Explicit MPC as an alternative to online optimization.

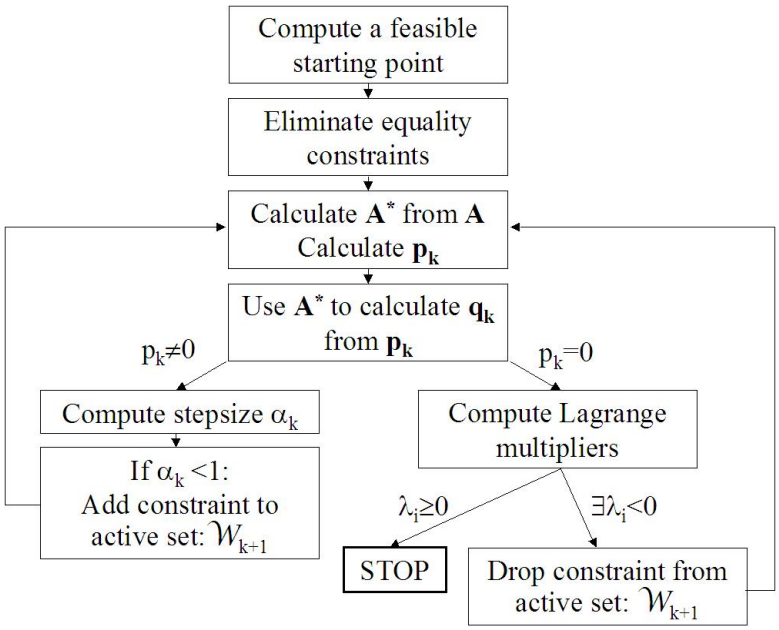


Figure 9.2: The flow chart of the Active Set Method presented in section 9.3.

Reducing the computation time is critical to increase the number of applications to which MPC can be applied. Next to Explicit MPC, other approaches to reduce the online computation time have recently been developed, such as the extended online active set strategy used by Ferreau et al. (2007).

9.3 Implementation of an Active Set Method

9.3.1 Motivation

When solving QPs online, speed is essential. For small to mid-size problems ASMs are often faster than IPMs. ASMs also have a nice geometrical interpretation, which makes them easier to understand. The major disadvantage of implementing an ASM is the complicated sparse decomposition, which is very hard to implement efficiently. If the method needs to be implemented on a dedicated processor, this is a serious drawback. In this section an alternative implementation of ASM is presented, that avoids this implementation problem. The method is a combination of the previously described ASM and the gradient-projection method (Nocedal and Wright, 1989). The following sections describe the algorithm, starting from the general optimization problem defined in Eqn (9.1). An overview of the algorithm is given in Fig. 9.2.

9.3.2 Eliminate equality constraints

First of all the equality constraints from (9.1) are removed. In the case of MPC, the equality constraints are the dynamic model, and give the relation between initial state

and input on one hand, and state and output on the other hand. Using this model, the objective can be written as a function of the initial state and the input. We then have an optimization problem with only inequality constraints. The problem can be written as:

$$\min_{\mathbf{v}} \frac{1}{2}\mathbf{v}^T\mathbf{H}\mathbf{v} + \mathbf{v}^T\mathbf{r} \tag{9.24}$$

subject to

$$\boldsymbol{\theta}_i^T\mathbf{v} \quad \geq \quad b_i, \qquad i \in \mathcal{I} \tag{9.25}$$

Note that the elimination of the constraints comes down to a transformation of the optimization variable. A number of variables contained in $\mathbf{x}$ are eliminated. The result of this elimination is called $\mathbf{v}$. This changes the objective function, where $\mathbf{G}$ and $\mathbf{d}$ become respectively $\mathbf{H}$ and $\mathbf{r}$, as well as the inequality constraints, where vector $\mathbf{a}_i$ becomes $\boldsymbol{\theta}_i$. Eliminating the equality constraints reduces the problem dimension, and thus the computation time, considerably. The dimension of the optimization vector $\mathbf{v}$ only depends on the control horizon and the input dimension. Thus it is computationally interesting to choose the control horizon shorter than the prediction horizon. We assume that a feasible starting point $\mathbf{v}^0$ is available, that satisfies all the constraints. The problem of finding such a starting point is treated in the next section.

In a standard ASM the equality constraints are not eliminated. However, in the case of MPC, the elimination is very straightforward, and the dimension reduction of the optimization vector speeds up the online optimization. Moreover, the elimination of these equality constraints can be performed offline. This elimination has to be repeated each time the model or the objective function change. This means that these calculations have to be performed online, but at a slower rate such as once a minute. Thus the calculations are no longer time-critical, and can occur at lower priority.

9.3.3 Calculate the optimal step

An ASM is also an iterative procedure, just like the IPM. In every iteration a better solution is calculated. This intermediate solution is always feasible, and the value of the objective function always decreases from one iteration to the next. The solution of (9.24) in iteration k can be written as $\mathbf{v}_k = \mathbf{v}_{k-1} + \alpha_k\mathbf{q}_k$ where $\mathbf{v}_{k-1}$ is the result in the previous iteration and $\mathbf{v}_0 = \mathbf{v}^0$. α_k is the step size. The solution of the unconstrained problem is $\mathbf{v}_k^{unconstr} = \mathbf{v}_{k-1} + \mathbf{p}_k$. $\mathbf{p}_k$ is the optimal step for the unconstrained problem, and is calculated by solving (9.24) without constraints.

The algorithm starts with an empty active set. In the next section is explained how this active set is updated. Each member of the active set is a hyperplane, to which the solution $\mathbf{v}_{k+1}$ has to belong. Such a hyperplane is defined by a vector $\boldsymbol{\theta}_i$, perpendicular to all points in the hyperplane. In other words, if the step $\mathbf{p}_k$ has a component along one of these vectors $\boldsymbol{\theta}_i$, this means the solution will move away from the hyperplane. The active set thus forms a basis for the infeasible region. The active set can be written as a matrix $\mathbf{A}^\dagger$, where each row contains an orthonormal basis vector. To avoid moving into the infeasible region, all components of $\mathbf{p}_k$ along any of the vectors contained in the rows of $\mathbf{A}^\dagger$ need to be removed. Assuming that the

basis is orthonormal, the optimal step $\mathbf{q}_k$ can be defined as:

$$\mathbf{q}_k = \mathbf{p}_k - \mathbf{A}^{\dagger^T}\mathbf{A}^{\dagger}\mathbf{p}_k \tag{9.26}$$

where each row of $\mathbf{A}^{\dagger}$ contains a vector of the orthonormal basis defined by the active set. The step size α_k is determined just as for the standard ASM:

$$\begin{gathered} \alpha_k \in [0,1] \\ \alpha_k = \min\left(1, \min_{i \in \mathcal{W}_k, \boldsymbol{\theta}_i^T \mathbf{q}_k < 0} \frac{b_i - \boldsymbol{\theta}_i^T \mathbf{v}_k}{\boldsymbol{\theta}_i^T \mathbf{q}_k}\right) \end{gathered} \tag{9.27}$$

with $\mathcal{W}_k$ the active set.

9.3.4 Transforming the active set

The update of the active set is analogous to the standard ASM. A constraint is added to the active set if the step size α_k is smaller than 1. If the optimal step $\mathbf{q}_k$ is zero, the Lagrangian multipliers are used to check whether all constraints are necessary. If not, the superfluous constraint, indicated by a negative Lagrangian multiplier, is removed. The difference with the standard method is that the active set is transformed into an orthonormal basis. The active set matrix $\mathbf{A}$ (see Eqn (9.9)) is thus transformed into matrix $\mathbf{A}^{\dagger}$ using Gram-Schmidt orthonormalization (Friedberg et al., 1989). Depending on the problem size, the orthonormalization can be performed recursively (Huang et al., 2001) or the basis can be recalculated at each update.

9.3.5 Rotational symmetry

So far it was assumed implicitly that the objective function is rotationally symmetric relative to its minimum. In fact the solver minimizes the distance between the unconstrained solution and the constrained solution. However, this is only correct if the objective is rotationally symmetric.

In Fig. 9.3 is illustrated what happens if the objective function is not rotationally symmetric. The algorithm ends up in point A, which would be the solution if the objective had been rotationally symmetric. Point B is the correct solution. This can be dealt with by performing another transformation of variables. Assume that the objective function is $\mathbf{v}^T\mathbf{H}\mathbf{v}$. It can be transformed into a rotationally symmetric problem by defining a new variable $\mathbf{w} = \sqrt{\mathbf{H}}\mathbf{v}$ such that $\mathbf{v}^T\mathbf{H}\mathbf{v} = \mathbf{w}^T\mathbf{I}\mathbf{w}$ with $\mathbf{I}$ the identity matrix. The transformation performed on the objective function and on the constraints then is $\mathbf{v} = \left(\sqrt{\mathbf{H}}\right)^{-1}\mathbf{w}$. Note that this transformation is only possible if $\mathbf{H}$ is well conditioned. Due to the origin of $\mathbf{H}$, this condition is in most situations no problem. After all, $\mathbf{H}$ is the objective function of the MPC after eliminating the equality constraints. Moreover, bad condition of $\mathbf{H}$ should always be avoided with any method, since it will lead to very slow convergence. This can be illustrated by an alternative implementation of the algorithm. If instead of the optimal step, the direction of steepest descent is followed, an objective function that is not rotationally symmetric is no issue. This approach implies some other problems, such as that the step size is no longer contained in the interval $[0,1]$, and the convergence speed depends on the

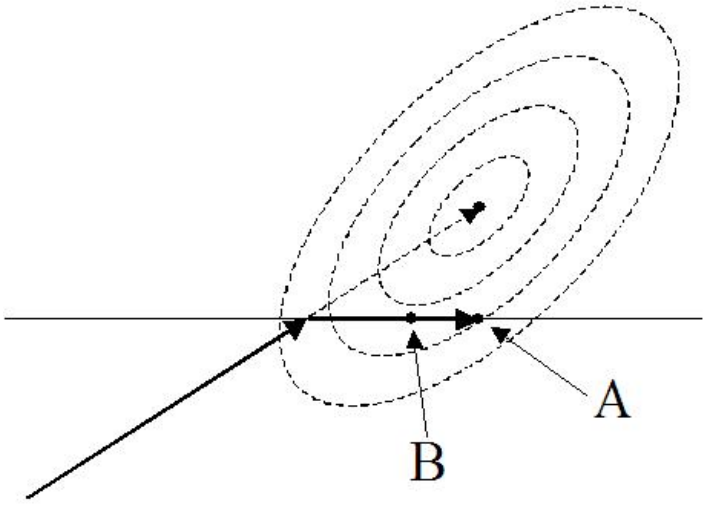


Figure 9.3: Performance of the implemented ASM if the objective function is not rotationally symmetric. A is the solution found by the algorithm. B is the correct solution. The horizontal line is a constraint. This problem is tackled in section 9.3.5.

condition of the problem. Experiments have shown that this gradient approach has more problems with a badly conditioned objective function than the earlier presented method with the transformation in order to obtain a rotationally symmetric system.

9.3.6 Stopping criterion

Different stopping criteria can be used. It should be taken into account that in a real-life application terminating the algorithm may also be necessary to avoid numerical problems. The criteria described below are implemented in the above presented method. A standard ASM will have very similar stopping criteria. Theoretical criteria, without considering the impact of numerical errors, are:

- the number of constraints equals the degrees of freedom. In this case the solution is defined exactly, and no more iterations are necessary.
- the maximal step is taken. The optimal solution is reached, and the optimizer can be terminated.

Following criteria are necessary because of time or numerical constraints:

- if the intermediate solution no longer satisfies the constraints, the optimizer should be stopped. Rounding errors may cause the system to leave the feasible region. Typically the first constraint error is only of the size of the machine precision. If the optimizer is allowed to continue, the error will grow exponentially due to the fact that an invariant condition of the ASM (the method always needs to stay in the feasible region), is violated.

- if the step in an iteration is very small, and no blocking constraint is present (which caused the algorithm to take a small step), the optimizer should be stopped as well.
- the optimizer needs to be stopped if time runs out. A possible method to implement this is to compare the available time left to the time needed for the last iteration.

Depending on the application and the circumstances, some of these criteria may be left out, and others may be added.

9.3.7 Final remarks

Some special numerical precautions need to be taken to implement this algorithm (or any algorithm for that matter) in a real-life application. Some of the precautions necessary for this method when implemented in LabView are presented in this section.

Machine precision The calculated step size and the calculated step is truncated to a lower precision than the machine precision. This keeps small numerical errors from spreading throughout the calculation.

Feasibility The feasible region is made slightly smaller for the calculation of the step size α_k by increasing the right hand side of the equation. This reduces the probability of exiting the feasible region due to numerical errors.

9.4 Initialization of an Active Set Method

9.4.1 Motivation

An ASM requires an initial point in the feasible region. For some problems this initial point may be available from physical insights, for other problems it needs to be calculated. In this section is shown how this initial value can be calculated, and how the above presented ASM can be adapted to solve the resulting Linear Program (LP). Since the initialization has to be done in every time step, computation speed is equally important for the LP and the QP.

9.4.2 Slack variables

Assume that the equality constraints have been eliminated, such that there are only inequality constraints left. The procedure presented below can be extended to the more general case of a problem with both equality and inequality constraints. To find a feasible initial point, slack variables are introduced on the constraints. The following LP is then defined:

$$\min_{(\mathbf{v},\mathbf{z})} \mathbf{e}^T \mathbf{z} \tag{9.28}$$

subject to

$$\begin{array}{rcl} \mathbf{a}_i^T \mathbf{v} & \geq & b_i + \gamma_i z_i, \quad i \in \mathcal{I}, \\ \mathbf{z} & \geq & 0. \end{array} \tag{9.29}$$

where z_i is the slack variable of the ith inequality. This LP determines a the initial value for $\mathbf{v}$ that minimizes a linear function of the slack variables. The LP needs to be initialized from a feasible point $(\mathbf{v}, \mathbf{z})$. The slack vector $\mathbf{z}$ can however be calculated from $\mathbf{v}$ as $\min\left(\frac{\mathbf{a}_i^T \mathbf{v} - b_i}{\gamma_i}\right)$, thus initialization is no issue here.

The LP in (9.28) minimizes the 1-norm of the slack vector. It is also possible to minimize the ∞-norm of the slack vector (by using one value z for all constraints) or to minimize the 2-norm (by using a quadratic objective function). Since a LP can be solved faster than a QP, the 1-norm and the ∞-norm are used the most. More information about feasibility can be found in section 2.2.4.

9.4.3 Linear programming

LPs are generally solved with the simplex method. This method is a simplified version of an ASM. Thus, with some minor modifications, the above presented ASM can be used for solving LPs as well. There are two main differences between the method for QPs and the method for LPs:

- the optimal step of the unconstrained LP would be infinitely large. Thus instead of this optimal step, the steepest descent direction of the LP is used.
- closely related to the previous remark, the optimal step size is no longer defined for a LP, whereas for a QP the optimal step size was 1. Thus the step size is allowed to be infinitely large. The stopping criterion thus has to be modified: step size is no longer relevant for the termination of the algorithm.

Concluding, by using a different calculation for the optimal step, and by removing the upper limit (i.e. the stopping criterion) on the step size, the ASM presented in the previous section can be used for solving LPs.

9.5 Computation speed

In this section the presented ASM method is compared to an IPM method. Both methods are implemented in standard LabView by the author, the IPM according to section 9.1.2, the ASM as described in section 9.3. Note there are no sparse matrix operations available in LabView. This may be a disadvantage for the IPM, but the purpose here is to find a method which is computationally tractable as well as easy to implement.

Simple problem The comparison is first made on a simple problem (Nocedal and Wright, 1989) (Fig. 9.4):

$$\min_{\mathbf{x}} q(\mathbf{x}) = (x_1 - 1)^2 + (x_2 - 2.5)^2 \tag{9.30}$$

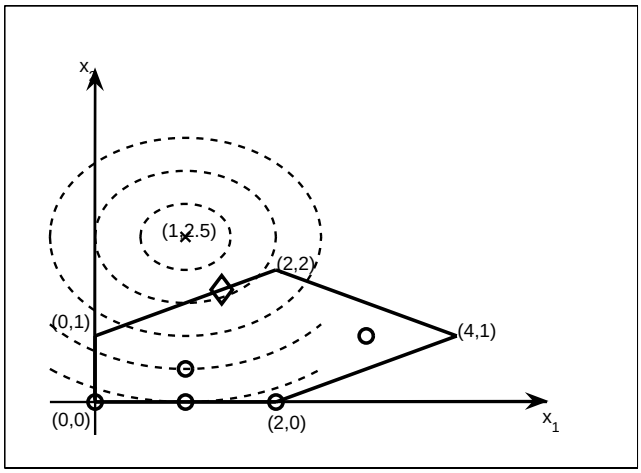


Figure 9.4: The optimization problem as described in Nocedal and Wright (1989). The 'x' marks the unconstrained optimum, the diamond marks the constrained optimum, the circles are the different initialization values, the thick lines are the constraints, the dashed lines are the contour lines of the objective function.

subject to

$$\begin{aligned} x_1 - 2x_2 + 2 &\geq 0, \\ -x_1 - 2x_2 + 6 &\geq 0, \\ x_1 + 2x_2 + 2 &\geq 0, \\ x_1 &\geq 0, \\ x_2 &\geq 0. \end{aligned} \tag{9.31}$$

The calculation time of both methods depends on the initialization value. Five different starting positions are defined on Fig. 9.4. For a real-life implementation, a real-time operating system (LabView RT) is used. This system is better suited for control because of its predictable timing behavior, while the timing behavior of a Windows system depends on the active processes and the user interaction (Craessaerts et al., 2005). The computation speed of both methods is compared on LabView RT as well as on Windows.

The real-time operating system runs on a 2.0 GHz Pentium M 760 processor with 512Mb RAM. The Windows operating system runs on an Intel Core Duo processor (T2500) at 2 GHz with 1Gb RAM. Comparing the computation speed of both systems is very difficult, since the allocation of the tasks to the different cores of the processor depends on the type of task. A possible speed criterion is the number of multiplications per second the processor can execute within a LabView for-loop (since the algorithm is also implemented in LabView). Both systems can perform about 7.10^7 multiplications per second. More details on the comparison of both systems can be found in Appendix A.

Table 9.1: The time required (in seconds) to solve the optimization problem of Eqn (9.30) for IPM and ASM, for different initialization values and different operating systems (LabView RT and Windows).

Initialization value	LabView RT		Windows	
	IPM	ASM	IPM	ASM
(0,0)	6.28E-04	2.07E-04	2.28E-03	3.44E-04
(1,0)	5.43E-04	2.08E-04	1.97E-03	3.44E-04
(2,0)	6.43E-04	2.07E-04	2.31E-03	3.28E-04
(1,0.5)	3.89E-04	2.07E-04	1.52E-03	3.44E-04
(3,1)	3.94E-04	2.09E-04	1.45E-03	3.60E-04

One optimization is too short to get a correct measurement of the execution time by the system clock, especially in Windows. Therefore each optimization is performed 1000 times, after which the average is calculated. The results are shown in Table 9.1.

The ASM outperforms the IPM by at least a factor 2 on this simple problem. The ASM is hardly influenced by the choice of the initialization value, unlike the IPM. For a real-life implementation it is a clear advantage of the ASM that the calculation time does not depend on the initialization value. The calculation time of the ASM will however depend on the number of active constraints at the solution. It is not clear what the reason is for the strong dependency of the IPM on the initialization value. It may have to do with different constraint situations at the initialization.

LabView RT is clearly faster than Windows. The performance of the Windows computer may be negatively influenced by non-optimal management of the dual core processor. Also note that IPM is almost a factor 10 slower than the ASM on Windows. It is not clear what causes this difference with LabView RT.

Real-life example The behavior of the optimization routines is illustrated with a real-life application. In Chapter 10 cruise control for the combine harvester will be discussed. More details about the system model can be found there. In this section one of the optimization problems which has occurred online is used to compare both methods.

This application is implemented at a relatively fast sampling rate (20 Hz). The objective is to regulate the machine speed towards a desired value, whilst both input signals, namely the engine speed of the diesel engine and the pump setting, remain in the feasible region. To reduce the emission, the engine speed should also be kept as low as possible. Further details can be found in Chapter 10.

The system has five states and two inputs. The prediction horizon is set at ten samples, and the control horizon is five samples. Since there are six state constraints per time sample (upper and lower limit on pump setting and engine speed, and upper and lower limit on the machine speed) and four input constraints per time sample (upper and lower limit on the change of pump setting and engine speed), the optimization problem has 70 constraints. Both methods, ASM and IPM, are compared offline on an optimization problem that occurred during real-time operation. The selected problem occurred during an acceleration when the machine speed was at about 50% of the set

point. The ASM (as described in section 9.3) solves the problem in $1.3 . 10^{-3}$ seconds whilst the IPM needs $1 . 10^{-1}$ seconds, both with LabView RT. Note that the controller in practice only has $5 . 10^{-2}$ seconds available to come up with the solution (20 Hz sampling rate), which means the IPM is useless for this application. The increasing gap between the ASM and the IPM as the problem becomes more complex, can be partly attributed to the fact that the IPM, as implemented in standard LabView, uses a normal solver instead of a sparse solver, which becomes more important for larger problems.

9.6 Conclusions

To implement MPC a QP problem needs to be solved in every time step. One of the objectives of this research is a fast QP solver, which can easily be implemented on different platforms. Control engineers often have to build a prototype application using programming environments which do not contain suited optimization algorithms.

The computation speed of general ASMs and IPMs depends on the implementation of good sparse solvers. To make the implementation more accessible (and quicker), an alternative ASM is proposed, which does not require advanced sparse solvers. This ASM is also extended to solve LPs, which are needed to determine a correct initialization value of the ASM.

This new ASM is compared to a standard IPM on a simple problem from literature. The ASM was found to be at least two times faster than the IPM. The computational performance of both methods is also compared on a larger, real-life application, namely cruise control on the combine harvester which will be discussed in Chapter 10. The computation time of the ASM is almost 100 times smaller than that of the IPM.

In conclusion, the proposed algorithm enables a swift implementation, and is much faster than the IPM when implemented using standard LabView.

Chapter 10

Application: cruise control

There is more to life than increasing its speed
-Mahatma Gandhi-



Operator comfort is a hot topic in the world of farming machinery. Operators work long days, and good operators are increasingly hard to find. Ease of operation, and operator comfort in general, are therefore important aspects in the competition between different manufacturers.

Environmental legislation is also becoming a more and more important catalyst for machine automation. Recent legislation forces the manufacturers to further reduce the noise and the exhaust fumes of their machines during road transport (European Commission, 2004, European Commission, 2005, European Commission, 2006). Thus it makes sense to reduce the engine speed whenever possible, i.e. in road conditions. In this chapter a MPC is designed to control the pump setting and the engine speed of the diesel engine such that the engine speed is minimized, without losing acceleration performance. This chapter has been published as Coen et al. (2008a) and Coen et al. (2008b).

10.1 System description

The propulsion system of a combine harvester consists of a diesel engine, which drives a variable hydrostatic pump (Fig. 10.1). The hydrostatic flow thus depends on the engine speed of the diesel engine and on the pump setting. The pump setting as well as the engine speed can be set independently by the operator. On our test machine, a New Holland CR combine harvester, the engine speed can vary between a minimum level of 1 300 rpm and a maximum level of 2 100 rpm. Note however, that different combinations of engine speed and pump setting result in the same machine speed.

During road transport the operator can vary the pump setting as well as the engine speed of the diesel engine. In field conditions this is not possible. Since some components such as the cleaning system are mechanically connected to the diesel engine, the engine has to run at maximal speed while harvesting.

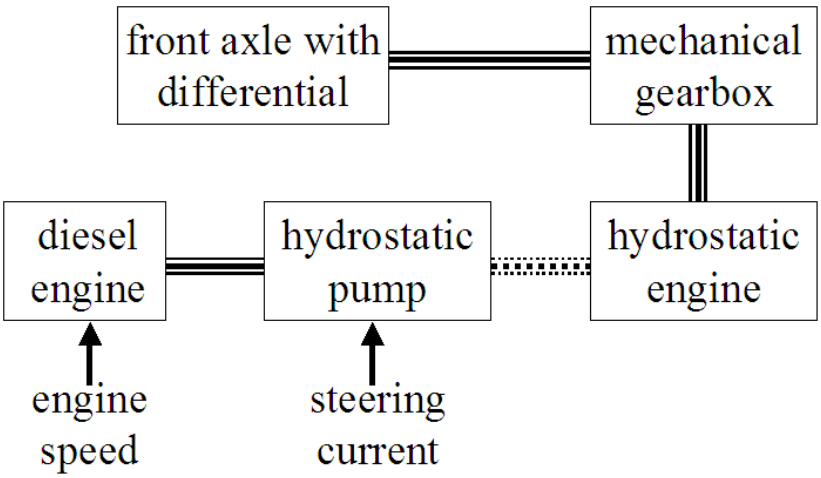


Figure 10.1: A scheme of the propulsion system of a combine harvester.

In road conditions, the speed of the machine can thus be controlled by two independent inputs, namely the pump setting and the engine speed of the diesel. An operator does not have the time to make use of this extra degree of freedom, and will always run the diesel engine at maximal engine speed, whilst controlling the machine speed with the pump setting. However, the most straightforward way to reduce the exhaust fumes and the noise, is to run with the engine speed as low as possible.

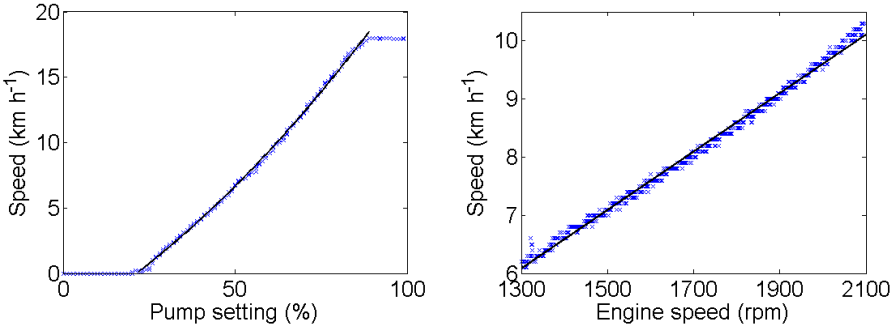
10.2 Objective

The purpose of this controller is to run the combine harvester at the lowest possible engine speed, without losing on acceleration performance. In other words, during acceleration the engine speed is allowed to increase, but after the acceleration to the set point, the engine speed has to return to the lowest possible value without causing a set point error. To reach this goal, the objective function contains the following terms:

output error: the difference between the desired and the actual machine speed.

input effort: change of engine speed and change of pump setting. This is a regularization term to make the trade-off between both control inputs, and between the control inputs and the control error.

engine speed: the engine speed minus the minimal engine speed. This term leads to a minimization of the engine speed, and thus of engine noise and exhaust fumes.



(a) (b)

Figure 10.2: The response of the speed to a slope input of (a) pump setting at 2 100 rpm (F, 20060224) and (b) engine speed at 60% pump setting (F, 20060308). The measurements are indicated with 'x'. The solid line is respectively the quadratic and the linear least squares approximation.

10.3 System modeling

Static model

First of all, a static model is derived. This model statically links pump setting and engine speed to the machine speed. To derive this model, two test series have been performed:

- slowly increase the pump setting, while the engine speed remains constant (for several engine speed values).
- slowly increase the engine speed, while the pump setting remains constant (for several pump settings).

These experiments (Fig. 10.2) show that whereas the speed is more or less linear in the engine speed, the relation between pump setting and machine speed contains a dead zone followed by a quadratic relation and finally a saturation. Note that since these experiments are performed on an asphalt road, wheel slip is negligible.

Thus, the machine speed *GS* can be written in terms of the engine speed *ES* and the pump setting *PS* as follows:

$$GS = ES \cdot f(PS) \tag{10.1}$$

where f is a nonlinear function of the pump setting. This relation can thus also be written as:

$$GS = \alpha(PS) \cdot ES \cdot PS \tag{10.2}$$

with α a pump setting-dependent amplification. This amplification can be calculated by dividing the measured relation between pump setting and machine speed by the engine speed. In practice, however, a realigned model is often used. This model is independent of the actual plant response, because it is realigned to the measurements

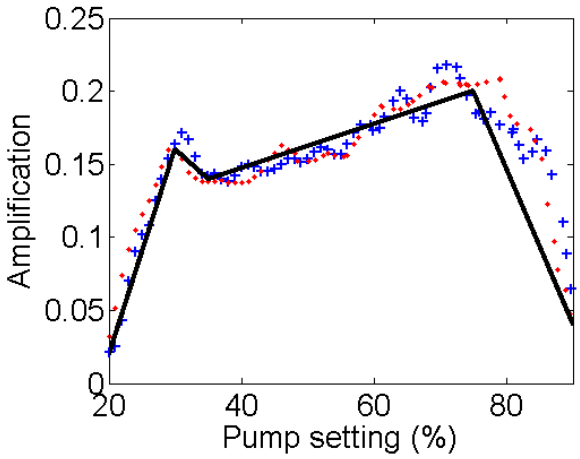


Figure 10.3: The local amplification between the pump setting-engine speed product and the machine speed as a function of the pump setting. The empirical amplification has been obtained for two engine speeds, namely at 1 300 rpm ('+') and at 2 100 rpm ('.'). Note that there are no significant differences. The solid line represents a piecewise linear approximation. For the sake of clarity the amplification has already been rescaled by multiplying with the minimal engine speed (1 300 rpm) (F, 20060307a).

in every time step (Maciejowski, 2002). It predicts the change in output as a function of the change in input. Eqn (10.2) then becomes:

$$\Delta GS \quad = \quad \beta\left(PS\right) \cdot \Delta\left(ES \cdot PS\right) \tag{10.3}$$

where β is the local amplification of the system. Eqn (10.3) is a local linear approximation (in the actual pump setting-engine speed point) of Eqn (10.2). Due to the dependency of the amplification factor on the pump setting, the local and global amplification are not the same. The local amplification β is the amplification of a local linear approximation, or in other words, β is the derivative of α (to the pump setting) in the working point. This local amplification is shown in Fig. 10.3.

Dynamic model

Since a change in the travel speed requires acceleration or deceleration of the combine harvester subject to friction, the system will only respond to relatively low frequencies. Based on the previous section in which the static model structure was shown, a dynamic state-space model structure can be proposed. Based on physical knowledge, the system contains three states:

- the actual pump setting to the hydrostatic pump
- the actual engine speed of the diesel engine
- the actual machine speed

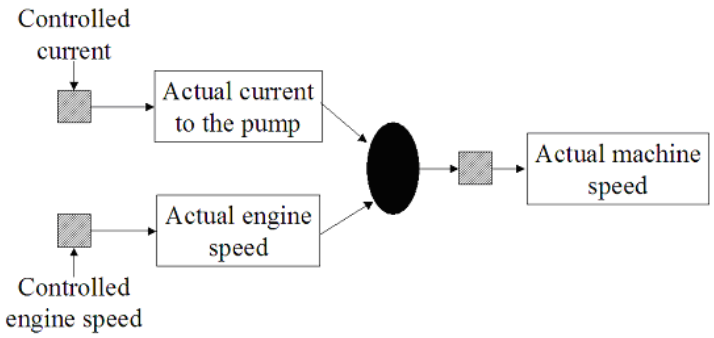


Figure 10.4: An overview of the model of the system. The actual pump setting, actual engine speed and actual travel speed are states of the system. The black oval represents the nonlinear dependency of the travel speed on the engine speed and the pump setting. The squares represent the possible locations of dynamics.

The last state, the machine speed, was shown to depend nonlinearly on the two other states through the earlier derived static model.

Before the dynamic model can be determined, the relationships in the system that may contain dynamic effects, need to be identified. Assuming that there are no dynamic effects within the nonlinear relation (of the static model), there remain three relationships in which dynamic effects can be present:

- between the pump setting set point and the actual pump setting
- between the engine speed set point and the actual engine speed
- between the static model and the actual machine speed

The general system structure as shown in Fig. 10.4, is called a Wiener-Hammerstein structure (Nesic, 1999, Nelles, 2001), because it is a linear dynamic system, followed by a nonlinearity and again a linear dynamic system.

Experiments have shown that the engine speed and the pump setting to the hydrostatic valve respond sufficiently fast such that those dynamics can be neglected. This means that only the relationship between the nonlinear transformation of the pump setting and the engine speed on one hand and the actual machine speed on the other hand, contains dynamics. Multisine excitations (Pintelon and Schoukens, 2001) show that the dynamics form a low-pass filter, as could be expected.

For MPC the prediction performance of the model is essential. Moreover, because the system is nonlinear, the system response depends on the type of input. Since the most typical input for the controller will be a stepwise increase of the reference speed, the dynamics are also modeled based on step responses. It is actually not important for the dynamics whether this step input is applied to the engine speed or to the pump setting, because both result in an increase in hydraulic flow, and thus give the same dynamical response. The dynamics of the hydraulic flow actually come from the swash plate dynamics of the pump, which can be described as a second order system (Jelali and Kroll, 2003, Watton, 1989).

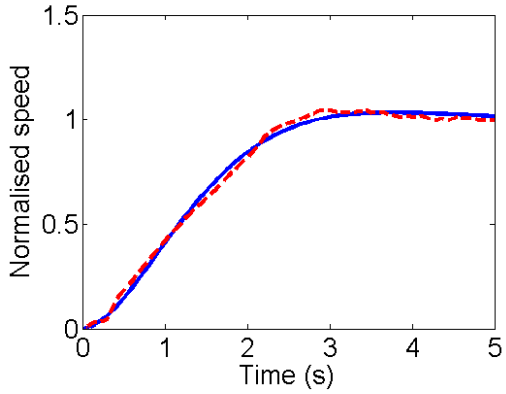


Figure 10.5: The normalized response of the second order estimated model (solid line) and the response of the real system (dashed line) on a step input (F, 20060307b).

Therefore, the following continuous time model structure is proposed, which is a general second order system:

$$G(s) = \frac{1}{s^2/\omega^2 + 2\xi s/\omega + 1} \tag{10.4}$$

with resonance frequency ω (rad s^{-1}) and damping ξ. The parameters are determined by means of a least squares method on the measured step response (Fig. 10.5), and have the following values:

$$\omega = 1.2250 \text{ rad s}^{-1} \tag{10.5}$$
$$\xi = 0.7336 \tag{10.6}$$

This model nicely fits to the experimental data as can be seen from Fig. 10.5.

Linearization and discretization

If possible, the best approach to nonlinear MPC is to linearize the model online, depending on the current working point. This way the linearization error is minimal, and the optimization problem remains a QP problem. Working with a nonlinear model in the optimization requires more complex solution methods, which has a significant impact on the computation time.

First of all the system needs to be discretized. The sampling rate is chosen to be 20 Hz, which is the highest rate possible on this machine. The discretization is performed with the Tustin discretization rule (Franklin et al., 2006). This leads to the following system:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \\ \mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k \end{cases} \tag{10.7}$$

with $\mathbf{u}_k$ a function of the pump setting-engine speed product (the static model of Eqn (10.2), (10.3)), $\mathbf{y}_k$ the ground speed and $\mathbf{x}_k$ the states of the system.

The actual inputs of the model should however be the change of the pump setting ΔPS_k and the change of engine speed ΔES_k. The input $\mathbf{u}_k$ thus needs to be rewritten in terms of these variables, and then linearized. Using Eqn (10.2) and (10.3), $\mathbf{u}_k$ can be written as follows:

$$\begin{aligned}
\mathbf{u}_k &= \alpha\left(PS_k\right) \cdot PS_k \cdot ES_k && (10.8)\\
&= \alpha\left(PS_k\right) \cdot \left(PS_{k-1} + \Delta PS_k\right) \cdot \left(ES_{k-1} + \Delta ES_k\right) && (10.9)\\
&\approx \alpha\left(PS_{k-1}\right) \cdot PS_{k-1} \cdot ES_{k-1} + \beta\left(PS_{k-1}\right) \cdot PS_{k-1} \cdot \Delta ES_k \\
&\quad + \beta\left(PS_{k-1}\right) \cdot \Delta PS_k \cdot ES_{k-1} + \beta\left(PS_{k-1}\right) \cdot \Delta PS_k \cdot \Delta ES_k && (10.10)
\end{aligned}$$

where α is the global amplification and β is the local amplification. Note that after linearizing the system, the local amplification as defined in Eqn (10.3) appears in the equations. By neglecting the second order term and defining a new input vector $\mathbf{v}_k$ containing the change of the pump setting and the change of the engine speed at time k, $\mathbf{u}_k$ can be written as:

$$\mathbf{u}_k = \alpha\left(PS_{k-1}\right) \cdot PS_{k-1} \cdot ES_{k-1} + \mathbf{B}_u \mathbf{v}_k \tag{10.11}$$

where

$$\begin{aligned}
\mathbf{B}_u &= \beta\left(PS_{k-1}\right) \cdot \left[ES_{k-1} PS_{k-1}\right] \\
\mathbf{v}_k &= \begin{bmatrix} \Delta PS_k \\ \Delta ES_k \end{bmatrix}
\end{aligned} \tag{10.12}$$

The system input thus becomes $\mathbf{v}_k$, and Eqn (10.11) becomes an extra system state with state equation:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \mathbf{B}_u \mathbf{v}_k \tag{10.13}$$

Note that $\mathbf{B}_u$ depends on the current state of the system. Using a constant $\mathbf{B}_u$ across the prediction horizon thus implies a linearization. To keep the pump setting and the engine speed within their physical range, constraints can be imposed. However, constraints can only be placed on inputs, states and outputs. Since the only inputs after linearization are ΔPS_k and ΔES_k, integrator states need to be added to calculate PS_k and ES_k. In this way constraints can be imposed on the calculated PS_k and ES_k. The extended state-space description then becomes:

$$\begin{cases}
\mathbf{x}_{k+1} = \mathbf{A}_L \mathbf{x}_k + \mathbf{B}_L \mathbf{v}_k, \\
\mathbf{y}_k = \mathbf{C}_L \mathbf{x}_k + \mathbf{D}_L \mathbf{v}_k
\end{cases} \tag{10.14}$$

with system matrices $\mathbf{A}_L \in \mathbb{R}^{5\times5}$, $\mathbf{B}_L \in \mathbb{R}^{5\times2}$, $\mathbf{C}_L \in \mathbb{R}^{1\times5}$ and $\mathbf{D}_L \in \mathbb{R}^{1\times2}$. The state vector $\mathbf{x}_k$ now contains five elements: the two model states of Eqn (10.4), the input state of Eqn (10.13) and two integrator states for PS_k and ES_k. The system matrices can be found in Appendix F, section F.2.

Three of these five system states cannot be measured, namely the two model states of Eqn (10.4) and the input state of Eqn (10.13). These states thus need to be estimated using a state estimator to compensate for model inaccuracy.

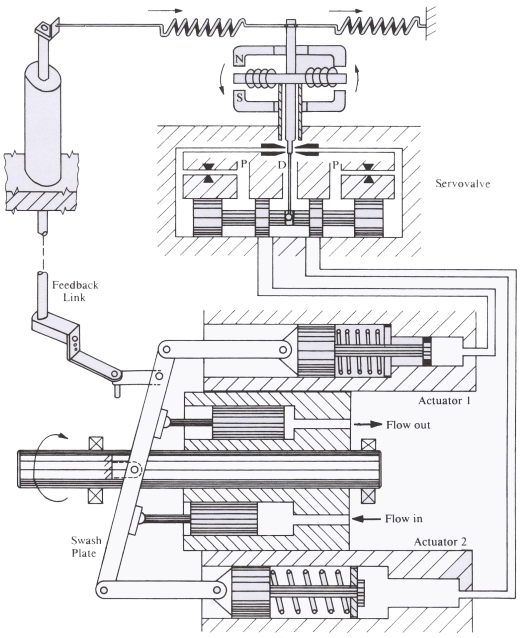


Figure 10.6: Schematic of the hydraulic pump in the propulsion system of the combine harvester.

Disturbances

Several experiments have been performed to investigate the sensitivity of the model to disturbances as road slope and machine mass. The mass of the combine harvester may vary considerably as a function of the grain in the bin.

In Fig. 10.6 a schematic of the hydraulic pump is shown. At the top of the figure the electric coil is located. The current to this coil is called the pump setting. This electric coil is coupled to a servo valve through a flipper, which on its turn controls the swash plate position of the hydraulic pump at the bottom of the figure. Note that there is a mechanical feedback present between the swash plate and the flipper.

A current change causes the flipper to move to the left or to the right. The servo valve then creates a pressure difference between both cylinders connected to the swash plate, causing the swash plate to tilt. The mechanical feedback places the flipper back in its central position. A current thus creates a force on the flipper which is counteracted by the force of the mechanical feedback. The pump itself is driven by the diesel engine. Because of its construction the flow only depends on its configuration (pump setting). As long as the diesel engine can deliver the necessary power, the machine speed will only vary if the pump setting or the engine speed change. Thus, the mass of the grain bin and the terrain slope theoretically do not influence the propulsion system.

However, because of the limitations of the diesel engine, the mass in the grain grain bin and the terrain slope may affect the machine speed in field conditions, when the threshing mechanism is on. In road conditions, it is very rare that the diesel engine

cannot provide the required power.

10.4 Nonlinear MPC

The operator has two degrees of freedom to control the machine speed whilst driving on the road, namely the pump setting and the engine speed. The objective of the cruise control system is to utilize this extra degree of freedom to minimize the exhaust fumes and the noise production of the combine harvester without losing performance. The lower the engine speed of the diesel engine, the less exhaust fumes and noise.

10.4.1 Objective definition

As already indicated above, the purpose of the control system is to reduce the engine speed without losing acceleration performance. The purpose of this controller is that the engine speed increases during acceleration, and then decreases again to the minimal possible engine speed needed to reach the requested speed set point.

The objective of the controller is to minimize the set point error, between the actual machine speed on one hand and desired machine speed on the other hand, with minimal input cost. The input cost is the change of engine speed and pump setting. This cost is chosen because it is related to the comfort of the operator. In order to minimize the engine speed an extra penalty term in the engine speed state is added to the objective function. This is the mathematical translation of our objective to operate the machine at minimal engine speed. The equality constraints are formed by the dynamic model of the system of Eqn (10.14). The inequality constraints are input constraints, such as on the change of pump setting, and state constraints, for instance on the pump setting and engine speed state. The input constraints are imposed to guarantee operator comfort, the state constraints confine the pump setting and engine speed to their physical region. The squared set point error, engine speed cost and input cost are summed to a quadratic objective function. The optimization problem at every time step thus becomes:

$$\min V(\mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{y}_1, \ldots, \mathbf{y}_N, \mathbf{u}_0, \ldots, \mathbf{u}_{M-1})$$

$$\begin{aligned} &V(\mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{y}_1, \ldots, \mathbf{y}_N, \mathbf{u}_0, \ldots, \mathbf{u}_{M-1}) = \\ &\sum_{k=1}^{N} (\mathbf{y}_k - \mathbf{r}_{y,k})^T \mathbf{S} (\mathbf{y}_k - \mathbf{r}_{y,k}) + \sum_{k=1}^{N} (\mathbf{x}_k - \mathbf{r}_{x,k})^T \mathbf{Q} (\mathbf{x}_k - \mathbf{r}_{x,k}) \\ &+ \sum_{k=0}^{M-1} \mathbf{v}_k^T \mathbf{R} \mathbf{v}_k \end{aligned} \tag{10.15}$$

subject to

$$\begin{aligned} &\left\{ \begin{array}{rcll} \mathbf{x}_{k+1} & = & \mathbf{A}_L \mathbf{x}_k + \mathbf{B}_L \mathbf{v_k} & \\ \mathbf{y}_k & = & \mathbf{C}_L \mathbf{x}_k + \mathbf{D}_L \mathbf{v}_k & k = 0 \ldots N-1 \end{array} \right. \\ &\left\{ \begin{array}{ll} \mathbf{x}_k \in \mathcal{X}, & k = 1 \ldots N \\ \mathbf{y}_k \in \mathcal{Y}, & k = 0 \ldots N-1 \\ \mathbf{v}_k \in \mathcal{V}, & k = 0 \ldots M-1 \end{array} \right. \end{aligned} \tag{10.16}$$

where $\mathbf{r}_{y,k}$ is the desired machine speed, $\mathbf{S} \in \mathbb{R}^{1\times1}$ is the weight on the set point error, $\mathbf{r}_{x,k}$ contains the desired state values, $\mathbf{Q} \in \mathbb{R}^{5\times5}$ is the weight on the state errors and $\mathbf{R} \in \mathbb{R}^{2\times2}$ is the input cost. The matrices $\mathbf{Q}$, $\mathbf{R}$ and $\mathbf{S}$ have the following structure:

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & w_{ES} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{R} = \begin{bmatrix} w_{\Delta PS} & 0 \\ 0 & w_{\Delta ES} \end{bmatrix}$$

$$\mathbf{S} = \begin{bmatrix} w_{error} \end{bmatrix} \tag{10.17}$$

where w_{ES} is the weight accorded to the engine speed cost, $w_{\Delta PS}$ is the penalty for a unit change of the pump setting, $w_{\Delta ES}$ is the penalty for a unit change of the engine speed and w_{error} is the penalty for a unit difference between the desired and the actual machine speed. The engine speed cost is defined as the squared difference between the actual engine speed and the minimal engine speed (i.e. 1 300 rpm). $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{V}$ represent the feasible region for respectively the vectors $\mathbf{x}_k$, $\mathbf{y}_k$ and $\mathbf{v}_k$. These regions are defined by the following inequality constraints:

$$\begin{aligned} \mathcal{X} &= \left\{ \mathbf{x} \in \mathbb{R}^{5\times1} \left| \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x} \geq \begin{bmatrix} PS_{min} \\ ES_{min} \\ -PS_{max} \\ -ES_{max} \end{bmatrix} \right. \right\} \\ \mathcal{Y} &= \left\{ \mathbf{y} \in \mathbb{R}^{1\times1} \left| \begin{bmatrix} 1 \\ -1 \end{bmatrix} \mathbf{y} \geq \begin{bmatrix} GS_{min} \\ -GS_{max} \end{bmatrix} \right. \right\} \\ \mathcal{V} &= \left\{ \mathbf{v} \in \mathbb{R}^{2\times1} \left| \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \mathbf{v} \geq \begin{bmatrix} \Delta PS_{min} \\ \Delta ES_{min} \\ -\Delta PS_{max} \\ -\Delta ES_{max} \end{bmatrix} \right. \right\} \end{aligned} \tag{10.18}$$

with PS_{min} the minimal pump setting (expressed in %), PS_{max} the maximal pump setting, ES_{min} the minimal engine speed (expressed in rpm), ES_{max} the maximal engine speed, GS_{min} the minimal machine speed, GS_{max} the maximal machine speed, ΔPS_{min} the minimal change of pump setting (expressed in %), ΔPS_{max} the maximal change of pump setting, ΔES_{min} the minimal change of engine speed (expressed in rpm) and ΔES_{max} the maximal change of engine speed. This description assumes that the feasible region is constant over the horizon, but if necessary this formulation can easily be extended to time-varying constraints.

A logical way to define the trade-off between both control inputs is to base it on the squared ratio of their ranges. The maximal pump setting change (ΔPS_{min},ΔPS_{max}) is limited to ±5 per time step, whereas the engine speed change ($\Delta ES_m in$, $\Delta ES_m ax$) is limited to ±100 per time step. Assume the weight of the change of pump setting is 1, the weight of the change of engine speed then becomes 0.0025 ($= \left(\frac{5}{100}\right)^2$). The weight of the control error is tuned experimentally. The best results are obtained with a control error weight equal to 1. If the control error weight is 1, a 1 km/h error leads to the same cost as a 1% change of the pump setting. Note that since the input is the change of pump setting, the trade-off between the pump setting and the speed error is influenced by the control rate. For this application the control rate is 20 Hz.

This formulation of the objective means that the system will always try to lower the engine speed to the minimal engine speed. Since high speeds can only be maintained if the engine speed is above the minimal engine speed, a trade-off will occur between the control error and the minimization of the engine speed. In other words, the engine speed will not return to its minimal value, but there will remain a control error in steady state.

This issue can be solved by including an extra integrator state I in the system, and thus enlarge the system to six states. This extra state will integrate the difference between the set point and the actual speed. In order not to interfere with the fast acceleration dynamics, this integrator I will be very slow. The objective function is then extended with a penalty term on this integrator state. As long as there is a control error present, the integrator term will increase, allowing for a larger engine speed.

The integrator I always needs to remain larger than zero, and has to integrate the sum of the control error and the difference between the pump setting and the maximal pump setting. In this way I will increase if there is a control error, and decrease if the pump setting is not at the maximal value. The combination of these two terms disables the integrator during transients as long as the pump setting is increasing. The results of this approach will be illustrated in the section below.

10.4.2 Experimental set-up

For the experiments a New Holland CR combine harvester was used. This machine was equipped with a LabView PXI system as described in Chapter 5. In order to test the controller step changes were applied to the reference input of the controller. The experiments were performed on an asphalt parking lot.

10.4.3 Results

The above described controller is tested on the road for accelerations up till 12 km/h. The controller is tested with and without the engine speed minimization term. The effect of adding this engine speed minimization term is shown in Fig. 10.7. Since the engine speed minimization only kicks in after a few seconds, the operator did not feel a difference in acceleration speed. Moreover, because the noise was reduced at the reference speed, the cruise control with engine speed minimization was experienced as more comfortable by the operator.

Both MPCs start from minimal engine speed. In practice, the operator would always run the combine at maximal engine speed. Although the acceleration is much slower when accelerating at minimal engine speed than when accelerating at maximal speed, the operator does not sense a difference between accelerating with the engine at maximal speed and the engine speed controlled by the MPC. The acceleration feels equally smooth.

As can be seen in Fig. 10.7, the decreasing engine speed is compensated by the increasing pump setting. As already described in the previous section, this will cause problems at high speeds, because the pump setting cannot become sufficiently large to compensate for the decreasing engine speed. Due to limitations at the test site, the upper limit on the pump setting was lowered to 50% to be able show this effect at lower speeds. The maximal speed at minimum engine speed was about 5.5 km/h. In

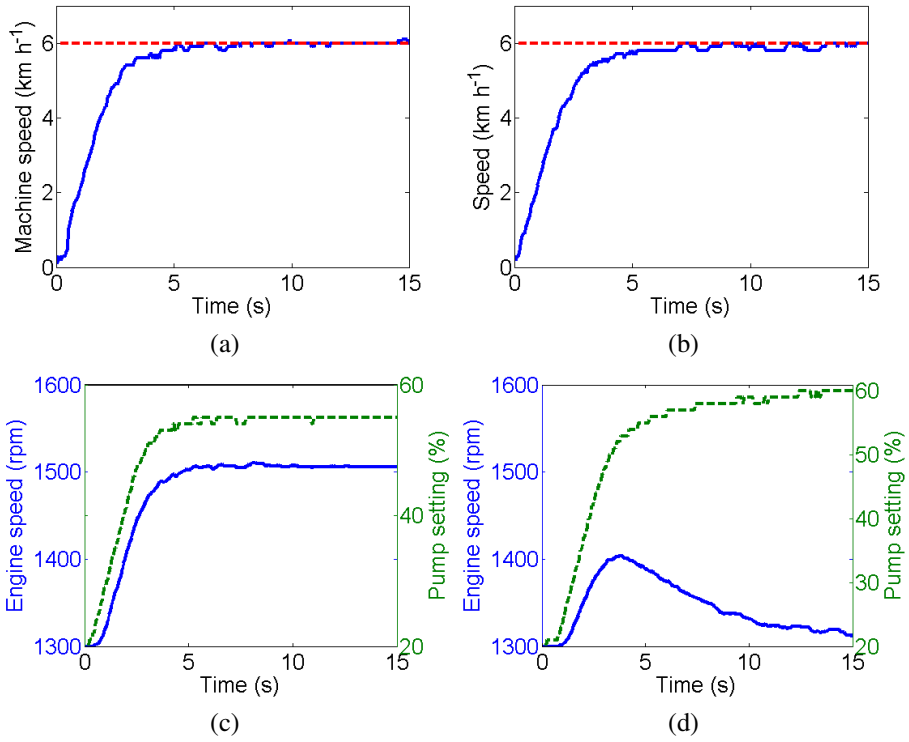


Figure 10.7: Response to a step change in the desired travel speed for MPC controller (a),(c) without engine speed minimization and (b),(d) with engine speed minimization. (a),(b) show the speed response. The solid line represents the travel speed and the dashed line represents the reference.
(c),(d) show the input signals. The dashed line represents the pump setting (right axis) and the solid line represents the engine speed (left axis).

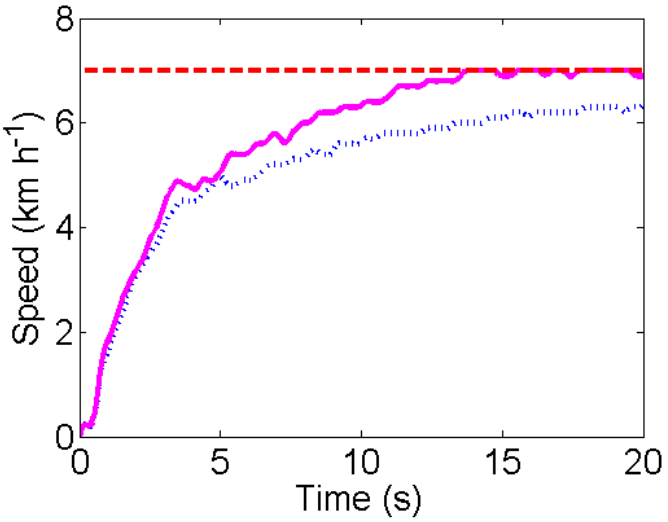


Figure 10.8: Response to a step change in the desired speed for an MPC controller with engine speed minimization. The pump setting is limited to 50% to provoke a steady-state error at low speeds. The dashed line represents the reference at 7 km/h, the dotted line represents the standard controller and the solid line represents the same controller with an extra integrator to avoid a steady-state error.

Fig. 10.8 a step response to 7 km/h of the controller is shown with and without the integrator term. Adding the extra integrator state thus removes the steady state error, while keeping the engine speed as low as possible.

10.5 Conclusions

This chapter presents a cruise control system for a combine harvester in road conditions. Because of increasing legislation, the combine manufacturers need to reduce the exhausts and the noise production of their machines. On the road, the speed depends on two input signals, namely the pump setting and the engine speed of the diesel engine. It is shown that MPC can use this extra degree of freedom to run the combine at minimal engine speed, without losing acceleration performance.

Chapter 11

Application: bale density control

The proof of the pudding is in the eating
-English proverb-



Straw and hay bales are often transported by truck over hundreds of kilometers. Straw and hay are sold per kilogram, but in most cases the number of bales which can be moved with one truck is constrained by the volume, and not by the weight. Thus, increasing the bale density, and as a result the truck weight, leads to clear economic benefits: the transport cost per kg decreases. The objective of a bale density control system is to deliver bales of the density set by the operator, respecting machine constraints such as the maximal force exerted by the plunger. As the crop properties change, the baling process will change as well. Even a small change in moisture content of the crop may cause an entirely different behavior.

This chapter applies the Stochastic MPC methodology developed in Chapter 8 of this thesis to the bale density control problem. First of all the system is described and the control objective is presented. The parts of the baler which are relevant for the density control algorithm are then modeled. Finally Stochastic MPC is implemented on the baler, and tested in the field.

11.1 System description

This chapter implements Stochastic MPC on large square balers (Fig. 11.1 (a)). This section describes the working principles of a baler. There exist a number of different types of balers, but since the field experiments make use of a New Holland BB950A baler, this section also focuses on this type of machine.

A square baler consists of a pick-up unit, a precompression chamber, a compression chamber and a tying mechanism. In most modern balers the crop intake in the bale chamber is from below, which enables the baler to be pulled behind the tractor. Earlier balers were pulled partially at the side of the tractor because they had to be fed sideways.



(a) (b)

Figure 11.1: (a) A picture of a New Holland big square baler in action (W, New Holland Benelux (2008)).
(b) A close-up of the rear of the bale chamber.

The pick-up unit feeds the crop, such as straw or hay, in the precompression chamber. The precompression chamber is the temporary storage, where one slice of the baler is prepared. The actual compression chamber contains a plunger which is coupled to the tractor's PTO (a power take-off) and rotates at a fixed speed. For a New Holland BB950A this plunger has a period of about 1.4 seconds. Once the precompression chamber is full, the contents of the precompression chamber is moved to the compression chamber when the plunger is at its farthest point. In most balers this synchronization is performed mechanically.

The plunger then pushes the new slice against the existing bale, moving the bale through the compression channel. Thus the compression action of the plunger actually combines two effects:

- the new slice is compressed.
- the entire bale is moved through the compression chamber.

This means that two crop properties are important, namely the compressibility of the crop and the friction coefficient between the bale and the bale chamber. The amount of compression will be determined by the distribution of the plunger force over the two effects. The lesser the bale moves, the more the new slice will be compressed. How much the bale moves is influenced by hydraulic cylinders at the end of the bale chamber, which can apply a variable lateral pressure to the bale in the chamber (see Fig. 11.1 (b)). In order to be able to vary the lateral pressure, the rear opening of the bale chamber is also made variable.

A baler is limited by the maximal force the plunger can exert, the maximal amount of crop the pick-up can cope with and the maximal lateral pressure on the rear of the bale chamber. The maximal plunger force is secured with a safety bolt. The density of the bale is related to the force exerted by the plunger. According to Srivastava et al. (2006), assuming only two opposite sides of the bale chamber converge, the force can be calculated as follows:

$$F_c \quad \propto \quad \frac{E_h}{d_c} f_h \tag{11.1}$$

with F_c the compressive force, E_h the effective modulus of elasticity of the crop, d_c the depth of the bale chamber and f_h the coefficient of friction between the crop and the bale chamber. It is clear that the friction and the elasticity modulus depend on the actual crop condition, which will vary in time. Also note that on a New Holland BB950 both vertical sides as well as the top side converge, thus Eqn (11.1) is only approximately valid.

On the test machine the pressure on the hydraulic cylinder on the bale chamber can only vary slowly. Due to the thin pipes from the pump to the cylinder, the pressure is rate-limited. Since it is very large, the cylinder can only move very slowly. This is not relevant in normal operation, where the cylinder does not move that much, it just exerts pressure on the bale. The pressure depends on the current through the valve. This current is controlled by a PWM signal, which is the control variable. The resistance which converts the PWM signal to a current is sensitive to the temperature, and thus time-varying.

One of the main tasks of the operator is to adjust the speed to assure that the precompression chamber is full just before a plunger stroke occurs. Otherwise the precompression chamber will be overfilled, which implies a too high compression in the precompression chamber. If there is too much compression in the precompression chamber, the compression of the slices will vary in function of the height of the slice. This leads to asymmetric slices, which on their turn lead to a crooked bale.

11.2 Objective

It is assumed that the density of the bale is proportional to the force on the plunger. The objective of the density controller thus is to keep the force measurement as close as possible to the set point set by the operator, while respecting the following constraints:

force limit: to guard the mechanical integrity of the system, the force on the plunger has to be kept below the maximal allowable force. Otherwise the safety bolt will break.

pressure limit: the pressure exerted on the end of the bale has to be kept smaller than the maximal pressure for which it has been designed.

current and PWM limit: the current and the PWM signal to the pressure valve have to be constrained to the specifications of the valve.

The density control system has to be able to cope with several different operating regimes, the most important of which are:

normal operation: the baler is continuously in the crop. This is the most common situation.

crop entry in normal operation: the baler re-enters the crop. Between crop exit and crop entry, there are a lot so-called non-active plunger strokes, these are plunger strokes without new material entering the main compression chamber. Every non-active plunger stroke that occurs, the bale gets a bit more stuck in the bale chamber. Crop entry occurs very often.

crop entry with zero pressure: each time the machine is shut down, the pressure is removed from the cylinders of the bale chamber. If the machine then re-enters the crop, it will take some time before the pressure is back at its normal level. This situation occurs at least once every field. After all, between two fields the baler is shut down to drive on the road.

crop entry with empty bale chamber: if the machine has to drive on the road, or if the bale chamber has to be cleaned, the bale chamber is emptied for which the bale needs to be removed. This is a very complex situation to control, because the bale chamber will have to close completely at the start, and then re-open very fast once the material starts to accumulate in the chamber to avoid overload.

Bale density control thus shows the same challenges as capacity control on the combine harvester:

- respect the constraints on force, pressure, current and PWM.
- time-varying system which depends on the crop properties.
- maintain maximal performance in all circumstances.

11.3 Data exploration

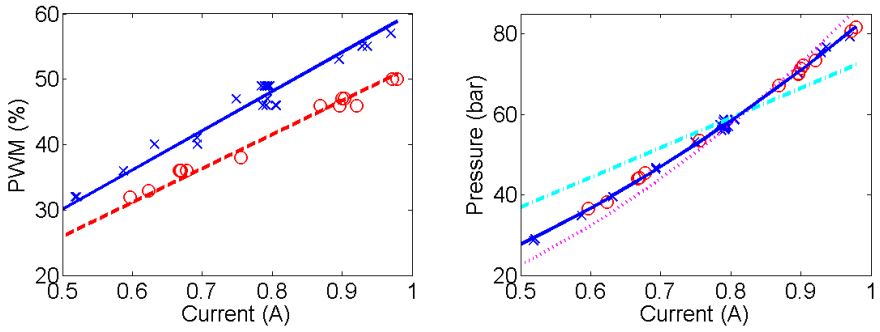
For density control there are four relevant system variables, namely:

- peak force on the plunger F_B, available every active plunger stroke (maximally every 1.4 seconds)
- pressure on the bale chamber P_B
- current to the pressure valve I_P
- PWM signal to the pressure valve V_P

The relation between force and pressure depends on the crop properties, and is thus time-varying. In this section the relations between pressure, current and the PWM signal are explored by applying several constant PWM levels to the pressure valve.

This experiment is performed two times on the same field. The first time around noon, and the second time around three o'clock. This way the effect of the oil temperature can also be examined. This is especially important for the relation between current and PWM. In Fig. 11.2 (a) the steady state value of the PWM signal is plotted as a function of the current. For each of the experiments the linear least squares approximation (with intercept zero) is also shown. The slope is proportional to the electrical resistance. For the first experiment the R squared value is 0.96 and the gain is 60.2. For the second experiment the R squared value is 0.98 and the gain is 51.9. The resistance has thus decreased with almost 15% in a couple of hours time. This time-variability of the resistance will have to be taken into account by the current controller.

In Fig. 11.2 (b) the steady-state value of the pressure and the current is shown. There is no noticeable difference between both experiments. However, one can assume that this relation between pressure and current will be dependent on the machine



(a) (b)

Figure 11.2: (a) The stationary relation between PWM and current in the first experiment ('x') and in the second experiment ('o') a couple of hours later with their respective trend line.
(b) The stationary relation between current and pressure in the first experiment ('x') and in the second experiment ('o'). The dash-dot line is a linear approximation (intercept 0). The solid line is a quadratic approximation (intercept 0), the dotted line is quadratic approximation with only a quadratic term (F, 20070811b).

type. Thus the relation will have to be estimated online as well. In Fig. 11.2 (b) least squares approximations with the following three functions are shown:

1. $y = \alpha x$: the R squared value is 0.87, $\alpha = 73.9$ (dash-dot line)
2. $y = \alpha x^2$: the R squared value is 0.96, $\alpha = 89.9$ (dotted line)
3. $y = \alpha x^2 + \beta x$: the R squared value is 0.998, $\alpha = 59.1$ and $\beta = 25.7$ (solid line)

where x is the current and y the pressure. An approximation with an intercept different from zero is not considered because this makes it very complicated to update the model parameters online. The actual relation between both variables is clearly best approximated by the third function. However, since it is better to limit the adaptive parameters to one, the second relation is the best choice for the actual implementation. The R squared is only slightly smaller, and only one parameter needs to be estimated.

It should come as no surprise that the relation between current and pressure is quadratic. This is very common for pressure valves (Jelali and Kroll, 2003, Watton, 1989).

11.4 System modeling

This section focuses on modeling the baler process. The PWM signal V_P is the actual control signal of the system. V_P results in a current I_P, which on its turn leads to a pressure P_B in the cylinders of the bale chamber. As shown in the previous section, the amplification between V_P and I_P depends on the electrical resistance of the pressure valve, and is time-varying.

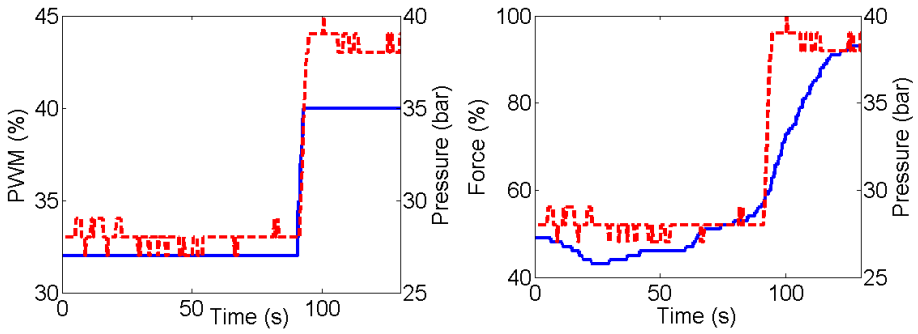


Figure 11.3: The step response of the density system of a big square baler.
(a) The PWM signal (solid line) and the pressure signal (dashed line) during a step change of the PWM.
(b) The force signal (solid line) and the pressure signal (dashed line) during a step change of the PWM (F, 20070811a).

The rate of change of the pressure P_B is limited to a change of about 5 bar/s by the oil flow that can be maximally fed to the cylinders. This nonlinearity poses a challenge to the density controller. The exact value of the rate limit also depends on the properties of the crop currently in the bale chamber. Moreover, field tests confirm that the pressure P_B depends on the square of the current to the valve I_P (Fig. 11.2 (b)). The model for pressure will thus be in terms of $C_{P,squared}$.

Statically, with constant crop properties, the force on the plunger F_B is proportional to the pressure P_B applied to the bale chamber. Of course, the relation between F_B and P_B is dependent on the properties of the crop currently in the bale chamber. The density of the bale can be assumed to be proportional to the force F_B. Currently a lot of work is being done to measure the density online for each bale that exits the machine. This would make it possible to continuously update the relation between density and force F_B.

The step responses of the force and pressure signals are shown in Fig. 11.3 (F, 20070811a). For the controller implementation this system is modeled as follows:

Force model:

$$\begin{aligned} F_{B,k+1} &= F_{B,k} + K_{P_B,F_B}\gamma_F\left(T_{F,k} + u_{P,k}\right) \\ T_{F,k+1} &= \left(1-\gamma_F\right)\left(T_{F,k} + u_{P,k}\right) \\ P_{B,k+1} &= P_{B,k} + u_{P,k} \end{aligned} \tag{11.2}$$

Pressure model:

$$\begin{aligned} P_{B,k+1} &= P_{B,k} + K_{I_P,P_B}\gamma_P\left(T_{P,k} + u_{C,k,squared}\right) \\ T_{P,k+1} &= \left(1-\gamma_P\right)\left(T_{P,k} + u_{C,k,squared}\right) \\ C_{P,k+1,squared} &= C_{P,k,squared} + u_{C,k,squared} \end{aligned} \tag{11.3}$$

with $u_{P,k}$ and $u_{C,k,squared}$ respectively the change of pressure and the change of the squared current. K_{P_B,F_B} and K_{I_P,P_B} are the respective amplifications between

pressure and force, and current squared and pressure. $T_{F,k}$ and $T_{P,k}$ are respectively states of the force and the pressure model. γ_F and γ_P are parameters of the first order force and pressure models. The matrix description of these models can be found in Appendix F, section F.3. The parameters γ_F and γ_P are estimated on the step responses to be:

$$\begin{array}{rclrcl} \gamma_F & = & 0.333 & f_F & \leq & 0.046 \text{ Hz} \\ \gamma_P & = & 0.2 & f_F & = & 0.178 \text{ Hz} \end{array} \tag{11.4}$$

where f_F and f_P represent the respective cut-off frequencies. For the force only an upper limit of the cut-off frequency is known since the force model is only executed on an active plunger stroke. The minimal time between two plunger strokes is 1.4 seconds. The pressure model is executed at 5 Hz.

11.5 Stochastic MPC

11.5.1 Control structure

The plunger force can only be measured at a very low sampling rate, maximally once every 1.4 seconds. Since the pressure P_B is related to the force F_B, and is measured at a much higher sampling rate than the force, a master-slave structure is in place (Franklin et al., 2006). Moreover, partially due to the rate-limit on the pressure, the dynamics of the pressure signal are much slower than the electric dynamics of the pressure valve (between I_P and V_P). This leads to the following three-stage control structure (Fig. 11.4):

1. *Force controller:* this controller updates at every active plunger stroke, which occurs at multiples of 1.4 seconds. The output is a pressure set point based on the current force measurement. This controller should take the maximal force and maximal pressure into account.

2. *Pressure controller:* the controller updates at 5 Hz. The output is a current set point based on the pressure value. The controller needs to take the maximal pressure and the maximal current into account.

3. *Current controller:* the controller can update very fast, at for instance 25 Hz, since all used variables are electrical. The output is the PWM signal V_B sent to the pressure valve. This output is calculated based on the measured current. The controller should take the maximal PWM signal and maximal current into account.

Each stage has its own challenges. The relation between force and pressure contains slow dynamics, which need to be modeled. Between current and pressure the rate limit of the pressure is important for the control performance. The quadratic relation between current and pressure also needs to be modeled. For the current-PWM relation especially the time-varying behavior is important. There are no significant dynamics present, but the controller gain needs to adapt to the circumstances.

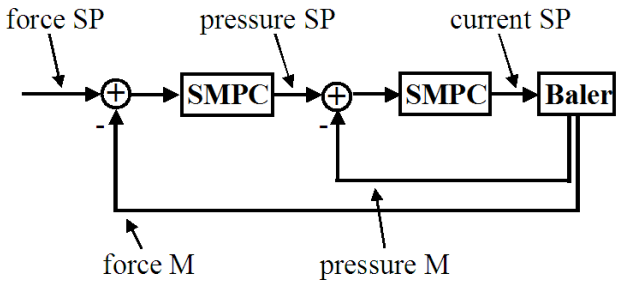


Figure 11.4: The control structure of the bale density control system. 'SP' indicates a setpoint, 'M' indicates a measurement. Inside the baler block an Adaptive PID controller guards the current set point.

11.5.2 Objective definition

As stated above, the controller consists of three stages. The force and the pressure controller will both be implemented using the Stochastic MPC methodology. The current controller is a variable gain PID controller to accommodate the time-varying resistance. This chapter only discusses the force and the pressure controller. Below the objective function and the constraints are presented for the force controller and the pressure controller.

Force controller The objective of the master controller is to minimize the set point error between the actual force and desired force with minimal input cost. The input cost is the change of pressure.

The inequality constraints are input constraints, on change of the pressure, and state constraints, on force, pressure and on the difference between pressure and pressure set point. To avoid wind-up, the pressure set point should never become too much larger than the actual pressure. This difference can for instance be limited to 5 bar. The optimization problem at every time step thus becomes:

$$\min V(\mathbf{y}_1, \ldots, \mathbf{y}_{N_P}, \mathbf{u}_0, \ldots, \mathbf{u}_{N_C-1})$$

$$\begin{aligned} V(\mathbf{y}_1, \ldots, \mathbf{y}_{N_P}, \mathbf{u}_0, \ldots, \mathbf{u}_{N_C-1}) &= \sum_{k=1}^{N_P} (\mathbf{y}_k - \mathbf{r}_{y,k})^T \mathbf{S}_F (\mathbf{y}_k - \mathbf{r}_{y,k}) \\ &\quad + \sum_{k=0}^{N_C-1} \mathbf{u}_k^T \mathbf{R}_F \mathbf{u}_k \end{aligned} \tag{11.5}$$

subject to

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}_F \mathbf{x}_k + \mathbf{B}_F \mathbf{u}_k & k = 0 \ldots N_P - 1 \\ \mathbf{y}_k = \mathbf{C}_F \mathbf{x}_k + \mathbf{D}_F \mathbf{u}_k & \end{cases}$$

$$\begin{cases} \mathbf{x}_k \in \mathcal{X}, & k = 1 \ldots N_P \\ \mathbf{u}_k \in \mathcal{U}, & k = 0 \ldots N_C - 1 \end{cases}$$

where $\mathbf{r}_{y,k}$ is the desired force, $\mathbf{S}_F \in \mathbb{R}^{1\times 1}$ is the weight on the set point error and $\mathbf{R}_F \in \mathbb{R}^{1\times 1}$ is the weight on the input cost. The matrices $\mathbf{R}_F$ and $\mathbf{S}_F$ have the following structure:

$$\mathbf{R}_F = \left[\ w_{\Delta P_B}\ \right] \quad \mathbf{S}_F = \left[\ w_{error}\ \right] \tag{11.6}$$

where $w_{\Delta P_B}$ is the cost of the change of the pressure and w_{error} is the cost of the difference between the desired and the actual force. $\mathcal{X}$ and $\mathcal{U}$ represent the feasible region for respectively the vectors $\mathbf{x}_k$ and $\mathbf{u}_k$. These regions are defined by the following inequality constraints:

$$\begin{aligned}
\mathcal{X} &= \left\{ \mathbf{x} \in \mathbb{R}^{3\times 1} \left| \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \mathbf{x} \geq \begin{bmatrix} F_{B,min} \\ P_{B,min} \\ -F_{B,max} \\ -P_{B,max} \\ P_{B,actual} - \Delta_{max} \\ -P_{B,actual} - \Delta_{max} \end{bmatrix} \right. \right\} \\
\mathcal{U} &= \left\{ \mathbf{u} \in \mathbb{R}^{1\times 1} \left| \begin{bmatrix} 1 \\ -1 \end{bmatrix} \mathbf{u} \geq \begin{bmatrix} \Delta P_{B,min} \\ -\Delta P_{B,max} \end{bmatrix} \right. \right\}
\end{aligned}$$

with $F_{B,min}$ the minimal force (expressed in %), $F_{B,max}$ the maximal force, $P_{B,min}$ the minimal pressure (expressed in bar), $P_{B,max}$ the maximal pressure, $P_{B,actual}$ the actual pressure measurement ($P_{B,k}$ is the pressure set point in the force controller model), Δ_{max} the maximal difference between the pressure set point and the actual value, $\Delta P_{B,min}$ the minimal change of pressure (expressed in %) and $\Delta P_{B,max}$ the maximal change of pressure. This description assumes that the feasible region is constant over the horizon. Of course this formulation can easily be extended to time-varying constraints.

Pressure controller The pressure controller will minimize the set point error between the actual and the desired pressure with minimal input cost. The input cost is the change of the squared current.

The inequality constraints are input constraints, on change of the squared current, and state constraints, on pressure, squared current and on the change of pressure. The constraint on the change of pressure includes the rate limit of the system in the controller. This constraint can easily be written as a constraint on the change of squared current. The optimization problem at every time step thus becomes:

$$\min V(\mathbf{y}_1, \ldots, \mathbf{y}_{N_P}, \mathbf{u}_0, \ldots, \mathbf{u}_{N_C-1})$$

$$\begin{aligned}
V(\mathbf{y}_1, \ldots, \mathbf{y}_{N_P}, \mathbf{u}_0, \ldots, \mathbf{u}_{N_C-1}) &= \sum_{k=1}^{N_P} (\mathbf{y}_k - \mathbf{r}_{y,k})^T \mathbf{S}_P (\mathbf{y}_k - \mathbf{r}_{y,k}) \\
&\quad + \sum_{k=0}^{N_C-1} \mathbf{u}_k^T \mathbf{R}_P \mathbf{u}_k
\end{aligned} \tag{11.7}$$

subject to

$$\begin{cases} \mathbf{x}_{k+1} &= \mathbf{A}_P\mathbf{x}_k + \mathbf{B}_P\mathbf{u}_k \\ \mathbf{y}_k &= \mathbf{C}_P\mathbf{x}_k + \mathbf{D}_P\mathbf{u}_k \end{cases} \qquad k = 0 \ldots N_P - 1$$

$$\begin{cases} \mathbf{x}_k \in \mathcal{X}, & k = 1 \ldots N_P \\ \mathbf{u}_k \in \mathcal{U}, & k = 0 \ldots N_C - 1 \end{cases}$$

where $\mathbf{r}_{y,k}$ is the desired pressure, $\mathbf{S}_P \in \mathbb{R}^{1\times 1}$ is the weight on the set point error and $\mathbf{R}_P \in \mathbb{R}^{1\times 1}$ is the input cost. The matrices $\mathbf{R}_P$ and $\mathbf{S}_P$ have the following structure:

$$\mathbf{R}_P = \begin{bmatrix} w_{\Delta I_P} \end{bmatrix} \quad \mathbf{S}_P = \begin{bmatrix} w_{error} \end{bmatrix} \tag{11.8}$$

where $w_{\Delta C_{P,squared}}$ is the cost of the change of the squared current and w_{error} is the cost of the difference between the desired and the actual pressure. $\mathcal{X}$ and $\mathcal{U}$ represent the feasible region for respectively the vectors $\mathbf{x}_k$ and $\mathbf{u}_k$. These regions are defined by the following inequality constraints:

$$\begin{aligned} \mathcal{X} &= \left\{ \mathbf{x} \in \mathbb{R}^{3\times 1} \left| \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \mathbf{x} \geq \begin{bmatrix} P_{B,min} \\ C_{P,min} \\ -P_{B,max} \\ -C_{P,max} \end{bmatrix} \right. \right\} \\ \mathcal{U} &= \left\{ \mathbf{u} \in \mathbb{R}^{1\times 1} \left| \begin{bmatrix} 1 \\ -1 \end{bmatrix} \mathbf{u} \geq \begin{bmatrix} \Delta C_{P,min} \\ -\Delta C_{P,max} \end{bmatrix} \right. \right\} \end{aligned}$$

with $P_{B,min}$ the minimal pressure (expressed in bar), $P_{B,max}$ the maximal pressure, $C_{P,min}$ the minimal squared current (expressed in A^2), $C_{P,max}$ the maximal squared current, $\Delta C_{P,min}$ the minimal change of squared current (expressed in A^2) and $\Delta C_{P,max}$ the maximal change of squared current.

11.5.3 Parameter uncertainty

Just like for the capacity control application, the state uncertainty is assumed to depend solely on the parameter uncertainty. This implies that only input changes give rise to uncertainty. On the square baler a somewhat different approach is taken to calculate the parameter uncertainty. There is, however, no specific reason to use a different calculation for the parameter uncertainty for the square baler than for the combine harvester. The same technique of Chapter 8 could be used here as well.

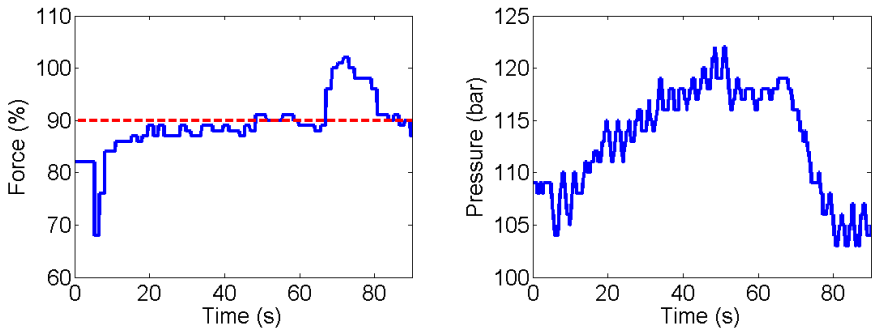
The process gains K_{P_B,F_B} and K_{I_P,P_B} are calculated using Fuzzy RLS as described in Chapter 6, section 6.5. This way the parameter error can be assumed to be normally distributed. The error e_{model} of the force and the pressure model of Eqn 11.2 and Eqn 11.3 can be calculated as follows:

$$e_{model}(k) = y_p(k) - y_m \tag{11.9}$$

where $y_p(k)$ is the output predicted by the model and $y_m(k)$ is the measured output both at time step k. The parameter uncertainty ΔK can then be defined as:

$$|e_{model}(k)| = \Delta K(k) \cdot |\Delta x| \tag{11.10}$$

with Δx the change of input. ΔK can then be estimated using Fuzzy RLS. The following precautions are taken:



(a) (b)

Figure 11.5: (a) The force set point (dashed line) and the measured force (solid line). Halfway this measurement the speed of the tractor is increased considerably, causing the force to increase.
(b) The measured pressure on the hydraulic cylinders. The pressure needs to be decreased to keep the force on the same level (F, 20080731).

- if $|\Delta x| \leq \Delta x_{min}$ or $|e_{model}| \leq e_{model,min}$ the Fuzzy RLS algorithm is disabled, and replaced by a moving average with the same update rate as follows:

$$\Delta K(k+1) \quad = \quad (1-\gamma)\Delta K + \gamma \Delta K_{min} \tag{11.11}$$

  with γ the standard Fuzzy RLS update rate. This way the uncertainty always returns to the minimal uncertainty if there is no significant model error.

- ΔK is bounded as follows: $\Delta K_{min} \leq \Delta K \leq \Delta K_{max}$ with possibly ΔK_{min} and ΔK_{max} expressed as a function of the actual parameter value K_{P_B,F_B} or K_{I_P,P_B}. This way the variation of the uncertainty taken into account by the controller is more or less under control.

11.5.4 Results

Due to bad weather conditions in France in the summer of 2008, it was not possible to do extensive testing with this set-up. This section should thus be seen as a preview of what is possible with Stochastic MPC in field conditions.

An interesting way to test the responsiveness of this control system is to vary the speed of the tractor pulling the baler. If the speed is changed, the relation between pressure and force will also vary. In the field experiment in Fig. 11.5 the tractor speed is considerably increased, which causes the force on the plunger to rise. The pressure then decreases swiftly to steer the force back to the set point level. The drop in the force measurement at the start of the run marks crop entry. Before crop entry, there is no new force measurement available, and the measurement is frozen to its last value. At the start of the run, the pressure needs to be increased to control the force towards the set point.

The pressure on the hydraulic cylinders is determined by the current to the pressure valve, and also depends on the properties of the bale in the bale chamber. As

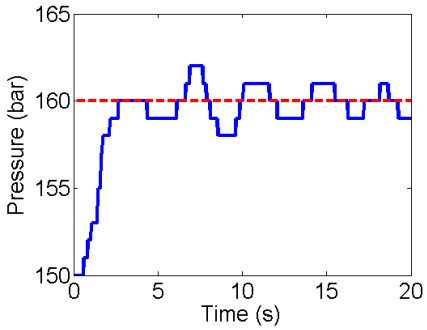


Figure 11.6: The measured pressure (solid line) and the pressure limit (dashed line).

mentioned earlier, the pressure needs to be limited to avoid damaging the cylinders. When entering the crop with zero pressure or with an empty bale chamber, or when baling extremely dry straw, the pressure limit becomes active. Fig. 11.6 shows an experiment where the machine enters the crop with zero pressure. The controller then encounters the pressure limit of 160 bar. Note that although the resolution of the pressure measurement is limited to 1 bar, the oscillation around the constraint does not exceed 2 bar. With the standard commercial controller the system oscillates about 10 bar around the constraint.

11.6 Conclusions

This chapter illustrates that Stochastic MPC as developed in this research (Chapter 8) can be more generally applied. It is an interesting methodology for all applications with a time-varying model uncertainty where performance is the main objective. By taking into account the instantaneous model uncertainty, the controller will respond as swiftly as possible in all circumstances.

Due to weather conditions this application could not be fully tested in the field. The results shown here are obtained without tuning the controller parameters (the values assumed based on system knowledge were used), since there was no time to analyze the data and tune the controller. These tests should be repeated more extensively to be able to accurately establish the controller performance. Even with these initial parameters the controller already performs quite well.

Chapter 12

Conclusions & future research

God does not pay weekly, but he pays at the end.
-Dutch proverb-



This chapter presents the conclusions of this research. First of all the methodology presented in this work to control an uncertain system is discussed. Secondly the results are summarized for the three applications considered in this research, namely capacity control on a combine harvester, cruise control on a combine harvester and bale density control on a square baler. Finally some possibilities for future research are given.

12.1 Conclusions

12.1.1 Stochastic MPC methodology

In this research a methodology to include model uncertainty in the control problem has been derived. The objectives of this research, presented in Chapter 4, can now be evaluated.

Objective 4.1: System model In Chapter 6 the model of the capacity control application has been derived. First of all, the relations between the process variables are presented, based on expert knowledge. Then dynamic models are determined where necessary, in order to guarantee a good time prediction behavior of the model. After all, for MPC good time prediction is the most important requirement. Delay plays an important role in this system, because of the large transport delays. The variation of the delay, as well as some influence factors have been examined. Within the same crop the variation of the delay is limited, but there are considerable differences in delay between for instance wheat and corn.

Due to the time-varying and nonlinear nature of the process, an adaptive model is required. Different estimation techniques have been compared on field data. For the further steps of the Stochastic MPC methodology a normally distributed parameter error is a very interesting feature. A Recursive Least Squares (RLS) estimator, a Fuzzy

RLS estimator and a Sliding Mode estimator are considered. Fuzzy RLS is a modification to RLS introduced by the author. It simply means that the update rate of the RLS depends on the error. If the error is larger, the update rate is made larger. This way Fuzzy RLS can avoid large parameter errors, and render the parameter error normally distributed. For the rest of the research Fuzzy RLS is used as parameter estimation technique. Note that the actual distribution of the parameter is irrelevant. Stochastic MPC is only concerned with the distribution of the actual parameter around the time-varying parameter estimate (the parameter error). In other words, although the parameter itself may not be normally distributed, the parameter error can be normally distributed by using the proper estimation technique.

Objective 4.2: Capacity control In this work a capacity control system is developed which can control feed rate, engine load and grain mass flow, subject to constraints on ground speed, feed rate, engine load and input changes. In Chapter 7 a classic MPC methodology is followed to design a capacity controller. The important tuning parameters of MPC, such as the prediction and control horizon, are explained. The effect of these parameters is illustrated with field tests.

With classic MPC and a normal controller tuning, which is acceptable for the operator, it is not possible to respect the constraints on this system, because the model is too uncertain. Only by tuning the controller extremely conservatively, the system can be made stable in the neighborhood of constraints.

Objective 4.3: Model and process uncertainty When not taken into account explicitly, uncertainty severely limits the controller performance, especially when considered over a wide range of fields with different properties. In this work a methodology is proposed to take this uncertainty into account.

The objective of taking the uncertainty into account is two-fold:

- make the controller optimal irrespective of the field conditions.
- detune the controller near constraints as a function of the uncertainty.

For this application, three types of model error are relevant:

slow drift: the model parameters, and especially the gain factors, are related to biological and machine properties such as crop density, which slowly vary through the field. This type of model error is minimized by using an adaptive model.

input induced error: due to the slow drift of the process parameters, and due to the nonlinearities in the system, the linear adaptive model is never exact. This implies that an input change does not have the expected effect.

step disturbances: terrain conditions, such as muddy, change of slope or switching on/off of the 4×4, can cause process variables to suddenly increase or decrease.

In this work the focus lies on the input induced error. Only model error caused by an input change, is taken into account by the controller as model uncertainty. After all, slow drift is dealt with by the adaptive model. Considering step disturbances as model uncertainty without a good prediction model, would lead to a conservative solution.

Moreover, for this type of applications it is more important to have a fast controller, which can cope with disturbances as they occur, than to avoid constraint violation by disturbances completely by designing a conservative controller.

The model error caused by input changes depends on the parameter error. This parameter error can be quantified by the online estimator. This work presents a procedure to derive a model for the state uncertainty, based on the process model and making use of the estimated parameter uncertainty. The predicted state uncertainty is an estimator for the standard deviation of the state, and is a function of the input change. The uncertainty model starts from the capacity control problem, and thus only deals with uncertain parameters in a forward path. In other words, the state uncertainty is never multiplied by the uncertain parameter that caused it. In this case, one can easily see that if a process gain is uncertain, the state uncertainty will be proportional to the input change (multiplicative uncertainty).

A methodology based on Stochastic MPC is proposed in Chapter 8 which takes the model uncertainty into account. The objective function of classic MPC is replaced by the expected value of the objective function. The constraints are replaced by probabilistic constraints.

Taking the expected value implies that the objective function includes a deterministic control error term and a state uncertainty term. This state uncertainty term is predicted using the uncertainty model, based on the parameter uncertainty, and is a function of the input change. In other words, Stochastic MPC can be seen as an automatic tuning of the objective function, as a function of the model uncertainty. This way the controller tuning remains optimal, even in changing field and machine conditions.

The uncertainty model can only be calculated exactly if the uncertain parameters only occur in the input matrix. In this work a pragmatic solution is proposed to deal with uncertain parameters in the state matrix. This is however only an approximation, and subject to a number of assumptions.

Probabilistic constraints are constraints that are satisfied with a given probability. To be able to do this, the parameter error is assumed to be normally distributed. The concept is very simple. Instead of requiring a state to be smaller than some upper bound, the state increased with some factor γ times the state uncertainty needs to be smaller than the upper bound. The factor γ of course depends on the probability level, and as mentioned above, the state uncertainty is an estimate for the standard deviation of the state. For instance, to satisfy a constraint in 98% of the cases, γ needs to be 2.

Objective 4.4: Implementation In this research, implementation plays an important role. The controllers need to be tested and evaluated on the real-life application, and it should be possible to implement the controller for a commercial application in the foreseeable future. Thus, special attention has been paid to the computational complexity of the controller.

The controller was implemented on a National Instruments PXI system (Appendix A and Appendix B). The most time-consuming operation is solving the optimization problem. In Chapter 9 a modified Active Set Method (ASM) is designed. This method has two benefits over a standard ASM:

- whereas a standard ASM requires specialized algorithms to solve sparse problems, this ASM takes a more graphical interpretation. It actually comes down

to projection of the search direction onto the active set. This graphical interpretation of ASM is more suited to implement in LabView because this is a graphical programming language, and there is a library of basic matrix operations available.

- since computation speed is an issue, as much calculations as possible are performed offline. The graphical ASM interpretation requires the equality constraints to be eliminated offline.

Note however, that these equality constraints are the process model, which has been made adaptive. Thus, the equality constraints are time-varying, and the optimization matrices need to be updated at a fixed time interval. In this work the optimization matrices are updated at 1 Hz, which more than suffices. The system does not change that fast that it would be interesting to update the optimization matrices at 5 Hz, which is the control rate and the update rate of the estimators. The computational complexity of the model update rate is function of the prediction and the control horizon (Appendix C).

Objective 4.5: Validation First of all, in Chapter 10, a cruise control system is implemented on a combine harvester using MPC. This is a nice example of what is possible with MPC for the automation of agricultural machinery. This cruise control system reduces the exhaust fumes and the engine noise (which both depend on the engine speed), without affecting the driving experience of the operator.

In Chapter 11 the Stochastic MPC methodology is used to design a bale density controller on a large square baler. Although the Stochastic MPC methodology is derived based on the capacity control application, it can be generalized for other applications. The results for the bale density control application are very promising.

12.1.2 Applications

IntelliCruise

At the start of this research, experiments were performed with PID and fuzzy control. These experiments served as performance benchmark for the later developed MPC.

One of the feed rate controllers designed for this benchmark exercise was later on implemented for the commercial machine. It is now sold as an optional feature to the New Holland flagship combine harvester (New Holland CR and CX) under the brand name 'IntelliCruise'. IntelliCruise was awarded a silver medal for innovating an existing feature on AgriTechnica 2007, the world's largest exhibition for agricultural machinery.

MPC for capacity control on a combine harvester

The main application of this research is capacity control on the combine harvester. MPC is very well suited for this application because:

- MPC can deal very well with large time delays without losing performance.
- by using an adaptive model, an approximate (linearized) model can be used, which is robust to environmental and mechanical changes (see Chapter 6).

- constraints can easily be included (Chapter 8).
- if information of the field in front of the combine is available (for instance a yield map of the previous year or a GPS map of the area), this information can be used in the controller. This would be be an interesting extension of the current controller.
- the controller objective can easily be changed in the field. In other words, the same controller can be used for controlling the speed, the engine load and the grain mass flow.
- the controller can be made dependent on the current field condition by changing the model and including the model uncertainty (see Chapter 8).

The only downside of MPC is that it requires much more work to implement the controller for the commercial machine than a classic controller such as a PID. MPC also raises the performance requirements (and thus the cost) for the processor hardware. It is a computationally intensive procedure, and it is recommended to use a floating-point processor instead of a cheaper fixed-point processor which is still often used in industry. However, the optimization procedure presented in Chapter 9 is very well suited to implement MPC efficiently on processors for which there are little or no mathematical libraries available.

Cruise control on the combine harvester

On the road the operator has two inputs to control the machine speed, namely the pump setting and the engine speed of the diesel engine. The machine speed depends on the product of the pump setting and the engine speed by second order dynamics. In harvesting conditions it is not allowed to vary the engine speed since it needs to be at maximal level for proper operation of the threshing and cleaning mechanism. The operator however hardly ever uses this extra degree of freedom. On the road the combine usually runs at maximal engine speed and only the pump setting is used to vary the speed.

New legislation (European Commission) forces manufacturers to reduce the noise level and the exhaust fumes of their machinery, which are both related to the engine speed of the diesel engine. The extra degree of freedom available to control the machine speed can be used by a controller to reduce the average engine speed.

In Chapter 10 a controller is designed using MPC which keeps the engine speed as low as possible, without compromising on acceleration performance. The nonlinear system is linearized in every time step around the current working point. During acceleration the engine speed is increased to make the necessary torque available. Once the machine speed approaches its set point, the engine speed is lowered again. This decrease in engine speed is compensated by increasing the pump setting further. An integrator state needs to be added to the system to avoid steady-state errors when the requested speed cannot be reached at minimal engine speed.

Bale density control on a square baler

In Chapter 11 the Stochastic MPC approach is validated on another agricultural machine with a time-varying, uncertain model, namely a large square baler. For this

application the objective is to regulate the pressure on the compression chamber such that the force on the plunger is kept on the set point. The control variable is the pressure on the end of the compression chamber. The force on the plunger is assumed to be proportional to the density of the bales. The controller also needs to respect a number of constraints such as a force and pressure limit.

Due to a large difference in dynamics and sampling rate, a three-stage control structure is used. The force is only measured at every active plunger stroke, which means that there is at least 1.4 seconds between two measurements. The pressure varies much slower than the current (hydraulics versus electronics), and can thus be sampled and controlled at a lower rate.

Stage 1 keeps the current to the hydraulic valve on its set point using an Adaptive PID controller. Stage 2 generates a current set point based on a pressure set point and stage 3 calculates a pressure set point from the force set point. Stage 2 and 3 make use of the Stochastic MPC methodology. Whilst stage 1 is executed at 25 Hz and stage 2 at 5 Hz, stage 3 is event-driven, and is only executed each active plunger stroke, or in other words, each time a new measurement value is available.

Due to weather conditions it was not possible to extensively tune and test this controller, but the first results are promising. The controller seems to deal quite well with special conditions such as crop entry with an empty bale chamber. The main benefits of Stochastic MPC for this application lie in its constant performance independent of the crop conditions, even in special situations such as crop entry. Further tests are however needed to evaluate the performance of this controller.

12.2 Future research

More general uncertainty model The uncertainty model used in Chapter 8 for Stochastic MPC can only be calculated exactly if the uncertain parameters occur in the input matrix. Uncertain parameters in the state matrix are dealt with in a very ad hoc manner. Moreover, this approach is only possible for parameters which are in a forward path starting from the input. In other words, the calculated state uncertainty may never be multiplied by the parameter that gave rise to the uncertainty in the first place. It has to be possible to relax these limitations and design for a more general uncertainty model.

Incorporation of the feedback effect MPC is in a sense an open loop optimization technique. The optimization assumes that the complete calculated input sequence is applied to the system, without considering new process values. In practice however, this is not the case. Because the optimization is repeated in every time step, the controller is actually closed loop. Not taking this 'closed loop effect' into account leads to conservative solutions. It is interesting to examine whether including this feedback effect in the Stochastic MPC approach would improve the performance.

Delays Further study is required to examine whether it is possible to determine the delay online, in field conditions. The crop entry and exit might be used for this purpose. If the delay can be obtained online, the influence of an accurate delay estimate

on the controller performance can be analyzed. Moreover, if the delay is determined online, it is no longer necessary to change the delay as a function of the crop type.

Model step disturbances In this work step disturbances are not taken into account. The only way to deal with this type of disturbances without inserting conservatism into the controller, is to model the disturbance. For instance, engaging four-wheel drive affects the system model, and causes a sudden decrease of the machine speed. If this disturbance can be modeled, and thus predicted, the controller can avoid or at least reduce the disturbance of the machine speed.

Benchmark capacity control To be able to objectively quantify the performance of the capacity control system designed in this research, the system needs to be compared to the performance of an experienced operator and to the performance of other capacity control systems which are commercially available.

Due to the dependency of the process on the crop input, a reliable comparison of the performance of the controllers and the operator requires that each is tested in similar conditions. The best way to guarantee similar conditions for all the controllers and the operator is to use a large field with 'homogenous variation'. In other words, variation is required to be able to assess the performance, but in order to have similar input for all the experiments, the variation needs to be predictable. For instance a field where the crop density increases in one direction would be an interesting test field. This experiment then needs to be repeated in different conditions and for different crops and varieties.

Bale density control Due to weather conditions the field test experiments were not completed. The bale density control system has to be further tuned and tested. In this research a relatively simple process model was used. It may be possible to improve the controller performance by using a more complex model.

Bibliography

M. Amoodeh, M. Khoshtaghaza, and S. Minaei. Acoustic on-line grain moisture meter. *Computers and Electronics in Agriculture*, 52:71–78, 2006.

J. Anthonis, K. Maertens, G. Strubbe, J. De Baerdemaeker, and H. Ramon. Design of a friction independant mass flow sensor by force measurement on a circular chute. *Biosystems Engineering*, 84(2):127–136, 2003.

Y. Arkun, J. Hollet, W. Canney, and M. Morari. Experimental study of internal model control. *Industrial & Engineering Chemistry Process Design and Development*, 25: 102–108, 1986.

K. Astrom and B. Wittenmark. On self tuning regulators. *Automatica*, 34(1):90–113, 1973.

I. Batina. Model predictive control for stochastic systems by randomized algorithms. PhD thesis, Technische Universiteit Eindhoven, Eindhoven, Netherlands, 2004.

J. Baumgarten, C. Bussmann, and A. Wilken. Agricultural working machine. United States Patent Application US2006/0123757 A1, 2005.

W. Behnke. Method and apparatus for determining optimal adjustments of work units in an agricultural harvesting machine. United States Patent Application US2003/0066277 A1, 2002.

W. Behnke, N. Diekhans, J. Huster, and G. Quincke. Automatic control of a top transfer device of a combine harvester so that it correctly directs harvested grain into a collection trailer by use of a satellite or gps positioning devices. European Patent Application EP 1 219 159 A1, 2002.

T. Bellemans, B. De Schutter, and B. De Moor. Model predictive control for ramp metering of motorway traffic: a case study. *Control Engineering Practice*, 14(5): 441–466, 2006.

A. Bemporad and C. Filippi. Suboptimal explicit mpc via approximate multiparametric quadratic programming. In *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, Florida, USA, 2001.

A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38:3–20, 2002.

A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos. Corrigendum to 'the explicit linear quadratic regulator for constrained systems'. *Automatica*, 39:1845–1846, 2003.

D. Bernstein. Feedback control: an invisible thread in the history of technology. *IEEE Control Systems Magazine*, 22(2):53–68, 2002.

G. Best. Alternative energy crops for agricultural machinery biofuels - focus on biodiesel. In *Proceedings of the 16th Members' Meeting of the Club of Bologna*, Bologna, Italy, Nov. 2005.

D. Bichel and G. Cornisch. The sidehill 6600 combine - a new concept for the corn belt. ASAE Meeting Paper Number 74-1599, 1974.

R. Bitmead, M. Gevers, and V. Wertz. *Adaptive optimal control - the thinking man's GPC*. Prentice hall, USA, 1990.

X. Blasco, M. Martinez, J. Herrero, C. Ramos, and J. Sanchis. Model-based predictive control of greenhouse climate for reducing energy and water consumption. *Computers and Electronics in Agriculture*, 55(1):49–70, 2007.

Bloomberg. Tarweprijs haalt record, suiker wordt goedkoper. De Tijd, 2007. September 27.

F. Borelli, M. Baotic, A. Bemporad, and M. Morari. Efficient on-line computation of closed-form constrained optimal control laws. In *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, Florida, USA, 2001.

S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, UK, 2004.

H. Broman, M. Pope, and T. Hansson. A mathematical model of the impact response of the seated subject. *Medical Engineering & Physics*, 18(5):410–419, 1996.

T. Burks, S. Shearer, J. Fulton, and C. Sobolik. Effects off time-varying inflow rates on combine yield monitor accuracy. *Applied Engineering in Agriculture*, 20:269–275, 2004.

C. Bussmann and N. Diekhans. Prozessabhängige Vorfahrtregelung beim Mähdrescher. In *Proceedings of VDI Mähdrescher Kolloquium*, Dresden, Germany, Mar. 2007.

E. Camacho and C. Bordons. *Model Predictive Control*. Springer, UK, second edition, 2004.

J. Chaplin, N. Hemming, and B. Hetchler. Comparison of impact plate and torque-based grain mass flow sensors. *Transaction of the ASAE*, 47(4):1337–1345, 2004.

T. Chateau, C. Debain, F. Collange, L. Trassoudaine, and J. Alizon. Automatic guidance of agricultural vehicles using a laser sensor. *Computers and Electronics in Agriculture*, 28:243–257, 2000.

C. Chen and L. Shaw. On receding horizon feedback control. *Automatica*, 18:349–352, 1982.

D. Clarke and P. Gawthrop. Generalisation of the self-tuning regulator. *Electronics Letters*, 11(2):40–41, 1975.

D. Clarke, C. Mohtadi, and P. Tuffs. Generalized predictive control - part i. *Automatica*, 23:137–148, 1987a.

D. Clarke, C. Mohtadi, and P. Tuffs. Generalized predictive control - part ii. *Automatica*, 23:149–160, 1987b.

L. Clarke. On the agricultural mechanization development in african countries. strategies and roles of the private sector and the governments. In *Proceedings of the 14th Members' Meeting of the Club of Bologna*, Bologna, Italy, Nov. 2003.

T. Coen. Iwt scholarship 2004-2008. Internal report, MeBioS, Katholieke Universiteit Leuven, Leuven, Belgium, 2008.

T. Coen and J. De Baerdemaeker. Nonlinear model-based predictive control for cruise control. In *Proceedings of the VDI Mähdrescher Kolloquium*, Dresden, Germany, Mar. 2007.

T. Coen and B. Missotten. Apparatus and method for controlling the speed of a combine harvester. pending, 2007.

T. Coen, J. Anthonis, and J. De Baerdemaeker. Nonlinear model-based predictive control for cruise control on combine harvesters. In *Proceedings of the European Control Conference (ECC)*, Kos, Greece, July 2007a.

T. Coen, B. Missotten, and J. De Baerdemaeker. Feed rate control on a combine harvester. In *Proceedings of the International Conference on Agricultural Engineering*, Hannover, Germany, Nov. 2007b.

T. Coen, J. Anthonis, and J. De Baerdemaeker. Cruise control on a combine harvester using model-based predictive control with constraints. *Computers and Electronics in Agriculture*, 63(2):227–236, 2008a.

T. Coen, W. Saeys, B. Missotten, and J. De Baerdemaeker. Cruise control on a combine harvester using model-based predictive control. *Biosystems Engineering*, 99 (1):47–55, 2008b.

T. Coen, A. Vanrenterghem, W. Saeys, and J. De Baerdemaeker. Autopilot for a combine harvester. *Computers and Electronics in Agriculture*, 63(1):57–64, 2008c.

T. Coen, W. Saeys, and J. De Baerdemaeker. Capacity control on a combine harvester using model-based predictive control. *Biosystems Engineering*, 2009. submitted.

P. Couchman, M. Cannon, and B. Kouvaritakis. Mpc for stochastic systems. In *International Workshop on Assessment and Future Directions of NMPC*, Freudenstadt-Lauterbad, Germany, 2005.

P. Couchman, M. Cannon, and B. Kouvaritakis. Stochastic mpc with inequality stability constraints. *Automatica*, 42(12):2169–2174, 2006.

CP. Landbouwgewassen in 2007 fors duurder door china en biobranstoffen. De Tijd, 2008. January 3.

G. Craessaerts. Multivariate identification and control of the cleaning process on combine harvesters. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 2008.

G. Craessaerts, K. Maertens, and J. De Baerdemaeker. A windows-based design environment for combine automation via canbus. *Computers and Electronics in Agriculture*, 49(2):233–245, 2005.

G. Craessaerts, W. Saeys, B. Missotten, and J. De Baerdemaeker. A genetic input selection methodology for identification of the cleaning process on a combine harvester, part i: Selection of relevant input variables for identification of the sieve losses. *Biosystems Engineering*, 98(2):166–175, 2007a.

G. Craessaerts, W. Saeys, B. Missotten, and J. De Baerdemaeker. A genetic input selection methodology for identification of the cleaning process on a combine harvester, part ii: Selection of relevant input variables for identification of material other than grain (mog) content in the grain bin. *Biosystems Engineering*, 98(3): 297–303, 2007b.

G. Craessaerts, B. Missotten, and J. De Baerdemaeker. A biomass cleaner: improvements in crop harvesting machines and related methods. European Patent Application, 2008.

A. Craig. Speed control for a self propelled working machine. European Patent Application EP 1 680 952 A1, 2005.

C. Cutler and B. Ramaker. Dynamic matrix control - a computer control algorithm. In *Proceedings of the Joint American Control Conference*, San Francisco, USA, 1980.

J. Daintith and D. Gjertsen. *Oxford Dictionary of Scientists*. Oxford University Press, USA, 1999.

DDW. Biobrandstoffen wegen amper op voedselprijzen. De Tijd, 2008. April 26.

Y. Delepeleire. Biobrandstoffen catastrofaal voor het klimaat. De Standaard, 2008. Wednesday, April 30.

K. Deprez, D. Moshou, J. Anthonis, J. De Baerdemaeker, and H. Ramon. Improvement of vibrational comfort on agricultural vehicles by passive and semi-active cabin suspension. *Computers and Electronics in Agriculture*, 49(13):431–440, 2005.

N. Diekhans. Device for and a method of a throughput measurement in agricultural machines. United States Patent US 5,795,221, 1998.

N. Diekhans. Agricultural machine with a guidance system. United States Patent Application US2003/0145571 A1, 2003.

N. Diekhans and W. Behnke. Lost-grain detector for harvesting machine. United States Patent US 4,902,264, 1990.

J. Durham. Feedback control. http://digitalcommons.carleton.edu/pacp/5, 2004.

C. Edwards and S. Spurgeon. *Sliding mode control: theory and application*. Taylor & Francis, UK, 1998.

M. Eimer. Automatic control of combine threshing process. In *Proceedings IFAC Symposium on Automatic Control for Agriculture*, Saskatoon, Canada, June 1974.

W. Fechner, W. Busher, and J. Gersonde. Measuring device for measuring harvested crop throughput. United States Patent Application US 2005/0026662 A1, 2005.

H. J. Ferreau, P. Ortner, P. Langthaler, L. del Re, and M. Diehl. Predictive control of a real-world diesel engine using an extended online active set strategy. *Annual Reviews in Control*, 31(2):293–301, 2007.

G. Franklin, J. Powell, and A. Emami-Naeini. *Feedback control of dynamic systems*. Addison-Wesley Publishing, USA, third edition, 1994.

G. Franklin, J. Powell, and A. Emami-Naeini. *Feedback control of dynamic systems*. Pearson, USA, fifth edition, 2006.

S. Friedberg, A. Insel, and L. Spence. *Linear Algebra*. Prentice Hall, UK, 1989.

C. Garcia and M. Morari. Internal model control - part i. *Industrial Engineering Chemical Process Design and Development*, 21:308–323, 1982.

C. Garcia, D. Prett, and M. Morari. Model predictive control: theory and practice-a survey. *Automatica*, 25(3), 1989.

E. Gasparetto. Conclusions and recommendations. In *Proceedings of the 16th Members' Meeting of the Club of Bologna*, Bologna, Italy, Nov. 2005.

R. Gerhards, M. Sokefeld, C. Timmerman, and W. Kuhbauch. Site-specific weed control in maize, sugar, beet, winter wheat, and winter barley. *Precision Agriculture*, 3:25–35, 2002.

I. Ghijs. 250 kilo graan voor een volle tank brandstof. De Standaard, 2008. Saturday, January 19.

N. Hemming and J. Chaplin. Determining lag time for mass flow in a combine harvester. *Transaction of the ASAE*, 48:823–829, 2005.

T. Herlitzius and H. Boesch. Possible increase of combine efficiency by using the feedrate-ground speed control on John Deere combines STS, CTS and WTS. In *Proceedings of AgEng2004*, Leuven, Belgium, Sept. 2004.

G. Hevia, M. Mendez, and D. Buschiazzo. Tillage affects soil aggregation parameters linked with wind erosion. *Geoderma*, 140:90–96, 2007.

X. Huang, M. Caron, and D. Hindson. Recursive gram-schmidt orthonormalization procedure and its application to communiations. In *Proceedings of the Third IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications*, Taoyuan, Taiwan, 2001.

M. Jelali and A. Kroll. *Hydraulic Servo-systems: Modelling, Identification and Control*. Springer-Verlag, Berlin, Germany, 2003.

L. Jensen. An adaptive feedrate controller for combine harvesters. In *Proceedings International Symposium Modelling Identification and Control (IASTED)*, Grindelwald, Switzerland, Feb. 1988.

S. Karacan. Application of a non-linear long range predictive control to a packed distillation column. *Chemical Engineering and Processing*, 42:943–953, 2003.

R. Kawathekar and J. Riggs. Nonlinear model predictive control of a reactive distillation column. *Control Engineering Practice*, 15(2):231–239, 2007.

S. Keerthi and E. Gilbert. Optimal, infinite horizon feedback laws for a general class of constrained discrete time systems: stability and moving-horizon approximations. *Journal of Optimization Theory and Application*, 57:265–293, 1988.

D. Kendrick. Stochastic control for economic models: past, present and the paths ahead. *Journal of Economic Dynamics & Control*, 29:3–30, 2005.

T. Keviczky and G. Balas. Receding horizon control of an F-16 aircraft: a comparative study. *Control Engineering Practice*, 14:1023–1033, 2006.

M. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.

B. Kouvaritakis, M. Cannon, and V. Tsachouridis. Recent developments in stochastic mpc and sustainable development. *Annual Reviews in Control*, 28:23–35, 2004.

B. Kouvaritakis, M. Cannon, and P. Couchman. Mpc as a tool for sustainable development integrated policy assessment. *IEEE Transactions on Automatic Control*, 51 (1):145–149, 2006.

N. Kouvaritakis. The ispa meta model. Model developed under European Commission project EEN2-CT1999-00003, 2000a.

N. Kouvaritakis. Methodologies for integrating impact assessment in the field of sustainable development (minima-sud). Technical report, European Commission project EVG1-CT2002-00082, 2000b.

H. Kutzbach. Trends in power and machinery. *J. agric. Engng Res.*, 76:237–247, 2000.

H. Kutzbach. Contractors: new machines to increase efficiency of the various field operations. a critical analysis. In *Proceedings of the 8th Members' Meeting of the Club of Bologna*, Bologna, Italy, Oct. 1997.

K. Levien. Studies in the design and control of coupled distillation columns. PhD thesis, University of Wisconsin, Madison, USA, 1985.

K. Levien and M. Morari. Internal model control of coupled distillation columns. *AIChE journal*, 33:483–88, 1987.

S. Li. Agricultural mechanization promotion in china: current situation and future. In *Proceedings of the 15th Members' Meeting of the Club of Bologna*, Bologna, Italy, Nov. 2004.

G. Lopes, P. Magalhaes, and E. Nobrega. Optimal header height control system for combine harvesters. *Biosystems Engineering*, 81(3):262–272, 2002.

J. M. Maciejowski. *Predictive Control with Constraints*. Pearson Education, UK, first edition, 2002.

K. Maertens. Data-driven techniques for the on-the-go evaluation of separation processes in combine harvesters. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 2004.

K. Maertens and J. De Baerdemaeker. Flow rate based prediction of threshing process in combine harvesters. *Appl. Eng. Agric.*, 19(4):383–388, 2003.

J. Maxwell. On governors. In *Proceedings of the Royal Scoiety, No. 100*, London, UK, 1868.

D. Mayne and H. Michalska. Receding horizon control of non-linear systems. *IEEE Transactions on Automatic Control*, 35(5):814–824, 1990.

D. Mayne, J. Rawlings, C. Rao, and P. Scokaert. Constrained model predictive control: stability and optimality. *Automatica*, 36:789–814, 2000.

O. Mayr. *The origins of feedback control*. MIT Press, USA, 1970.

P. De Castro. The quality of productions. market needs. institutional and prescriptive aspects in the eu. In *Proceedings of the 14th Members' Meeting of the Club of Bologna*, Chicago, USA, July 2002.

A. De Greef. Amerikaanse winkels rantsoeneren rijst. De Standaard, 2008. Friday, April 25.

W. De Preter. Brood en pasta blijven nog duur. De Standaard, 2008. Tuesday, February 26.

J. De Temmerman, K. Deprez, I. Hostens, J. Anthonis, and H. Ramon. Conceptual cab suspension system for a self-propelled agricultural machine - part 2: operator comfort optimisation. *Biosystems Engineering*, 90(3):271–278, 2005.

European Commission. Emissions for non-road mobile machinery. Directive 2004/26/EC, 2004.

European Commission. Noise emissions for outdoor equipment. Directive 2005/88/EC, 2005.

European Commission. Machinery. Directive 2006/42/EC, 2006.

J. Marques da Silva, J. Soares, and D. Karlen. Implement and soil condition effects on tillage-induced erosion. *Soil and tillage research*, 78:207–216, 2004.

D. Van Hessem and O. Bosgra. Closed-loop stochastic dynamic process optimization under input and state constraints. In *Proceedings of the American Control Conference*, Anchorage, AK, USA, 2002.

D. Van Hessem and O. Bosgra. Stochastic closed-loop model predictive control of continuous nonlinear chemical process. *Journal of Process Control*, 16(3):225–241, 2006.

J. Van Loo. Measuring the moisture content of straw continuously. In *Proceedings of the First International Grain and Forage Harvesting Conference*, Iowa, USA, 1978.

E. Meadows, M. Henson, J. Eaton, and J. Rawlings. Receding horizon control and discontinuous state feedback stabilization. *International Journal of Control*, 62: 1217–1229, 1995.

H. Michalska and D. Mayne. Robust receding horizon control of constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 38(11):1623–1632, 1993.

B. Missotten. Measurement systems for the mapping and the evaluation of crop production performance. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 1998.

B. Missotten and B. Busschaert. Mass flow measuring device for a combine harvester. European Patent Application EP 1 380 203 A1, 2003.

B. Missotten, L. Claessens, K. Maertens, P. Jancsok, and C. Wallays. Apparatus and method for analysing the composition of crop in a crop-conveying machine. International Patent WO 2006/010761 A1, 2006.

R. Mooijman. Chinese consument en amerikaanse chauffeur maken voeding duurder. De Standaard, 2007a. Thursday, December 13.

R. Mooijman. De juiste prijs. De Standaard, 2007b. Wednesday, December 19.

R. Mooijman. Explosie tarweprijs maakt brood duurder. De Standaard, 2008. Saturday, July 14.

M. Morari and J. Lee. Model predictive control: past, present and future. *Computers and Chemical Engineering*, 23, 1998.

N. Murakami, A. Ito, J. Will, M. Steffen, K. Inoue, K. Kita, and S. Miyaura. Development of a teleoperation system for agricultural vehicles. *Computers and Electronics in Agriculture*, 63(1):81–88, 2008.

K. Narendra and J. Balakrishnan. Adaptive control using multiple models. *IEEE Transactions on Automatic Control*, 42(2):171–187, 1997.

O. Nelles. *Nonlinear System Identification*. Springer-Verlag, Berlin, Germany, 2001.

D. Nesic. Controllability for a class of simple wiener-hammerstein systems. *System and Control letters*, 36(1):51–59, 1999.

J. Neter, M. Kutner, C. Nachtsheim, and W. Wasserman. *Applied Linear Statistical Models*. McGraw-Hill, fourth edition, 1996.

H. Nguyen and G. Rogers. *Fundamentals of Mathematical Statistics*. Springer, 1989.

J. Nocedal and S. Wright. *Numerical Optimization*. Springer, USA, 1989.

H. Pangels, T. Pilarski, K. Fitzpatrick, M. Happold, R. Hoffman, and W. Whittaker. Agricultural harvester with robotic control. International Patent WO 98/46065, 1998.

T. Parisini and R. Zoppoli. A receding horizon regulator for nonlinear systems and a neural approximation. *Automatica*, 31(10):1443–1451, 1995.

K. M. Passino and S. Yurkovich. *Fuzzy control*. Addison Wesly Longman, USA, 1998.

R. Pintelon and J. Schoukens. *System identification, a frequency domain approach*. IEEE Press, USA, 2001.

B. Pluymers. Robust model based predictive control - an invariant set approach-. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 2006.

D. Prett and R. Gilette. Optimization and constrained multivariable control of a catalyc craacking unit. In *Proceedings of the joint automatic control conference*, 1980.

J. Primbs and V. Nevistic. Feasibility and stability of constrained finite receding horizon control. *Automatica*, 36(7):965–971, 2000.

A. Propoi. Use of lp methods for synthesizing sampled-data automatic systems. *Automatic Remote Control*, 24, 1963.

S. Qin and T. Badgwell. An overview of nonlinear model predictive control applications. In *IFAC Workshop on Nonlinear Model Predictive Control. Assessment and Future Directions.*, Ascona, Switzerland, 1998.

S. Qin and T. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11:733–764, 2003.

G. Quick. Grain harvesting 1776 - a bicentennial review. ASAE Meeting Paper Number 76-1555, 1976.

G. Quick and W. Buchele. *The grain harvesters*. ASAE, USA, first edition, 1978.

C. Rao, S. Wright, and J. Rawlings. Application of interior-point methods to model predictive control. *Journal of Optimization Theory and Applications*, 99(3):723–757, 1998.

J. Rawlings and K. Muske. Stability of constrained receding horizon control. *IEEE Transactions on Automatic Control*, 38(10):1512–1516, 1993.

P. Reyns. Continuous measurement of grain and forage quality during harvest. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 2002.

J. Richalet, A. Rault, J. Testud, and J. Papon. Model predictive heuristic control: applications to industrial processes. *Automatica*, 14:413–428, 1978.

RMG. Minder, maar grotere boerderijen. De Standaard, 2007. Saturday, July 28.

R. Rouhani and R. Mehra. Model algorithmic control (mac): basic theoretical properties. *Automatica*, 18(4):401–414, 1982.

L. Santos, P. Alfonso, J. Castro, N. Oliveira, and L. Biegler. On-line implementation of nonlinear mpc: an experimental case study. *Control Engineering Practice*, 9(8): 847–857, 2001.

Y. Sarig. Traceability of food products. In *Proceedings of the 14th Members' Meeting of the Club of Bologna*, Bologna, Italy, Nov. 2003.

P. Scokaert and J. Rawlings. Constrained linear quadratic regulation. *IEEE Transactions on Automatic Control*, 43(8):1163–1169, 1998.

SDW. 260 000 ha nodig om biobrandstofdoelstelling te halen. De Standaard, 2006. Thursday, November 9.

A. Segre. Eu enlargement and its influence on agriculture and mechanization. In *Proceedings of the 15th Members' Meeting of the Club of Bologna*, Bologna, Italy, Nov. 2004.

L. Spokas, D. Steponavicius, and P. Wacker. Steuerung des dreschprozesses durch das niveau der Kornbeschädigung. In *Proceedings of VDI Mähdrescher Kolloquium*, Dresden, Germany, Mar. 2007.

A. Srivastava, C. Goering, R. Rohrbach, and D. Buckmaster. *Engineering principles of Agricultural Machines*. ASABE, USA, second edition, 2006.

STM. Grotere landbouwbedrijven tegen 2016. De Standaard, 2006. Tuesday, December 5.

S. Strogatz. *Nonlinear dynamics and chaos*. Westview Press, USA, 2000.

G. Strubbe. Method and apparatus for measuring combine grain loss. United States Patent US 5,015,997, 1991.

G. Strubbe and B. Missotten. Harvester with crop flow rate sensor. European Patent Application EP 0 958 729 A1, 1999.

V. Utkin. *Sliding modes in control optimization*. Springer-Verlag, Germany, 1992.

A. Vahidi, A. Stefanopoulou, and H. Peng. Recursive least squares with forgetting for online estimation of vehicle mass and road grade: Theory and experiments. *Vehicle System Dynamics*, 43:31–55, 2005.

R. Vansichen and J. De Baerdemaeker. Signal processing and system dynamics for continuous yield measurements on a combine. In *Trends in agricultural engineering*, Prague, Czech Republic, 1992.

VDB. Minder boeren investeren weer fors in meer landbouwmachines. De Tijd, 2007. December 4.

K. Verhoeven. Boeren wordt weer leuk. De Standaard, 2007. Saturday, August 25.

A. Vivas and P. Poignet. Predictive functional control of a parallel robot. *Control Engineering Practice*, 13(7):863–874, 2005.

E. Vranken, R. Gevers, J.-M. Aerts, and D. Berckmans. Performance of model-based predictive control of the ventilation rate with axial fans. *Biosystems Engineering*, 91(1):87–98, 2005.

C. Wallays. Monitoring of mog on combine harvesters. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 2008.

C. Wallays and B. Missotten. A method an apparatus for detecting errors in electronically processed images. British patent application GB 07 17986 4, 2008a.

C. Wallays and B. Missotten. Crop particle discrimination methods and apparatus. British patent application (CNH reference MD-367), 2008b.

J. Watton. *Fluid Power Systems*. Prentice Hall, England, 1989.

J. Wilson. Guidance of agricultural vehicles - a historical perspective. *Computers and Electronics in Agriculture*, 25:3–9, 2000.

WLE. Tarwe was nog nooit duurder. De Standaard, 2008. Tuesday, February 26.

L. Zadeh and B. Whalen. On optimal control and linear programming. *IRE Transactions on Automatic Control*, 7(4):45, 1962.

Online references

Agritechnica. http://www.agritechnica.com, 2007. The largest fair in the world for agricultural machinery; Accessed on December 15th, 2008.

A. Bemporad. http://www.dii.unisi.it/~bemporad/, 2008. The site of prof. Alberto Bemporad; Accessed on September 9th, 2008.

Britannica. http://www.britannica.com/EBchecked/topic/135480/control-system, 2008. A definition of control system; Accessed on October 4th, 2008.

Claas. http://www.claas.com, 2008. Agricultural machinery manufacturer; Accessed on August 23rd, 2008.

dSpace. http://www.dspaceinc.com, 2008. Rapid prototyping system; Accessed on May 15th, 2008.

Farmphoto. http://www.farmphoto.com, 2008. Digital photography for farmers; Accessed on August 11th, 2008.

Harvestec. http://www.harvestec.com, 2008. Combine harvester header manufacturer; Accessed on August 23rd, 2008.

IWT. http://www.iwt.be, 2008. Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen); Accessed on June 3rd, 2008.

Case New Holland. http://www.cnh.com, 2008. Agricultural machinery manufacturer; Accessed on August 23rd, 2008.

Club of Bologna. http://www.clubofbologna.org, 2008. A world taskforce on the strategies for the development of agricultural mechanisation (UNACOMA think-thank); Accessed on July 13th, 2008.

CNH Global. http://www.cnh.com, 2008a. Investor relations, report of Q4 2007; Accessed on August 4th, 2008.

CNH Global. http://www.cnh.com, 2008b. Investor relations, report of Q1 2008; Accessed on August 4th, 2008.

De Standaard. http://www.standaard.be, 2008. Articles are available in the archive with a subscription; Accessed on August 27th, 2008.

De Tijd. http://www.tijd.be, 2008. Articles are available in the archive with a subscription; Accessed on August 27th, 2008.

FOD Economie. http://mineco.fgov.be/informations/indexes/indint1xls_inflatie3_nl.htm, 2008a. Inflation figures; Accessed on July 28th, 2008.

FOD Economie. http://www.statbel.fgov.be/port/agr_nl.asp, 2008b. Het Landbouwportaal; Accessed on August 7th, 2008.

John Deere. http://www.deere.com, 2008. Agricultural machinery manufacturer; Accessed on August 23rd, 2008.

National Agricultural Statistics Service. http://www.nass.usda.gov/Statistics_by_State/New_York/Historical_Data/PricesReceived/WheatPrices.xls, 2008. Historical wheat price; Accessed on October 4th, 2008.

National Instruments. http://www.ni.com, 2008. LabView, and industrial measurement equipment; Accessed on May 15th, 2008.

New Holland Austria. http://www.newholland.com, 2008. Select the Austrian version of the site; Accessed on August 27th, 2008.

New Holland Benelux. http://www.newholland.com, 2008. Select the Benelux version of the site; Accessed on August 24th, 2008.

Shelbourne Reynolds. http://www.shelbourne.com, 2008. Combine harvester header manufacturer; Accessed on August 23rd, 2008.

The Mathworks. http://www.mathworks.com, 2008. Developer of Matlab; Accessed on May 15th, 2008.

US Department of Labor, Bureau of Labor Statistics. ftp://ftp.bls.gov/pub/special.requests/cpi/cpiai.txt, 2008. Consumer Price Index; Accessed on October 4th, 2008.

Northstar. http://www.cornheader.com, 2008. Combine harvester header manufacturer; Accessed on August 23rd, 2008.

Softing. http://www.softing.com, 2008. producer of CANcards; Accessed on July 17th, 2008.

Trimble. http://www.trimble.com, 2008. manufacturer of GPS systems; Accessed on October 26th, 2008.

VILT. http://www.vilt.be/feitenencijfers/akkerbouwsector.html, 2008. Vlaams Informatiecentrum over Land- en Tuinbouw; Accessed on October 5th, 2008.

Wikipedia. http://en.wikipedia.org/wiki/Centrifugal_Governor, 2008a. A definition and picture of the centrifugal governor; Accessed on July 30th, 2008.

Wikipedia. http://en.wikipedia.org/wiki/Image:US_Historical_Inflation.svg, 2008b. A graph and dataset of the historical inflation in the 20th Century in the US; Accessed on August 29th, 2008.

Curriculum Vitae

There is only one thing in the world worse than being talked about,
and that is not being talked about.
-Oscar Wilde-

Tom Coen was born on February 12th 1981 in Mechelen, Belgium. He went to high school in the Onze-Lieve-Vrouwecollege in Vilvoorde, and graduated in 1999 in the option Latin-Mathematics. In 2004 he graduated as a mechanical-electrotechnical engineer, option datamining and automation at the Katholieke Universiteit Leuven (Catholic University of Leuven).

From July 2004 till November 2008 he was a research engineer at the laboratory for Agricultural Machinery (later called MeBioS, division of Mechatronics, Biosensors and Statistics) at the Faculty of Bioscience Engineering at the university of Leuven. Starting from January 2005 he held a PhD scholarship from the IWT (Institute for the Promotion of Innovation through Science and Technology — IWT Vlaanderen, W, IWT (2008)) entitled: 'Onderzoek naar een foutbestendige regelaar voor een tijdsvariabel niet-lineair proces toegepast op de rijsnelheidsregeling voor maaidorsers' ('Research towards a fault-tolerant controller for a time-varying nonlinear process applied to combine harvesters'). This research was done in close collaboration with CNH Belgium NV. IntelliCruise, the feed rate control system he designed, received a silver medal at the 2007 Agritechnica (Hannover, Germany) (W, Agritechnica (2007)). He also holds a patent for this implementation.

Publication list

Writing is no trouble:
you just jot down ideas as they occur to you.
The jotting is simplicity itself
- it is the occurring which is difficult.
-Stephen Leacock-

Journal papers

T. Coen, W. Saeys, H. Ramon, and J. De Baerdemaeker. Optimizing the tuning parameters of least squares support vector machines regression for NIR spectra. *Journal of Chemometrics*, 20(5):184-192,2006.

T. Coen, W. Saeys, B. Missotten, and J. De Baerdemaeker. Cruise control on a combine harvester using model-based predictive control. *Biosystems Engineering*, 99(1):47-55,2008.

T. Coen, J. Anthonis, and J. De Baerdemaeker. Cruise control using model-based predictive control with constraints. *Computers and Electronics in Agriculture*, 63(2):227-236,2008.

T. Coen, A. Vanrenterghem, W. Saeys, and J. De Baerdemaeker. Autopilot for a combine harvester. *Computers and Electronics in Agriculture*,63(1):57-64,2008.

T. Coen, W. Saeys, and J. De Baerdemaeker. Capacity Control on a Combine Harvester using Model-based Predictive Control. *submitted to Biosystems Engineering*, 2009.

Book chapters

W. Saeys, T. Coen and H. Ramon. Biosystems Analysis and Design. In *Biosystems Engineering*, McGraw-Hill, submitted 2008.

Conference papers

T. Coen, N. Jans, P. Van de Ponseele, I. Goethals, J. De Baerdemaeker, and B. De Moor. Modeling the relationship between human perception and sound quality parameters using LS-SVMs. In *Proceedings of the International Conference on Noise & Vibration Engineering (ISMA), Leuven, Belgium*, 2004.

T. Coen, N. Jans, P. Van de Ponseele, I. Goethals, J. De Baerdemaeker, and B. De Moor. Engine sound comfortability: relevant sound quality parameters and classification. In *Proceedings of the 16th IFAC World Congress, Prague, Czech Republic*, 2005.

G. Craessaerts, T. Coen, and J. De Baerdemaeker. Identification of sensor faults on combine harvesters using intelligent methods. In *Proceedings of the 16th IFAC World Congress, Prague, Czech Republic*, 2005.

J. Paduart, J. Schoukens, R. Pintelon and T. Coen. Nonlinear state space modeling of multivariable systems. In *Proceedings of the 14th IFAC Symposium on System Identification (SysID), Newcastle, Australia*, 2006.

T. Coen, J. Paduart, J. Anthonis, J. Schoukens, and J. De Baerdemaeker. Nonlinear system identification on a combine harvester. In *Proceedings of the American Control Conference (ACC), Mineapolis, USA*, 2006.

A. Vanrenterghem, T.Coen, P. Calmeyn, and J. De Baerdemaeker. Autopilot for a harvester. In *Proceedings of the IFAC workshop on Bio-Robotics, Information Technology and Intelligent Control for Bioproduction Systems, Sapporo, Japan*, 2006.

T. Coen, and J. De Baerdemaeker. Nonlinear Model-based Predictive Control for cruise control. In *Proceedings of the VDI Mähdrescher Kolloquium, Dresden, Germany*, 2007.

T. Coen, J. Anthonis, and J. De Baerdemaeker. Nonlinear Model-based Predictive Control for cruise control on combine harvesters. In *Proceedings of the European Control Conference (ECC), Kos, Greece*, 2007.

T. Coen, B. Missotten, and J. De Baerdemaeker. Feed rate control on a combine harvester. In *Proceedings of the International Conference on Agricultural Engineering, Hannover, Germany*, 2007.

T. Coen, W. Saeys and J. De Baerdemaeker. Model-based predictive control for capacity control on a combine harvester. In *Proceedings of the CIGR International Conference of Agricultural Engineering XXXVII Congresso Brasileiro de Engenharia Agricola, Foz do Iguacu, Brasil*, 2008.



Figure 12.1: The presentation of the silver medal winner IntelliCruise at Agritechnica (W, New Holland Benelux (2008); W, Agritechnica (2007)).

Awards

Silver Medal Winner (prize for innovating an existing feature) at the 2007 Agritechnica with the CNH Feed rate Control System.

Patents

K. Maertens, B. Missotten, D. Depreitere, T. Coen. Apparatus and method for controlling the speed of a combine harvester. GBA 0721909, 2007.

Other publications

T. Coen, G. Craessaerts, and J. De Baerdemaeker. Testing control strategies on a combine harvester with NI LabView Real-Time, National Instruments Application Paper, 2006.

T. Coen, and J. De Baerdemaeker. Stochastic MPC. In *Proceedings of NI Week 2006, Austin, USA*, 2006.

Appendix A

Measurement platform

In this appendix the measurement and control platform used throughout all the experiments in this research is described. The platform was selected specifically for this project. First of all the specifications are listed. In a following section the different alternatives are listed. Finally the hardware actually used for the tests is described.

A.1 Requirements

The purpose of the platform is to measure all signals on the machine, as well as to solve a MPC problem online, and apply the calculated value to the machine. A normal control rate for these applications varies between 5 Hz and 20 Hz. Thus, the platform needs to satisfy the following requirements:

- CAN connection (most of the measurements on the combine are available through CANbus).
- analog data acquisition at sampling rates up to 100kHz (for related projects on the combine).
- real-time operating system to guarantee timely execution (very important with CAN and control applications).
- significant processing power to solve the MPC problem.
- easy and rapid programming of the real-time system.

Based on these specifications different alternatives were examined, which are discussed in the next section.

A.2 Considered hardware/software

In general the hardware selection also determines which software will be used to program the hardware. Following options where considered:

- Matlab with a real-time target (W, The Mathworks (2008))



Figure A.1: A PXI chassis from National Instruments (W, National Instruments (2008)).

- Matlab with dSpace (W, dSpace (2008))
- LabView with National Instruments hardware (PXI) (W, National Instruments (2008))

Matlab has the advantage that it is very flexible, and that a lot of toolboxes are already available for this platform. Moreover, most engineers are experienced users. However, real-time operation is only possible by compiling a Simulink diagram. Most toolboxes were not supported for real-time at the time of the start of this project.

LabView is less known. It is a graphical programming language (unlike Matlab). The core business of National Instruments is industrial measurement and control equipment.

A.3 Selected hardware

Because of the track record of National Instruments concerning hardware and data acquisition, and since most Matlab toolboxes cannot be used anyway, LabView in combination with National Instruments hardware is selected for the experiments. A rugged industrial PC (PXI) is acquired, with following specification:

chassis: PXI-1031 4 slot chassis

processor: PXI-8196 Pentium M 2.0GHz controller with real-time embedded software and 512 MB RAM

CAN connection: PXI-8461 high speed CAN interface

data acquisition: PXI-6221 M series DAQ (16 analog inputs, 24 digital I/O, 2 analog outputs)

More information can be found on the web (W, National Instruments (2008)). A PXI chassis is shown in Fig. A.1.

A.4 Benchmarking

In this section the real time platform is compared to a standard Windows computer to give an idea of the processing power and overall performance. The specifications of the Windows computer which is used for this comparison are:

- Intel T2500 2GHz CPU (dual core)
- 1GB RAM
- Windows XP Professional

Three comparison scenarios are considered, namely

scenario 1: multiply a number to itself in a loop.

scenario 2: identical to the previous situation, but with two loops in parallel

scenario 3: instead of two loops in parallel, two numbers are squared in the same loop

The graphical LabView code of these scenarios is shown in Fig. A.2. The loop is executed multiple times to remove the effect of starting and initializing the program. Based on this program, the number of operations per second that can be executed by both computers can be estimated. The results are summarized in Table A.1.

Note that the RT system significantly outperforms the Windows system. Also note that the Windows system contains much more overhead. The number of operations/second increases considerably if the number of loop executions is increased.

Both systems can perform more ops/s if two operations are performed per loop iteration, since in this case there is less influence of the loop overhead. Note that the Windows system does not cope very well with the two identical loops case. Probably this is caused by bad management of the dual core processor. Both loops can after all be attributed to different cores.

A.5 Conclusion

For all the experiments described in this research a National Instruments PXI system with LabView RT was used. This implies that all the software needs to be written in LabView. The structure of the designed software is discussed in Appendix B.

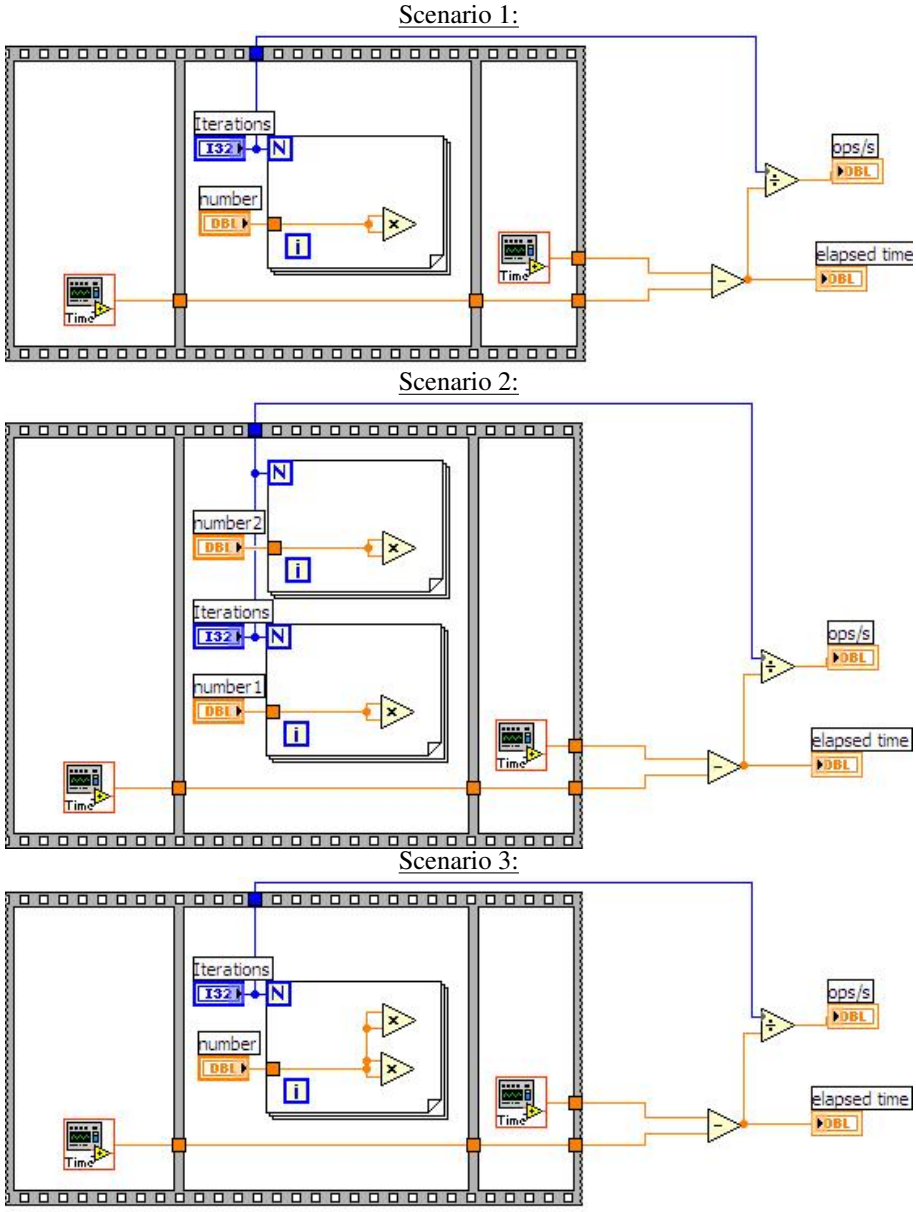


Figure A.2: Test scenarios in LabView Code.

Table A.1: Benchmarking the performance of LabView on Windows and a RT platform. The number of operations per second (ops/s) each platform can perform is shown below for the three described scenarios. Note that in some cases the number of ops/s significantly increases if more executions of the loop are performed.

	ops/s					
loop	1 loop		2 identical loops		2ops/loop	
executions	Windows	RT	Windows	RT	Windows	RT
1.00E+06	32.3E+06	76.9E+06	16.0E+06	76.9E+06	43.5E+06	95.2E+06
1.00E+07	49.0E+06	75.8E+06	11.6E+06	77.2E+06	63.9E+06	93.9E+06
1.00E+08	67.4E+06	75.6E+06	11.8E+06	77.1E+06	83.1E+06	93.6E+06
1.00E+09	70.5E+06	75.6E+06	11.6E+06	77.1E+06	85.9E+06	93.6E+06

Appendix B

Software

The purpose of this appendix is to discuss the overall structure of the designed software. The optimization algorithm which is used to solve MPC problems as well as its implementation is described separately in Chapter 9.

B.1 Available software at the start

At the start of this project a program called *CANbus Interface* was already available. This program had been written in C++ by Koen Maertens during his PhD (Maertens, 2004) (Fig. B.1). It made use of a Softing CAN card (W, Softing (2008)), plugged into the PCMCIA slot, to connect to the CANbus of the combine harvester. This interface could then be used to measure and log the CAN signals, and to control the machine.

However, since this program ran on Windows, accurate timing could only be guaranteed up to 5 Hz sampling rate. Moreover, it is very unlikely that a 5 Hz sampling rate had been possible if MPC calculations had to be done in the program. Implementation of MPC in C++ code is also a very tedious task.

Thus a new software and hardware platform was needed. As mentioned in the previous chapter, LabView and National Instruments hardware is selected for this task. The requirements of the software are listed in the next section.

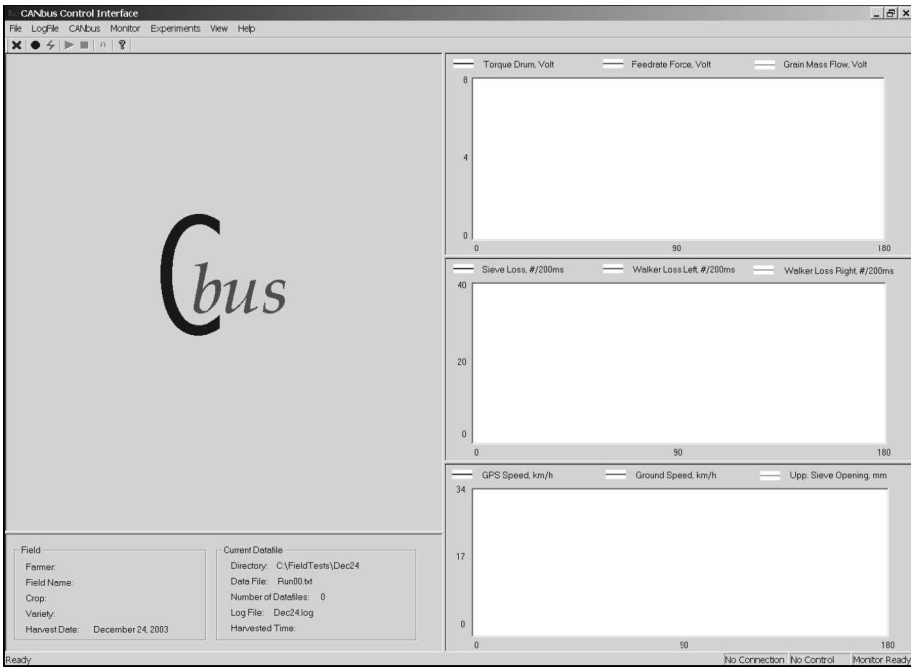


Figure B.1: The interface of *CANbus Interface* (Maertens, 2004).

B.2 Requirements

The software requirements are:

- log the raw data coming from CAN as well as from some extra sensors (through analog DAQ). Logging is needed at 5 Hz (for normal operation) and 20 Hz (for system identification).
- generate and apply excitations to the machine at 5 Hz and 20 Hz (steps, swept sine, random odd-odd multisine,...).
- visualize the current machine status online.
- calculate an online plant model (and log the model state at all times).
- test common controllers for capacity control (PID, fuzzy control).
- test MPC.
- log all the parameters of the controllers at initialization (for future reference).
- design a plant simulator to be able to debug the code, and to test different settings offline.

B.3 Software description

B.3.1 Overview

LabView software is programmed graphically, and thus has the unique capability to easily include parallel processing. In this program following parallel loops are created:

interface loop: this loop converts the operator actions (pressing a button,...) to a program state and also calculates and manages excitations.

control loop: this loop writes controller actions to CAN, reads data from CAN, and calculates the controller actions of the next time step.

display loop: this loop displays the measurement and control data on the user interface.

process loop: this loop calculates the adaptive model.

generate control matrices: this loop calculates the matrices needed for MPC control. This code is triggered if required (for example every x seconds the matrices are updated). It is put in a separate loop with low priority to make sure LabView calculates these matrices only if nothing else requires CPU power. This avoids that calculating the matrices interferes with normal operation.

control log loop: the logging of the control signals as well as the logging of the control parameters is done in this loop.

main log loop: the main logging of the raw data is performed in this loop.

For the cruise control application a similar program is built, which also allows to change the period of the main control loop to allow for 20 Hz control.

The different loops each have there own priority. This priority is used by LabView to determine which loop gets CPU time first if it is not possible to complete all actions within the available time frame. The loops (together with their sampling rate and their priority) in order of decreasing priority are listed below:

Loop	period (ms)	priority
control loop	200	1000
process loop	200	100
interface loop	300	50
display loop	200	30
control log loop	1000	10
main log loop	1000	10
generate control matrices	N/A	0

An overview of the interaction between the software components is illustrated in Fig. B.2. The arrows indicate data exchange. The line thickness gives an indication of the priority level of the component.

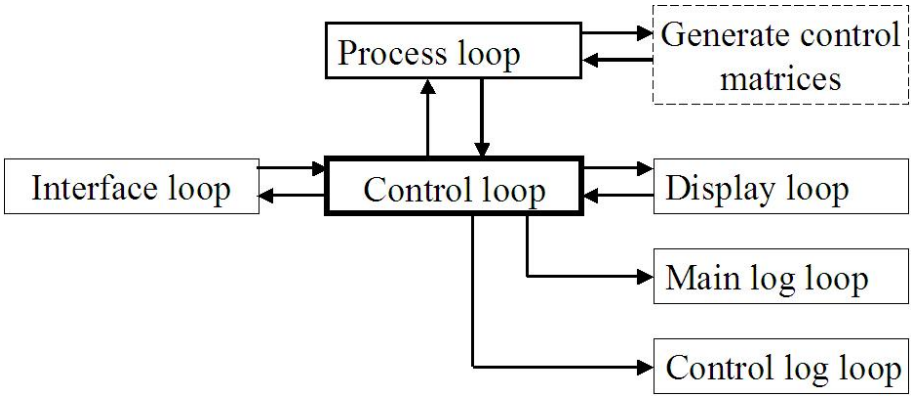


Figure B.2: Flow chart of the software, the line thickness indicates the priority level of the blocks. The arrows indicate the direction of the data exchange.

B.3.2 Control loop

The control loop performs the following steps in each period:

1. put the value which is calculated in the previous iteration on CAN.
2. read all the relevant values from CAN.
3. calculate the filtered values (low-pass second order Butterworth filter with a 1 Hz cut-off).
4. calculate the output value, or read the output value from a file. The control value can be calculated using a PID or fuzzy strategy, or using MPC. For cascaded controllers which consists of a supervisory controller (grain mass flow or losses), a feed rate controller and a speed controller, the output of this step is the feed rate/speed set point.
5. if needed, calculate the control value (fuzzy or PID strategy) based on the feed rate set point. The output of this step may be a hydrostatic flow, or a speed.
6. if needed, calculate the control value (fuzzy or PID strategy) based on the speed set point.

The section which reads from CAN can also be switched to simulation mode. The section which writes to CAN then writes to the input of the simulator.

B.3.3 Process loop

The process loop calculates the most recent values of the parameters of the adaptive model, as described in Chapter 6. Of course the model will only be updated if the combine is in the crop. The state ‘in crop’ is defined as:

- the header is down.



Figure B.3: The interface of the LabView program for capacity control.

- the header, and thus also the threshing mechanism, is engaged.
- the feed rate is larger than a given minimum value.

Moreover, each relation can be put on hold separately if either the input or the output variable are no longer within the predefined bounds.

B.3.4 Interface loop

For a computationally intensive application such as this one, it is very important to separate important tasks from less important ones. This is why the interface tasks need to be put in a separate - low priority - loop. The interface is shown in Fig. B.3. The interface loop contains two main parts, namely a parallel and a sequential part. The parallel part of the code contains functionality such as:

- calculate excitations with the parameters set by the user. The available excitations are steps, slope, sine, swept sine, random gaussian noise, random gaussian noise with increasing amplitude, random odd-odd multisine, blockwave, random uniform noise and random uniform noise with increasing amplitude.
- save calculated excitation sequences to file.
- load sequences from file. This also enables the user to use externally generated sequences.
- adapt the interface to the current user settings. For example the names of the parameters in the interface depend on the selected excitation.

The sequential part of this loop contains the application management, namely:

- start and stop control (and control logging).
- start and stop the main logging.
- open and close the CAN connection.
- start and stop the extra analog data acquisition.
- shut down the program in a correct manner (close CAN first,...).

There is a handshaking procedure between the interface loop and the relevant loop (control loop, logging loop) for these actions.

B.3.5 Display loop

The GUI consists of two kind of plots, dedicated plots and user configurable plots. For the user configurable plots the user can select the variable to plot from a pull-down menu. All these plotting actions are dealt with in the display loop.

B.3.6 Control log loop

This loop writes the parameter log, as well as the control logging. The parameter log links logging files to the control parameters which were set at that time. The control logging contains the control signals, as well as the filtered control inputs.

B.3.7 Main log loop

The main logging loop logs all the raw signals which are read into the program by CAN or analog data acquisition.

B.3.8 Generate control matrices

The matrices of the MPC problem depend on the system model. Since the model parameters are estimated online, the parameters of this model vary. Thus, the MPC matrices need to be updated regularly. Because this implies a significant computational load, it is very important to choose the right update rate. The matrices can for instance be updated every second or every 5 seconds, or even only if the model parameters vary more than a predefined threshold.

B.4 Conclusion

This program, LabView and the PXI system have shown to be a solid and stable tool to control the machine in a rough environment. The software often ran for up to 16 hours a day, without any problems worth mentioning.

The same methodology was used to develop a program to implement cruise control. This is a stripped version of this program, and it can also control the machine with MPC at 20 Hz. The interface is shown in Fig. B.4.

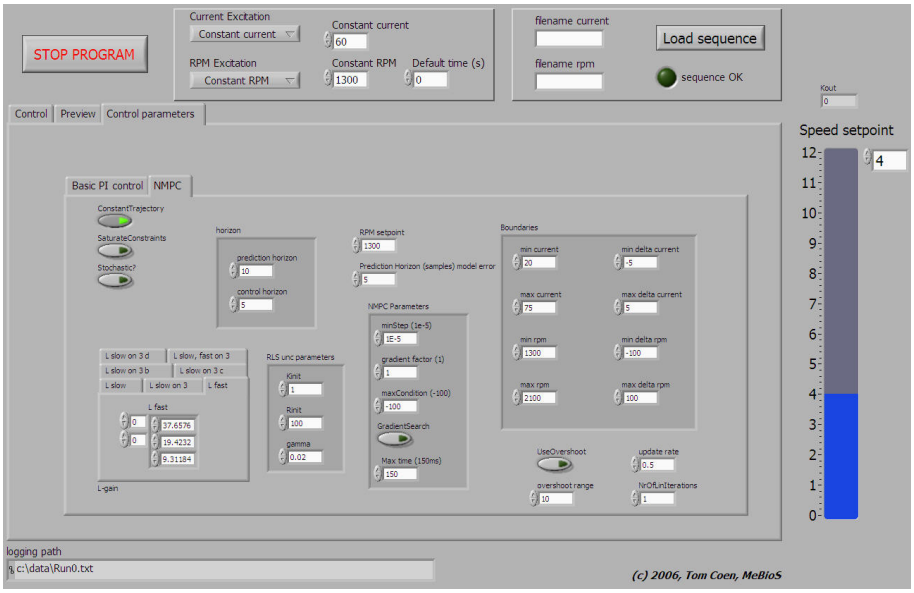


Figure B.4: The interface of the LabView program used for the implementation of cruise control.

Appendix C

Computational load of a model update

In this appendix the computational load which results from updating the model is analyzed. The results in this chapter led to the conclusion that the model update rate needs to be kept as low as possible in order to keep the controller computationally feasible.

C.1 Procedure

In each so-called model update the matrices of the optimization problem are updated. The optimization problem is written in terms of the input $\mathbf{U}$ and the initial state $\mathbf{x}_0$ (see Chapter 9), and can be written as:

$$\min_{\mathbf{U}} \left(\frac{1}{2} \mathbf{U}^T \mathbf{H} \mathbf{U} + \mathbf{f}^T \mathbf{U} \right) \tag{C.1}$$

subject to:

$$\mathbf{E}\mathbf{x}_0 + \mathbf{F}\mathbf{U} \quad \leq \quad \mathbf{b} \tag{C.2}$$

where $\mathbf{x}_0$ is the vector containing the initial condition of the system and $\mathbf{U}$ is a stacked vector containing all inputs $\mathbf{u}_1,\ldots,\mathbf{u}_{N_C-1}$ over the control horizon:

$$\mathbf{U} \quad = \quad \begin{bmatrix} \mathbf{u}_0 \\ \vdots \\ \mathbf{u}_{N_C-1} \end{bmatrix} \tag{C.3}$$

with $\mathbf{U} \in \mathbb{R}^{N_C p \times 1}$. The matrices $\mathbf{H}$, $\mathbf{E}$ and $\mathbf{F}$ as well as the vectors $\mathbf{f}$ and $\mathbf{b}$ need to be calculated from the system model. These matrices depend on the gains of the process model, and thus need to be updated in order to take new estimates for the gains into account. It is not possible to update these matrices without recalculating them since the structure of the matrices is too complex.

This appendix describes how the above mentioned matrices and vectors are calculated from the process gains and delays, given the model structure. The optimization of Eqn (C.1) and Eqn (C.2) also depends on the reference vector. In order to include this dependency, Eqn (C.1) is rewritten as follows:

$$\min_{\mathbf{U}} \left(\frac{1}{2} \mathbf{U}^T \mathbf{H} \mathbf{U} + \left(\mathbf{F_R} \mathbf{X}_{ref} + \mathbf{F_X} \mathbf{x}_0 \right)^T \mathbf{U} \right) \tag{C.4}$$

In other words:

$$\mathbf{f} \quad = \quad \mathbf{F_R} \mathbf{X}_{ref} + \mathbf{F_X} \mathbf{x}_0 \tag{C.5}$$

with $\mathbf{X}_{ref}$ the control reference and $\mathbf{x}_0$ the initial state. Note that $\mathbf{H} \in \mathbb{R}^{N_C p \times N_C p}$ with p the number of inputs and N_C the control horizon. $\mathbf{f} \in \mathbb{R}^{N_C p \times 1}$, $\mathbf{F_R} \in \mathbb{R}^{N_C p \times N_P q}$ and $\mathbf{F_X} \in \mathbb{R}^{N_C p \times q}$ with q the number of states. This means that by taking the control horizon N_C much smaller than the prediction N_P, the actual optimization problem remains of a relatively low dimension, even when the prediction horizon goes far into the future.

Each time the model is updated, the following steps need to be performed:

1. calculate the state space model. This depends on the most recent estimate of the gains.

2. build the matrices taking into account the prediction and control horizon.

3. multiply these large dimensional matrices to get the matrices defined in Eqn (C.4) and Eqn (C.2)

Step 1 is elaborately discussed in Chapter 6, and will be skipped here. Assume there is a state space model, which can be written as follows:

$$\mathbf{x}_{k+1} = \mathbf{A} \mathbf{x}_k + \mathbf{B} \mathbf{u}_k, \qquad \forall k \in [0, N_P - 1] \tag{C.6}$$

with $\mathbf{A} \in \mathbb{R}^{q \times q}$, $\mathbf{B} \in \mathbb{R}^{q \times p}$, $\mathbf{x}_k \in \mathbb{R}^{q \times 1}$ and $\mathbf{u}_k \in \mathbb{R}^{p \times 1}$. We assume that the output $\mathbf{y}$ is equal to the state $\mathbf{x}$, and that the output $\mathbf{y}$ thus does not need to be taken into account in the objective function:

$$\mathbf{y} \equiv \mathbf{x} \tag{C.7}$$

By defining the stacked vector $\mathbf{X}$ as follows:

$$\mathbf{X} \quad = \quad \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_{N_P} \end{bmatrix} \tag{C.8}$$

with $\mathbf{X} \in \mathbb{R}^{N_P q \times 1}$, and using the definition of $\mathbf{U}$ of Eqn (C.3), Eqn (C.6) can be rewritten as:

$$\mathbf{X} = \mathbf{A}^* \mathbf{x}_0 + \mathbf{B}^* \mathbf{U} \tag{C.9}$$

with

$$\mathbf{A}^* = \begin{bmatrix} \mathbf{A} \\ \mathbf{A}^2 \\ \vdots \\ \mathbf{A}^{N_P} \end{bmatrix} \tag{C.10}$$

$$\mathbf{B}^* = \begin{bmatrix} \mathbf{B} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{AB} & \mathbf{B} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{N_P-1}\mathbf{B} & \mathbf{A}^{N_P-2}\mathbf{B} & \cdots & \mathbf{A}^{N_P-N_C}\mathbf{B} \end{bmatrix} \tag{C.11}$$

and $\mathbf{A}^* \in \mathbb{R}^{N_P q \times q}$ and $\mathbf{B}^* \in \mathbb{R}^{N_P q \times N_C p}$. Assume that the constraints are given in terms of the stacked vector $\mathbf{U}$ and $\mathbf{X}$ of Eqn (C.3) and Eqn (C.8). The constraints can then be written as:

$$\mathbf{CX} + \mathbf{DU} \leq \mathbf{d} \tag{C.12}$$

with $\mathbf{C} \in \mathbb{R}^{s \times N_P q}$, $\mathbf{D} \in \mathbb{R}^{s \times N_C p}$ and $\mathbf{d} \in \mathbb{R}^{s \times 1}$. The objective function of the MPC can, according to Eqn (2.1), be written as:

$$\begin{aligned} V(\mathbf{y}_k, \mathbf{x}_j, \mathbf{u}_t) = & \sum_{k=1}^{N_P} (\mathbf{y}_k - \mathbf{y}_{ref,k})^T \mathbf{S}_k (\mathbf{y}_k - \mathbf{y}_{ref,k}) \\ & + \sum_{j=1}^{N_P} (\mathbf{x}_j - \mathbf{x}_{ref,j})^T \mathbf{Q}_j (\mathbf{x}_j - \mathbf{x}_{ref,j}) \\ & + \sum_{t=0}^{N_C-1} \mathbf{u}_t^T \mathbf{R}_t \mathbf{u}_t \end{aligned} \tag{C.13}$$

with $\mathbf{S}_k \in \mathbb{R}^{v \times v}$, $\mathbf{Q}_j \in \mathbb{R}^{q \times q}$ and $\mathbf{R}_t \in \mathbb{R}^{p \times p}$. For the sake of simplicity the term in $\mathbf{y}_k$ is dropped according to Eqn (C.7). The equation can be rewritten in terms of the stacked vectors of Eqn (C.3) and Eqn (C.8) as follows:

$$\begin{aligned} V(\mathbf{X}, \mathbf{U}) = & \left(\mathbf{X} - \mathbf{X}_{ref}\right)^T \mathbf{Q}^* \left(\mathbf{X} - \mathbf{X}_{ref}\right) \\ & + \mathbf{U}^T \mathbf{R}^* \mathbf{U} \end{aligned} \tag{C.14}$$

with $\mathbf{X}_{ref} \in \mathbb{R}^{N_P q \times 1}$ the reference vector for the stacked state vector $\mathbf{X}$, and matrices $\mathbf{Q}^*$ and $\mathbf{R}^*$ defined based on the matrices $\mathbf{Q}_j$ and $\mathbf{R}_t$ as follows:

$$\mathbf{Q}^* = \begin{bmatrix} \mathbf{Q}_1 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{Q}_{N_P} \end{bmatrix} \tag{C.15}$$

$$\mathbf{R}^* = \begin{bmatrix} \mathbf{R}_0 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{R}_{N_C-1} \end{bmatrix} \tag{C.16}$$

with $\mathbf{Q}^* \in \mathbb{R}^{N_P q \times N_P q}$ and $\mathbf{R}^* \in \mathbb{R}^{N_C p \times N_C p}$. As stated above, the goal is to write the objective function and the constraints in terms of the input (the optimization variable), the initial state and the reference vector. The objective function of Eqn (C.14) can be rewritten using Eqn (C.9) as follows:

$$\begin{aligned} V(\mathbf{x}_0, \mathbf{U}) &= (\mathbf{A}^*\mathbf{x}_0 + \mathbf{B}^*\mathbf{U})^T \mathbf{Q}^* (\mathbf{A}^*\mathbf{x}_0 + \mathbf{B}^*\mathbf{U}) \\ &\quad -2\mathbf{X}_{ref}^T \mathbf{Q}^* (\mathbf{A}^*\mathbf{x}_0 + \mathbf{B}^*\mathbf{U}) + \mathbf{U}^T \mathbf{R}^* \mathbf{U} \\ &\quad +\mathbf{X}_{ref}^T \mathbf{Q}^* \mathbf{X}_{ref} \qquad \text{(C.17)} \\ &= \mathbf{U}^T \left(\mathbf{R}^* + \mathbf{B}^{*T} \mathbf{Q}^* \mathbf{B}^* \right) \mathbf{U} \\ &\quad + \left(2\mathbf{x}_0^T \mathbf{A}^{*T} \mathbf{Q}^* - 2\mathbf{X}_{ref}^T \mathbf{Q}^* \right) \mathbf{B}^* \mathbf{U} \\ &\quad +\mathbf{x}_0^T \mathbf{A}^{*T} \mathbf{Q}^* \mathbf{A}^* \mathbf{x}_0 - 2\mathbf{X}_{ref}^T \mathbf{Q}^* \mathbf{A}^* \mathbf{x}_0 + \mathbf{X}_{ref}^T \mathbf{Q}^* \mathbf{X}_{ref} \qquad \text{(C.18)} \end{aligned}$$

All the terms that do not depend on $\mathbf{U}$ can be dropped from the objective function, since they will not have any effect on the result. The matrices defined in Eqn (C.4) thus are:

$$\mathbf{H} = \left(\mathbf{R}^* + \mathbf{B}^{*T} \mathbf{Q}^* \mathbf{B}^* \right) \qquad \text{(C.19)}$$

$$\mathbf{F_R} = -\mathbf{B}^{*T} \mathbf{Q}^{*T} \qquad \text{(C.20)}$$

$$\mathbf{F_X} = \mathbf{B}^{*T} \mathbf{Q}^{*T} \mathbf{A}^* \qquad \text{(C.21)}$$

with $\mathbf{H} \in \mathbb{R}^{N_C p \times N_C p}$, $\mathbf{F_R} \in \mathbb{R}^{N_C p \times N_P q}$ and $\mathbf{F_X} \in \mathbb{R}^{N_C p \times q}$. The constraints of Eqn (C.12) can also be rewritten making use of Eqn (C.9). This results in:

$$\mathbf{C}\mathbf{A}^*\mathbf{x}_0 + (\mathbf{C}\mathbf{B}^* + \mathbf{D})\,\mathbf{U} \leq \mathbf{d} \qquad \text{(C.22)}$$

The matrices of Eqn (C.2) can now be defined as follows:

$$\mathbf{E} = \mathbf{C}\mathbf{A}^* \qquad \text{(C.23)}$$

$$\mathbf{F} = \mathbf{C}\mathbf{B}^* + \mathbf{D} \qquad \text{(C.24)}$$

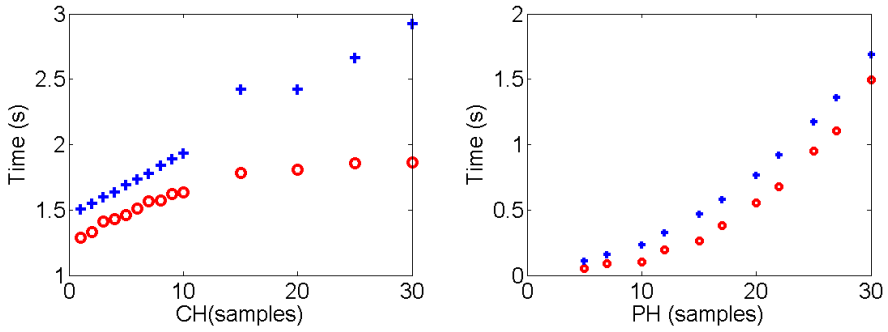
$$\mathbf{b} = \mathbf{d} \qquad \text{(C.25)}$$

with $\mathbf{E} \in \mathbb{R}^{s \times q}$, $\mathbf{F} \in \mathbb{R}^{s \times N_C p}$ and $\mathbf{b} \in \mathbb{R}^{s \times 1}$.

The matrices $\mathbf{H}$, $\mathbf{F_R}$, $\mathbf{F_X}$, $\mathbf{E}$, $\mathbf{F}$ and $\mathbf{b}$ are the inputs of the optimization method as described in Chapter 9. All the matrices, except for $\mathbf{b}$, depend on $\mathbf{A}^*$ and/or $\mathbf{B}^*$. This means that these matrices need to be recalculated each time the model needs to be updated. In the next section the computational load of this calculation is analyzed.

C.2 Computational load

In this section the computational load incurred with each model update is analyzed as a function of the control and the prediction horizon. The sampling rate of the model is 5 Hz. The focus lies on the performance on the PXI system, since this is also used in practice (see Appendix A). When compared to a Windows system (in Fig. C.1), the following system is used (the same as in Appendix A):



(a) (b)

Figure C.1: (a) The time (in seconds) required for a model update with a fixed prediction horizon (PH) (30 samples) and a variable control horizon (CH) (X-axis). The results are given for execution on Windows ('+') and on the PXI system ('o').
(b) The time (in seconds) required for a model update with a fixed control horizon (CH) (5 samples) and a variable prediction horizon (PH) (X-axis). The results are given for execution on Windows ('+') and on the PXI system ('o').

- Intel T2500 2GHz CPU (dual core)
- 1GB RAM
- Windows XP Professional

C.2.1 Complexity analysis

This section discusses the complexity analysis of the algorithm as it was implemented in the controller. Sparsity that was not taken advantage of in this implementation, is not taken into account here either.

The time needed to update the matrices $\mathbf{A}^*$ and $\mathbf{b}^*$ can be neglected relative to the time needed for the matrix multiplications to obtain $\mathbf{H}$, $\mathbf{F_R}$, $\mathbf{F_X}$, $\mathbf{E}$ and $\mathbf{F}$. The computational complexity of these calculations is expressed in terms of the problem dimensions. Only the matrix multiplications are taken into account, since matrix additions do not have a significant impact on the complexity.

H: Eqn (C.19) The calculation of $\mathbf{H}$ contains two multiplications which are:

$$\mathbf{B}^{*T}\mathbf{Q}^*, \left(\mathbf{B}^{*T}\mathbf{Q}^*\right)\mathbf{B}^* \quad : \quad O\left(N_C p \left(N_P q\right)^2\right) \tag{C.26}$$

$\mathbf{F_R}$: Eqn (C.20) The calculation of $\mathbf{F_R}$ contains one matrix multiplication which is:

$$\mathbf{B}^{*T}\mathbf{Q}^{*T} \quad : \quad O\left(N_C p \left(N_P q\right)^2\right) \tag{C.27}$$

$\mathbf{F_X}$: Eqn (C.21) The calculation of $\mathbf{F_X}$ contains two multiplications, which are:

$$\mathbf{B}^{*T}\mathbf{Q}^* \quad : \quad O\left(N_C p \left(N_P q\right)^2\right) \tag{C.28}$$

$$\left(\mathbf{B}^{*T}\mathbf{Q}^{*T}\right)\mathbf{A}^* \quad : \quad O\left(N_C p N_P q^2\right) \tag{C.29}$$

E: Eqn (C.23) The calculation of $\mathbf{E}$ contains one multiplication which is:

$$\mathbf{C}\mathbf{A}^{*T} \quad : \quad O\left(sN_P q^2\right) \tag{C.30}$$

F: Eqn (C.24) The calculation of $\mathbf{F}$ contains one multiplication which is:

$$\mathbf{C}^T\mathbf{B}^* \quad : \quad O\left(sN_P q N_C p\right) \tag{C.31}$$

Overall complexity Based on the above analysis, the effect of varying the problem dimensions on the model update time can be estimated. The computation time is quadratic in terms of the prediction horizon and the number of states, and varies linearly with the control horizon, the number of inputs and the number of constraints.

C.2.2 Experimental results

To simplify the analysis, the effect of the control and the prediction horizon is examined independently. In other words, the control horizon is varied with a fixed prediction horizon, and the prediction horizon is varied with a fixed control horizon. Each model update is executed five times (with an identical process model) to be able to discard random effects.

Fixed prediction horizon In Table C.1 the control horizon is varied for a fixed prediction horizon of 30 samples. As discussed in Chapter 7, the control horizon can often be set smaller than the prediction horizon, without loss of performance. Especially with an uncertain system, there is not much use in calculating the input far in the future. After all, only the first input will be applied to the machine, and all the next inputs will be discarded. In Chapter 7 the effect of the control horizon on the performance is examined. This effect depends on the process at hand. In Fig. C.1 (a) the results of Table C.1 are plotted, together with the results on a Windows platform. Note that there is a significant difference in timing behavior for large horizons, which, at first sight, cannot be explained. The behavior is, as could be expected from the explanation in the previous section, more or less linear with the control horizon (except for the deviation at higher control horizon values for the PXI system).

Fixed control horizon In Table C.2 the prediction horizon is varied for a fixed control horizon of 5 samples. In Chapter 7 the effect of the prediction horizon on the performance is examined. In Fig. C.1 (b) the results of Table C.2 are plotted, together with the results on a Windows platform. In this case the results on the Windows system and the PXI system (which is actually used in the field, see Appendix A) are more or less equivalent. The behavior is, as follows from the explanation in the previous section, more or less quadratic with the prediction horizon.

Table C.1: The calculation time (in seconds) needed to generate the matrices of Eqn (C.4) and Eqn (C.2) (with 2 constraints). Each calculation is repeated five times to eliminate random effects. The prediction horizon is kept constant at 30 samples. The control horizon is varied from 1 to 30 samples to show the effect of limiting the control horizon. For each control horizon the minimum time (Min.), average time (Av.), median time (Med.) and maximum time (Max.) is shown, as well as the standard deviation (Std.) over the five realizations.

	Control horizon													
It.	1	2	3	4	5	6	7	8	9	10	15	20	25	30
1	1.304	1.353	1.358	1.423	1.443	1.518	1.585	1.630	1.657	1.628	1.756	1.852	1.834	1.885
2	1.275	1.327	1.451	1.448	1.467	1.564	1.571	1.571	1.588	1.644	1.801	1.772	1.881	1.811
3	1.276	1.300	1.442	1.466	1.446	1.517	1.503	1.551	1.605	1.612	1.777	1.816	1.839	1.888
4	1.328	1.329	1.387	1.375	1.463	1.503	1.589	1.549	1.657	1.603	1.785	1.795	1.850	1.878
5	1.255	1.332	1.418	1.426	1.476	1.446	1.591	1.548	1.608	1.693	1.806	1.809	1.875	1.847
Min.	1.255	1.300	1.358	1.375	1.443	1.446	1.503	1.548	1.588	1.603	1.756	1.772	1.834	1.811
Av.	1.288	1.328	1.411	1.428	1.459	1.510	1.568	1.570	1.623	1.636	1.785	1.809	1.856	1.862
Med.	1.276	1.329	1.418	1.426	1.463	1.517	1.585	1.551	1.608	1.628	1.785	1.809	1.850	1.878
Max.	1.328	1.353	1.451	1.466	1.476	1.564	1.591	1.630	1.657	1.693	1.806	1.852	1.881	1.888
Std.	0.029	0.019	0.039	0.034	0.014	0.042	0.037	0.035	0.032	0.036	0.020	0.029	0.021	0.033

Table C.2: The calculation time (in seconds) needed to generate the matrices of Eqn (C.4) and Eqn (C.2) (with 2 constraints). Each calculation is repeated five times to eliminate random effects. The control horizon is kept constant at 5 samples. The prediction horizon is varied from 5 to 30 samples to show the effect of limiting the control horizon. For each prediction horizon the minimum time (Min.), average time (Av.), median time (Med.) and maximum time (Max.) is shown, as well as the standard deviation (Std.) over the five realizations.

	Prediction horizon											
It.	5	7	10	12	15	17	20	22	25	27	30	
1	0.053	0.087	0.105	0.196	0.264	0.379	0.555	0.679	0.953	1.107	1.497	
2	0.064	0.055	0.106	0.177	0.251	0.371	0.515	0.713	0.954	1.077	1.463	
3	0.064	0.065	0.107	0.165	0.263	0.339	0.505	0.69	0.968	1.183	1.505	
4	0.052	0.107	0.107	0.155	0.251	0.35	0.505	0.637	0.912	1.149	1.437	
5	0.063	0.075	0.117	0.156	0.264	0.339	0.502	0.634	0.877	1.066	1.499	
Min.	0.052	0.055	0.105	0.155	0.251	0.339	0.502	0.634	0.877	1.066	1.437	
Av.	0.059	0.078	0.108	0.170	0.259	0.356	0.516	0.671	0.933	1.116	1.480	
Med.	0.063	0.075	0.107	0.165	0.263	0.350	0.505	0.679	0.953	1.107	1.497	
Max.	0.064	0.107	0.117	0.196	0.264	0.379	0.555	0.713	0.968	1.183	1.505	
Std.	0.006	0.020	0.005	0.017	0.007	0.018	0.022	0.034	0.038	0.049	0.029	

Model update rate Throughout this text the most used horizon setting is a prediction horizon of 30 samples and a control horizon of 5 samples. According to Table C.1 and Table C.2 this model update would take almost 1.5 seconds. Considering the rate of change of the system, a model update rate of 1 Hz seems necessary. If one model update would take a little under 1.5 seconds, a model update rate of 1 Hz would be impossible. Now, by reducing the computational complexity of the model update as explained in section C.3, the same model update can be executed much faster.

C.3 Reduce the matrix sizes

Chapter 6 and Chapter 7 discuss the model structure used by the MPC in detail. Due to the nature of the process, which contains a lot of transport elements, the model contains a lot of delays. The dynamics also vary significantly across the different process variables. Some variables show very fast (and thus negligible) dynamics, other variables have slow to extremely slow dynamics. In Chapter 7 is stated that the sampling rate of the model depends on the dynamics, which is backed by the Nyquist criterion. It is clear that if there is a part of the system with very slow dynamics, and a part with fast dynamics, this will lead to oversampling the slow part of the system.

Sampling rate Oversampling means that the model uses more states than strictly necessary, and as is shown in section C.2.1, the number of states has an important impact on the model update time. The variables needed for the capacity control system can easily be divided in two classes:

fast dynamics: this includes the input (the hydrostatic flow), the ground speed, the feed rate and the engine load. Fortunately, all these signals also have small delays (see Chapter 6).

slow dynamics: this includes grain mass flow, and - as soon as they are included in the model - also sieve losses and return flow.

Because the slow dynamics occur with the signals with the largest delays, and since these are also the last measurements in the transport of the crop through the machine, the sampling rate of the model can easily be split. The first half of the model has a sampling rate of 5 Hz, and the second half has a sampling rate of 1 Hz.

Practical implementation A split sampling rate can be realized without touching the model. The delay between feed rate and grain mass flow, and between engine load and grain mass flow is divided by 5 to bring it from 5 to 1 Hz. The model itself however, remains unchanged. This means that the predictions of the model across the prediction horizon assume a faster response of grain mass flow than the real response. Since there is no feedback, and since no other process variables depend on grain mass flow, this faulty prediction has no effect on the controller what so ever. If the feedback caused by the return flow would also be modeled, this approach would no longer be possible.

The inputs of the controller, being the current state $\mathbf{x}_0$ and the reference vector $\mathbf{X}_{ref}$, do have to be modified:

$\mathbf{x}_0$**:** the process condition (with 5 Hz delays) is updated in every time step. Each iteration this process condition is then transformed in a process condition with a mixed 5 Hz/1 Hz sampling rate.

$\mathbf{X}_{ref}$**:** since the structure of the initial state has been changed, the structure of the reference has to be changed in an analogous manner.

Redundant grain mass flow model Grain mass flow depends on a combination of a model using feed rate and a model using engine load. Since the combination of these models depends on the situation, both models have to be delayed separately. This redundant grain mass flow model is at the core of the approach to uncertainty in this research. If one measurement fails, or one type of model is not reliable in a given field, the overall model can reconfigure automatically by according a larger weight to the second model. As described in the next paragraph, this nice property comes at a cost. If there is only one grain mass flow model, only one delay line is required, which would greatly reduce the problem size.

Problem size The redundant grain mass flow model requires two delay lines of about 50 samples (10 seconds) at 5 Hz, or two delay lines of 10 samples at 1 Hz. Moreover, if Stochastic MPC is used, the model doubles in size (because there is also an uncertainty model). The total size reduction thus is $2 \times 2 \times 40$ or 160 states, reducing the model from about 220 states to 60 states. In section C.2.1 the effect of reducing the number of states on the computation time is found to be quadratic. In Table C.3 the computation times with reduced number of states are shown as a function of the prediction horizon. The time required for a model update now reduces from 1.5 seconds to 0.2 seconds. The above described matrix reduction thus enables a model update rate of 1 Hz.

Feed rate control If the MPC is only used for feed rate or engine load control, there is no reason to model the slow grain mass flow. This will lead to even lower calculation times than the above described approach since grain mass flow - and its delay lines - can be dropped from the model entirely.

Table C.3: The calculation time (in seconds) needed to generate the matrices of Eqn (C.4) and Eqn (C.2) (with 2 constraints). However, the slow dynamics are now modeled at 1 Hz, which significantly reduces the calculation time. Each calculation is repeated five times to eliminate random effects. The control horizon is kept constant at 5 samples. The prediction horizon is varied from 5 to 30 samples to show the effect of limiting the control horizon. For each prediction horizon the minimum time (Min.), average time (Av.), median time (Med.) and maximum time (Max.) is shown, as well as the standard deviation (Std.) over the five realizations.

	Prediction horizon													
It.	5	7	10	12	15	17	20	22	25	27	30	40	50	60
1	0.002	0.004	0.01	0.015	0.038	0.069	0.078	0.09	0.124	0.151	0.234	0.533	1.1	1.814
2	0.002	0.004	0.02	0.037	0.038	0.069	0.067	0.081	0.176	0.16	0.214	0.541	1.03	1.684
3	0.002	0.004	0.02	0.026	0.05	0.04	0.069	0.083	0.134	0.182	0.202	0.542	1.104	1.824
4	0.013	0.003	0.021	0.026	0.066	0.049	0.094	0.072	0.126	0.168	0.216	0.512	1.118	1.77
5	0.002	0.003	0.021	0.017	0.048	0.04	0.058	0.073	0.122	0.134	0.203	0.541	1.065	1.633
Min.	0.002	0.003	0.010	0.015	0.038	0.040	0.058	0.072	0.122	0.134	0.202	0.512	1.030	1.633
Av.	0.004	0.004	0.018	0.024	0.048	0.053	0.073	0.080	0.136	0.159	0.214	0.534	1.083	1.745
Med.	0.002	0.004	0.020	0.026	0.048	0.049	0.069	0.081	0.126	0.160	0.214	0.541	1.100	1.770
Max.	0.013	0.004	0.021	0.037	0.066	0.069	0.094	0.090	0.176	0.182	0.234	0.542	1.118	1.824
Std.	0.005	0.001	0.005	0.009	0.011	0.015	0.014	0.007	0.023	0.018	0.013	0.013	0.036	0.084

C.4 Conclusion

An essential aspect of an adaptive controller is updating the controller with the new process model. In this research the approach is taken to minimize the online optimization time in every time step, which requires to use the most efficient problem representation.

Unfortunately, the most efficient representation for online optimization takes quite some computation time at every update. Since updating the model is essentially an offline event, this does not have to be a problem. The model update rate can be lowered to keep the calculation feasible. The only constraint is that enough model updates need to be performed to keep the optimization model in line with reality.

The computation time of a model update is quadratic in terms of the prediction horizon and the number of states, and varies linearly with the control horizon, the number of inputs and the number of constraints. The required calculation time can be further reduced by decreasing the number of required delay states.

Appendix D

Performance measures

In order to be able to compare different controllers, the controller performance needs to be quantified. In this appendix the measures which are used throughout the text are described. The performance of a controller consists of the input effort and the control error. Crop entry or set point changes should be treated separately from stationary runs.

D.1 Transition phenomena

For transition phenomena there are several standard measures available. In this text the following definitions are used (Franklin et al., 2006):

rise time t_r: the time required by the system to move from 10% of the set point to 90% of the set point.

settling time t_s: the time required for the system to arrive and stay within a 5% bound of the set point, after removal of possible steady-state offset.

overshootM_p: the largest value of the system response expressed in percentage of the set point.

Rise time is not a practical measure for this application, since the process value often does not start from zero because of material left in the system, which renders the rise time a very unreliable measure.

In case of disturbances, the term recovery time is used instead of settling time. Recovery time is then the time needed to recover from the disturbance and to return to a narrow band around the set point.

D.2 Regime conditions

As mentioned above, input effort and control error are the two most important aspects of the controller. One could simply calculate the RMS values of the input and the error, but this would neglect two very important issues:

- not all input changes are experienced the same. A continuous slow input change does not bother the operator whatsoever, whereas one or several aggressive input changes do bother the operator.
- the set point error has two causes, namely:

  changing conditions: if the conditions change, the input will have to be changed in order for the set point to be maintained. Error due to changing conditions is indeed control error.

  fast measurement noise: the controller is not supposed to suppress high-frequency noise. Thus, if the system contains noise the RMS value of the error is no longer a good measure for the controller performance.

D.2.1 Methodology

Both above described flaws in using the RMS value as performance measure can be dealt with in the same way: by including frequency information. The signal is frequency-wise split in two parts, a low-frequency part and a high-frequency part. For input effort the most important measure of the performance of the controller is the RMS value of the high-frequency part of the input signal, whereas for control error the most important measure is the RMS value of the low-frequency part of the controller.

D.2.2 Input effort

First of all, it is the change of pump setting that is relevant for the controller performance. After all, the actual speed has no effect on operator comfort. The acceleration however, needs to be within given boundaries.

The definition of 'low-frequency' and 'high-frequency' for input changes depends on operator comfort. A lot of studies have been done to determine the effect of vibrations on the comfort and the health of operators (Broman et al., 1996, De Temmerman et al., 2005, Deprez et al., 2005), and the conclusion is that frequencies higher than 0.5-1 Hz have a negative effect on operator comfort.

Moreover, as illustrated in Chapter 6 (section 6.2), the system only responds to low-frequency inputs. This means that inputs at higher frequencies can only cause undesired effects (such as reduce comfort and increase wear), and do not improve the control error.

The input signal is split in a low- and high-frequency component using a second order Butterworth filter with cut-off frequency 0.75 Hz (see Fig. D.1).

D.2.3 Control error

The criterium to split the error signal in two parts (frequency-wise) is the noise content. This depends on the signal at hand. In view of the time domain model of section 6.2 (Chapter 6), it is safe to assume that there is no significant system response higher than 0.8-1 Hz. This means that all process variations at frequencies higher than 1 Hz can be classified as noise.

The error signal will thus be split into two parts with a second order Butterworth filter with cut-off at 1 Hz (see Fig. D.2).

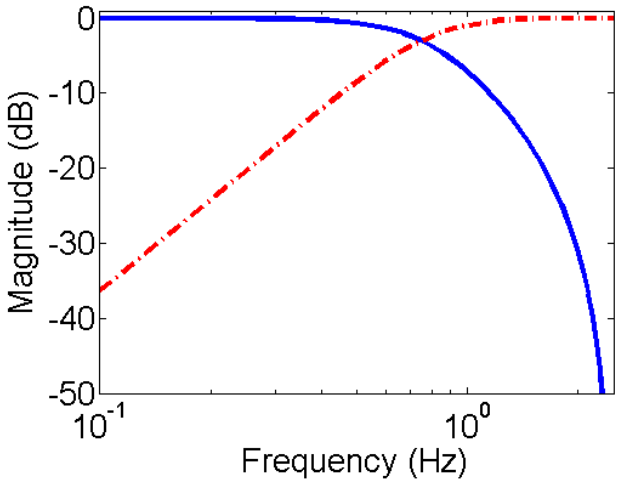


Figure D.1: The lowpass (solid line) and high-pass (dash-dot line) second order Butterworth filter which is used to split the input signal. The cut-off is located at 0.75 Hz. The sampling rate of these signals is 5 Hz.

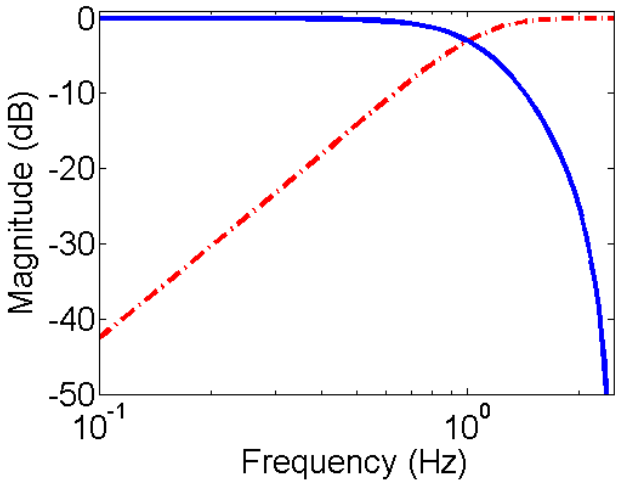


Figure D.2: The lowpass (solid line) and high-pass (dash-dot line) second order Butterworth filter which is used to split the error signal. The cut-off is located at 1 Hz. The sampling rate of these signals is 5 Hz.

D.3 Conclusion

In this appendix several measures are defined to quantify the performance of a controller, for transient as well as for regime conditions.

Transients The controller can be scored using the settling time, overshoot and recovery time.

Regime conditions The input effort is now quantified with three parameters:

- overall input effort I_C: the RMS value of the input signal.
- low-frequency input effort I_L: input changes needed to control the system.
- high-frequency effort I_H: input changes which affect operator comfort, and moreover, have no significant impact on the error.

The control error leads to three similar parameters, albeit with a slightly different interpretation:

- overall control error E_C: the RMS value of the error signal.
- low-frequency error E_L: error caused by too slow control actions. This is the best measure to quantify the control error performance.
- high-frequency error E_H: error due to process noise, which thus is not relevant for the controller performance.

Appendix E

Parameter uncertainty

In this appendix the detailed calculations are shown concerning parameter uncertainty, as discussed in Chapter 8.

E.1 Parameter multiplication

Consider the following example:

$$\begin{aligned} x_{k+1} &= x_k + K_1 u_k \\ z_{k+1} &= K_2 x_k \end{aligned} \tag{E.1}$$

where K_1 and K_2 are stochastic gain parameters, which can be written as:

$$K_1 = \hat{K}_1 + \tilde{K}_1 \tag{E.2}$$

$$K_2 = \hat{K}_2 + \tilde{K}_2 \tag{E.3}$$

with

$$\tilde{K}_1 \propto \mathcal{N}\left(0, (\hat{\sigma}_{K_1})^2\right) \tag{E.4}$$

$$\tilde{K}_2 \propto \mathcal{N}\left(0, (\hat{\sigma}_{K_2})^2\right) \tag{E.5}$$

where the hat indicates that the value is estimated online. $\epsilon_{x,k}$ and $\epsilon_{z,k}$ are the predicted uncertainties at time step k of the states x and z. The uncertainty model can then be written as follows:

$$\begin{aligned} \epsilon_{x,k+1} &= \epsilon_{x,k} + \hat{\sigma}_{K_1} u_k \\ \epsilon_{z,k+1} &= \hat{\sigma}_2 \epsilon_{x,k} \end{aligned} \tag{E.6}$$

with $\epsilon_{x,0} = 0$.

In order to see how to calculate $\hat{\sigma}_2$, the equation of z needs to be rewritten in terms of u and z, using Eqn (E.1):

$$z_{k+1} = K_2 x_0 + K_2 K_1 (u_0 + u_1 + \cdots + u_{k-1}) \tag{E.7}$$

based on Eqn (E.6) one can easily show that the above equation is equivalent with:

$$z_{k+1} = K_2 x_0 + K_2 K_1 \frac{\epsilon_{x,k}}{\hat{\sigma}_{K_1}} \tag{E.8}$$

neglecting the term in $\tilde{K}_2 x_0$, this can be written as:

$$\bar{z}_{k+1} + \tilde{z}_{k+1} = \hat{K}_2 x_0 + \left(\hat{K}_2 + \tilde{K}_2\right)\left(\hat{K}_1 + \tilde{K}_1\right)\frac{\epsilon_{x,k}}{\hat{\sigma}_{K_1}} \tag{E.9}$$

where $\bar{z}_k$, $\hat{K}_1$ and $\hat{K}_2$ are the deterministic part of respectively z at time step k, K_1 and K_2. $\tilde{z}_k$, $\tilde{K}_1$ and $\tilde{K}_2$ are the stochastic part of respectively z at time step k, K_1 and K_2. This leads to:

$$\tilde{z}_{k+1} = \left(\bar{K}_1\tilde{K}_2 + \bar{K}_2\tilde{K}_1 + \tilde{K}_1\tilde{K}_2\right)\frac{\epsilon_{x,k}}{\hat{\sigma}_{K_1}} \tag{E.10}$$

Since $\sigma_{z,k} = \sqrt{\mathrm{E}\left[\tilde{z}_k\right]}$, $\sigma_{z,k+1}$ can be approximated as $\epsilon_{z,k+1}$:

$$\epsilon_{z,k+1} = \sqrt{\mathrm{E}\left[\tilde{z}_{k+1}\right]} = \sqrt{\mathrm{E}\left[\left(\hat{K}_1\tilde{K}_2 + \hat{K}_2\tilde{K}_1 + \tilde{K}_1\tilde{K}_2\right)^2\right]}\frac{\epsilon_{x,k}}{\hat{\sigma}_{K_1}} \tag{E.11}$$

combining Eqn (E.6) with Eqn (E.11) leads to a definition for $\hat{\sigma}_2$:

$$\hat{\sigma}_2 = \frac{\sqrt{\mathrm{E}\left[\left(\bar{K}_1\tilde{K}_2 + \bar{K}_2\tilde{K}_1 + \tilde{K}_1\tilde{K}_2\right)^2\right]}}{\hat{\sigma}_{K_1}} \tag{E.12}$$

Using Eqn (E.4), (E.5) and $\mathrm{E}\left[\tilde{K}_1\tilde{K}_2\right] = 0$ (independent stochastic variables), the expected value can be calculated:

$$\begin{aligned}
\mathrm{E}\left[\left(\hat{K}_1\tilde{K}_2 + \hat{K}_2\tilde{K}_1 + \tilde{K}_1\tilde{K}_2\right)^2\right] &= \mathrm{E}\left[\left(\hat{K}_1\right)^2\tilde{K}_2^2 + \left(\hat{K}_2\right)^2\tilde{K}_1^2 + \tilde{K}_1^2\tilde{K}_2^2\right] \\
&\quad +2\mathrm{E}\left[\hat{K}_1\hat{K}_2\tilde{K}_1\tilde{K}_2\right] \qquad \text{(E.13)} \\
&\quad +\mathrm{E}\left[2\hat{K}_1\tilde{K}_1\left(\tilde{K}_2\right)^2 + 2\hat{K}_2\left(\tilde{K}_1\right)^2\tilde{K}_2\right] \\
&= \left(\hat{K}_1\right)^2\mathrm{E}\left[\left(\tilde{K}_2\right)^2\right] + \left(\hat{K}_2\right)^2\mathrm{E}\left[\left(\tilde{K}_1\right)^2\right] \\
&\quad +\mathrm{E}\left[\left(\tilde{K}_1\right)^2\left(\tilde{K}_2\right)^2\right] \qquad \text{(E.14)} \\
&= \left(\hat{K}_1\right)^2(\sigma_{K_2})^2 + \left(\hat{K}_2\right)^2(\hat{\sigma}_{K_1})^2 \\
&\quad +(\hat{\sigma}_{K_1})^2(\hat{\sigma}_{K_2})^2 \qquad \text{(E.15)}
\end{aligned}$$

Combining this with Eqn (E.12) leads to:

$$\hat{\sigma}_{K_1,K_2} = \frac{\sqrt{\left(\hat{K}_1\right)^2(\hat{\sigma}_{K_2})^2 + \left(\hat{K}_2\right)^2(\hat{\sigma}_{K_1})^2 + (\hat{\sigma}_{K_1})^2(\hat{\sigma}_{K_2})^2}}{\hat{\sigma}_{K_1}} \tag{E.16}$$

E.2 Calculate the model uncertainty

This section gives the details of the calculation of the model uncertainty used in Chapter 8, section 8.3.

k=1:

$$\begin{aligned} \epsilon_{z,1} &= \hat{\sigma}_{K_1,K_2}\epsilon_{x,0} \\ &= 0 \end{aligned} \tag{E.17}$$

k=2:

$$\epsilon_{x,1} = \hat{\sigma}_{K_1} u_0 \tag{E.18}$$

$$\begin{aligned} \epsilon_{z,2} &= \hat{\sigma}_{K_1,K_2}\epsilon_{x,1} \\ &= \hat{\sigma}_{K_1,K_2}\hat{\sigma}_{K_1} u_0 \end{aligned} \tag{E.19}$$

k=3:

$$\epsilon_{x,2} = \hat{\sigma}_{K_1}\left(u_0 + u_1\right) \tag{E.20}$$

$$\begin{aligned} \epsilon_{z,3} &= \hat{\sigma}_{K_1,K_2}\epsilon_{x,2} \\ &= \hat{\sigma}_{K_1,K_2}\hat{\sigma}_{K_1}\left(u_0 + u_1\right) \end{aligned} \tag{E.21}$$

k=4:

$$\epsilon_{x,3} = \hat{\sigma}_{K_1}\left(u_0 + u_1 + u_2\right) \tag{E.22}$$

$$\begin{aligned} \epsilon_{z,4} &= \hat{\sigma}_{K_1,K_2}\epsilon_{x,3} \\ &= \hat{\sigma}_{K_1,K_2}\hat{\sigma}_{K_1}\left(u_0 + u_1 + u_2\right) \end{aligned} \tag{E.23}$$

Appendix F

System matrices

This appendix gives the system matrices of the applications used in this research. The detailed application description can be found in respectively Chapter 6, Chapter 10 and Chapter 11.

F.1 Capacity control on a combine harvester

Details of this application can be found in Chapter 6, section 6.6. The state-space model is:

$$\mathbf{x}_{k+1} \quad = \quad \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{v}_k, \tag{F.1}$$

where the dimensions of the matrices and vectors depend on the delays used in the model. The vectors $\mathbf{x}_k$ and $\mathbf{v}_k$ are:

$$\mathbf{x}_k = \begin{bmatrix} GS_k^0 \\ \vdots \\ GS_k^{d_1} \\ FR_k^0 \\ \vdots \\ FR_k^{d_2} \\ EL_k^T \\ EL_k^0 \\ \vdots \\ EL_k^{d_3} \\ GMF_k^{T,FR} \\ GMF_k^{T,EL} \\ GMF_k \end{bmatrix} \qquad \mathbf{v}_k = \begin{bmatrix} \Delta u_k \end{bmatrix} \tag{F.2}$$

with

$$\begin{aligned} d_1 &= \max(d_{EL}, d_{FR}) + 1 \\ d_2 &= d_{FR-GMF} + 1 \\ d_3 &= d_{EL-GMF} + 1 \\ 1 &= \gamma_1 + \gamma_2 \end{aligned} \tag{F.3}$$

The matrix $\mathbf{A}$ can be quite easily derived from Eqn (6.17). Since the matrix is very large, it is not shown here. $\mathbf{B}$ can be written as:

$$\mathbf{B} = \begin{bmatrix} K_{GS} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{F.4}$$

F.2 Cruise control on a combine harvester

The details of this application can be found in Chapter 10. The linearized system looks as follows:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}_L \mathbf{x}_k + \mathbf{B}_L \mathbf{v}_k, \\ \mathbf{y}_k = \mathbf{C}_L \mathbf{x}_k + \mathbf{D}_L \mathbf{v}_k \end{cases} \tag{F.5}$$

with system matrices $\mathbf{A}_L \in \mathbb{R}^{5\times5}$, $\mathbf{B}_L \in \mathbb{R}^{5\times2}$, $\mathbf{C}_L \in \mathbb{R}^{1\times5}$ and $\mathbf{D}_L \in \mathbb{R}^{1\times2}$. The state vector $\mathbf{x}_k$ now contains five elements: the two model states of Eqn (10.4), the input state of Eqn (10.13) and two integrator states for PS_k and ES_k. The input vector is $\mathbf{v}_k = [\Delta PS_k; \Delta ES_k]$ and the matrices $\mathbf{A}_L$, $\mathbf{B}_L$, $\mathbf{C}_L$ and $\mathbf{D}_L$ are:

$$\begin{aligned} \mathbf{A}_L &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1.9105 & -0.9141 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\ \mathbf{B}_L &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \beta ES_{k-1} & \beta PS_{k-1} \\ 0 & 0 \\ \beta ES_{k-1} & \beta PS_{k-1} \end{bmatrix} \\ \mathbf{C}_L &= \begin{bmatrix} 0 & 0 & 0.0035 & 7.7E-5 & 0.0009 \end{bmatrix} \\ \mathbf{D}_L &= \begin{bmatrix} 0.0009\beta ES_{k-1} & 0.0009\beta PS_{k-1} \end{bmatrix} \end{aligned} \tag{F.6}$$

with β the local amplification, and ES_{k-1} and PS_{k-1} the previous value of respectively the engine speed and the pump setting. Note that the amplification between each of the inputs and the machine speed (situated in matrix $\mathbf{B}_L$) depends on the machine state (ES_{k-1} and PS_{k-1}), and thus will need to be approximated if the model is used over a prediction horizon. The most convenient choice is to keep the matrix $\mathbf{B}_L$ constant over the horizon.

F.3 Density control on a combine harvester

The details of this application can be found in Chapter 11. Eqn (11.2) and (11.3) show the following state space model:

$$\begin{aligned}
&\textit{Force model:} \\
F_{B,k+1} &= F_{B,k} + K_{P_B,F_B}\gamma_F\left(T_{F,k} + u_{P,k}\right) \\
T_{F,k+1} &= (1-\gamma_F)\left(T_{F,k} + u_{P,k}\right) \\
P_{B,k+1} &= P_{B,k} + u_{P,k} \qquad \text{(F.7)} \\
&\textit{Pressure model:} \\
P_{B,k+1} &= P_{B,k} + K_{I_P,P_B}\gamma_P\left(T_{P,k} + u_{C,k,squared}\right) \\
T_{P,k+1} &= (1-\gamma_P)\left(T_{P,k} + u_{C,k,squared}\right) \\
C_{P,k+1,squared} &= C_{P,k,squared} + u_{C,k,squared} \qquad \text{(F.8)}
\end{aligned}$$

with $u_{P,k}$ and $u_{C,k,squared}$ respectively the change of pressure and the change of the squared current. K_{P_B,F_B} and K_{I_P,P_B} are the respective amplifications between pressure and force, and current and pressure. γ_F and γ_P are the parameter of the first order force and pressure model.

Force model

The above described force model can be written as:

$$\left\{\begin{array}{rcl} \mathbf{x}_{k+1} &=& \mathbf{A}_F\mathbf{x}_k + \mathbf{B}_F\mathbf{v}_k, \\ \mathbf{y}_k &=& \mathbf{C}_F\mathbf{x}_k + \mathbf{D}_F\mathbf{v}_k \end{array}\right. \qquad \text{(F.9)}$$

with system matrices $\mathbf{A}_F \in \mathbb{R}^{3\times3}$, $\mathbf{B}_F \in \mathbb{R}^{3\times1}$, $\mathbf{C}_F \in \mathbb{R}^{1\times3}$ and $\mathbf{D}_F \in \mathbb{R}^{1\times1}$. The state vector $\mathbf{x}_k$ now contains three elements: the force on the plunger $F_{B,k}$, an integrator state $T_{F,k}$ and the pressure on the bale chamber $P_{B,k}$. The input vector is $\mathbf{v}_k = [\Delta P_{B,k}]$ and the matrices $\mathbf{A}_F$, $\mathbf{B}_F$, $\mathbf{C}_F$ and $\mathbf{D}_F$ are:

$$\begin{aligned}
\mathbf{A}_F &= \begin{bmatrix} 1 & K_{P_B,F_B}\gamma_F & 0 \\ 0 & 1-\gamma_F & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
\mathbf{B}_F &= \begin{bmatrix} K_{P_B,F_B}\gamma_F \\ 1-\gamma_F \\ 1 \end{bmatrix} \qquad \text{(F.10)} \\
\mathbf{C}_F &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\
\mathbf{D}_F &= \begin{bmatrix} 0 \end{bmatrix}
\end{aligned}$$

Pressure model

The pressure model is very analogous. It can be described as:

$$\left\{\begin{array}{rcl} \mathbf{x}_{k+1} &=& \mathbf{A}_P\mathbf{x}_k + \mathbf{B}_P\mathbf{v}_k, \\ \mathbf{y}_k &=& \mathbf{C}_P\mathbf{x}_k + \mathbf{D}_P\mathbf{v}_k \end{array}\right. \qquad \text{(F.11)}$$

with system matrices $\mathbf{A}_P \in \mathbb{R}^{3\times3}$, $\mathbf{B}_P \in \mathbb{R}^{3\times1}$, $\mathbf{C}_P \in \mathbb{R}^{1\times3}$ and $\mathbf{D}_P \in \mathbb{R}^{1\times1}$. The state vector $\mathbf{x}_k$ now contains three elements: the pressure on the bale chamber $P_{B,k}$, an integrator state $T_{P,k}$ and the square of the current to the hydrostatic valve $C_{P,k,squared}$. The input vector is $\mathbf{v}_k = [\Delta C_{P,k,squared}]$ and the matrices $\mathbf{A}_P$, $\mathbf{B}_P$, $\mathbf{C}_P$ and $\mathbf{D}_P$ are:

$$\begin{aligned}
\mathbf{A}_P &= \begin{bmatrix} 1 & K_{I_P,P_B}\gamma_P & 0 \\ 0 & 1-\gamma_P & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
\mathbf{B}_P &= \begin{bmatrix} K_{I_P,P_B}\gamma_P \\ 1-\gamma_P \\ 1 \end{bmatrix} \\
\mathbf{C}_P &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\
\mathbf{D}_P &= \begin{bmatrix} 0 \end{bmatrix}
\end{aligned} \tag{F.12}$$

Appendix G

Field test references

20050810a. Run11, 10/08/2005. Wheat harvest; Step excitation.

20050810b. Run11, run14, run16, run18 & run24, 10/08/2005. Wheat harvest; Step excitation.

20050810c. Run4, run6, run8 & run10, 10/08/2005. Wheat harvest; Two different constant speeds, uphill & downhill.

20050817. Run0 - run10, 17/08/2005. Wheat harvest; Normal operation, used to determine the range of the process variables.

20051102. controllerrun21, controllerrun22, controllerrun23, controllerrun24 & controllerrun25, 2/11/2005. Corn harvest; PID feed rate control.

20051103a. controllerrun10, 3/11/2005. Corn harvest; PID feed rate control.

20051103b. controllerrun2, controllerrun3, controllerrun4, controllerrun6, controllerrun7, controllerrun8 & controllerrun9, 3/11/2005. Corn harvest; fuzzy feed rate control.

20051121a. Run4, 21/11/2005. Corn harvest; Step excitation.

20051121b. Run4, run5, run9 & run57, 21/11/2005. Corn harvest; Step excitation.

20051122a. controllerrun32, controllerrun36, controllerrun40, controllerrun42, controllerrun44, controllerrun45 & controllerrun46, 22/11/2005. Corn harvest; Feed rate control with different error weight factors.

20051122b. controllerrun16, controllerrun20, controllerrun36 & controllerrun38, 22/11/2005. Corn harvest; Feed rate control used for benchmarking.

20051123a. controllerrun85-controllerrun89, 23/11/2005. Corn harvest; Feed rate control with different horizon lengths.

20051123b. controllerrun37 & controllerrun65, 23/11/2005. Corn harvest; Feed rate control used for benchmarking.

20060224. Run1, 07/03/2006. Road test; Slope excitation on pump setting in road condition with engine speed at 2 100 rpm.

20060307a. Run00 & run01, 07/03/2006. Road test; Slope excitation on pump setting for two engine speeds in road condition.

20060307b. Run05, 07/03/2006. Road test; Step excitation on pump setting in road condition.

20060308. Run42, 08/03/2006. Road test; Slope excitation on engine speed in road condition with pump setting at 60%.

20060710. Run0, run1, run2, run3, run4, run5, run6, run7, run8, run9, run10 & run11, 10/07/2006. Barley harvest; Normal operation in one field to assess the gain variation over a field.

20060711a. Run4, run5, run6, run7, run8 & run9, 11/07/2006. Barley harvest; Sine excitation.

20060711b. Run10, 11/07/2006. Barley harvest; Step excitation.

20060711c. Run10-run14 & run16-run28, 11/07/2006. Barley harvest; Step excitations at different speeds.

20060713a. Run12, run15 & run16, 13/07/2006. Barley harvest; Sine excitation.

20060713b. Run1, run3, run7, run10 & run11, 13/07/2006. Barley harvest; Step excitations at different speeds.

20060905a. controllerrun35, 05/09/2006. Wheat harvest; Engine load control with Stochastic MPC.

20060905b. Run0 - run12, 05/09/2006. Wheat harvest; Normal operation, used to determine the range of the process variables.

20060906. controllerrun13, controllerrun20 & controllerrun23, 06/09/2006. Wheat harvest; Engine load control with different model parameters.

20061017. Run0-run12, 17/10/2006. Corn harvest; Normal operation, used to determine the range of the process variables.

20061030a. controllerrun17, controllerrun18, controllerrun19, controllerrun20 & controllerrun21, 30/10/2006. Corn harvest; Feed rate control with different pump setting - ground speed delays.

20061030b. Run6 - run14, 30/10/2006. Corn harvest; Normal operation, used to determine the range of the process variables.

20061107a. controllerrun11 & controllerrun20, 07/11/2006. Corn harvest; Feed rate controller with error dependent weight function.

20061107b. controllerrun34 & controllerrun35, 07/11/2006. Corn harvest; Feed rate control with two different model update rates.

20061107c. controllerrun78, 07/11/2006. Corn harvest; Feed rate control with Stochastic MPC.

20061109b. controllerrun122, 09/11/2006. Corn harvest; Grain mass flow control with Stochastic MPC.

20061109c. controllerrun58, 09/11/2006. Corn harvest; Feed rate control & engine load constraint with Stochastic MPC.

20070811a. Balerdata_19, 11/08/2007. Wheat baling; Step excitation of the density system.

20070811b. Balerdata_12, balerdata_13, balerdata_14, balerdata_16, balerdata_18, balerdata_19, balerdata_37, balerdata_43, balerdata_44, balerdata_45, balerdata_47, balerdata_49, balerdata_50, balerdata_51 & balerdata_52, 11/08/2007. Wheat baling; Stationary at different PWM levels.

20080613. sim_controllerrun17, sim_controllerrun18 & sim_controllerrun19, 13/06/2008. Simulation; Feed rate control with different model gains.

20080731. 2008_7_31_17_37_15.mat, 31/07/2008. Wheat baling; Stochastic MPC active. Speed of the tractor decreased halfway through the run.

Index

Active Set Method
    computation speed, 149
    eliminate equality constraints, 144
    rotational symmetry, 146
    stopping criterion, 147
agricultural machinery
    commercial context, 32
    economical context, 29
    environmental context, 28
    historical context, 27
    social context, 29
agricultural products
    demand, 30
Agritechnica, 10, 42, 201, 205
applications
    capacity control, 48, 86, 111, 126, 130
    cruise control, 153
    density control, 167

baler, square, 167
benchmarking, 108, 149, 209
biofuels, 30

Case New Holland, ii, 4, 39, 53, 71, 163, 167, 182, 251
Claas, 39
combine harvester
    cleaning, 36
    crop collection, 34
    road map, 43
    sensors, 37
    separation, 36
    threshing, 34
    working principle, 33
constraints, 93
    feasiblity, 106
control in agriculture
    feed rate, 43
    guidance, 43
    road map, 43
crop throughput, 38

dSpace, 207

feasibility, 148
    slack variables, 148
fertilizer, 28
field tests
    benchmark, 108
    nonlinear MPC, 163
    Stochastic MPC, 130, 177
fuzzy control, 108

gold medal, 42
grain mass flow, 40
    impact plate-based, 40
    torque-based, 40
    volume-based, 40
grain price, 31
grain quality, 42

harvest rate, 38
history, 12, 14, 27
horizon, 16, 94

IntelliCruise, 10
IWT, ii, 201, 251

John Deere, 39, 43

Linear Programming, 149
losses, 41
LP, *see* Linear Programming

Mathworks
    Matlab, 207
measurements
    actual cutting width, 38

crop moisture, 39
engine load, 38
feed rate, 38
grain mass flow, 40
layer thickness, 39
losses, 41
machine speed, 38
MOG, 42
threshing torque, 39
model update, 221
computational load, 224
procedure, 221
modelstructures
model uncertainty, 23, 119, 123, 126
MOG, 42
moisture content, 39
MPC
adaptive model, 104
constraints, 17, 93, 106
delays, 102
dynamics, 102
error dependent objective, 97
explicit MPC, 143
feasibility, 19, 106, 148
general, 15
horizon, 94
model, 17, 93
multiple models, 26
objective function, 16, 94
objective function weight factors, 96
robust MPC, 24
stability, 21
Stochastic MPC, 113

National Instruments, 207
LabView, 207, 213
PXI, 208
Nonlinear MPC, 22, 161
norm-bounded uncertainty, 23

parameter uncertainty, 237
patents, 43, 205
performance measures, 233
pesticide, 28
PID control, 108
polytope uncertainty, 23
productivity, 27

QP, *see* Quadratic Programming
Quadratic Programming
Active Set Method, 144
equality-constrained, 139
inequality constrained, 140
Interior-Point Method, 149
Quadratic Programmning, 138

sampling rates
control rate, 99
model sampling rate, 100
model update rate, 105, 221, *see* model update
parameter update rate, 104
scale size, 30
silver medal, 42, 201, 205
Softing, 213
square baler, *see* baler, square
Stochastic MPC, 113
analysis, 124
combine harvester, 128, 130
constraints, 123
literature, 113
motivation, 113
objective function, 119
square baler, 167
uncertainty model, 119, 123

Acknowledgements

Tom Coen is supported by a scholarship of IWT-Flanders (Institute for the Promotion of Innovation through Science and Technology — IWT Vlaanderen W, IWT (2008)).

Tests on farming machinery are not possible without the support of a major manufacturer as Case New Holland (W, Case New Holland (2008)). Special thanks go to CNH for making combine harvesters (a New Holland CR and a New Holland CX) and a baler combination available for the countless field tests in Leuven. CNH also made baler field tests possible throughout Europe, which considerably extended the testing season.

It cannot be stressed enough how much the implementation of this PhD has been a team effort. This research would not have been possible without the support of the technical staff at KULeuven, especially Dirk Leroy, and the people at CNH, where special thanks go to Bart Missotten.

