# Identification in the frequency domain

## Improving the numerical stability

Adhemar Bultheel

Department of Computer Science
K.U.Leuven

CIMAC, La Laguna (Tenerife), 6-9 May 2008

# Summary

- Identification problem
- Orthogonal Rational Functions
- Orthogonal Vector Polynomials

Stable Linear Algebra techniques
to improve numerical stability

# Summary

- Identification problem
- Orthogonal Rational Functions
- Orthogonal Vector Polynomials

Stable Linear Algebra techniques
to improve numerical stability

# Summary

- Identification problem
- Orthogonal Rational Functions
- Orthogonal Vector Polynomials

Stable Linear Algebra techniques
to improve numerical stability

# Summary

- Identification problem
- Orthogonal Rational Functions
- Orthogonal Vector Polynomials

Stable Linear Algebra techniques
to improve numerical stability

# Summary

- Identification problem
- Orthogonal Rational Functions
- Orthogonal Vector Polynomials

Stable Linear Algebra techniques
to improve numerical stability

# Definition of Identification

- given noisy data
- find some model for the system
- by optimizing some cost function

# Definition of Identification

- given noisy data
- find some model for the system
- by optimizing some cost function

# Definition of Identification

- given noisy data
- find some model for the system
- by optimizing some cost function

# The story of Ceres





Piazzi

Gauss





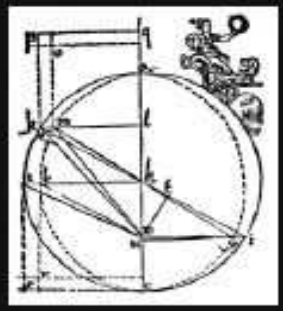
Piazzi

Gauss



Keppler







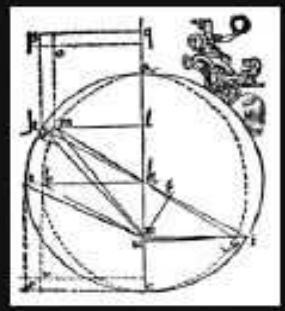
Piazzi

Gauss



Keppler







Piazzi



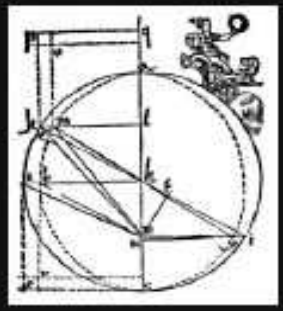
Von Zach

Cost



Model







Data



Validation

# Noise and model errors

- Noise error dominates
  - model and excitation loosely linked
  - cost function set by noise properties (ML)
- Model error dominates

# Noise and model errors

- Noise error dominates
  - model and excitation loosely linked
  - cost function set by noise properties (ML)
- Model error dominates
  - model and excitation closely related
  - design of experiment is important

# Noise and model errors

- Noise error dominates
  - model and excitation loosely linked
  - cost function set by noise properties (ML)
- Model error dominates
  - model and excitation closely related
  - design of experiment is important

# Noise and model errors

- Noise error dominates
  - model and excitation loosely linked
  - cost function set by noise properties (ML)
- Model error dominates
  - model and excitation closely related
  - design of experiment is important

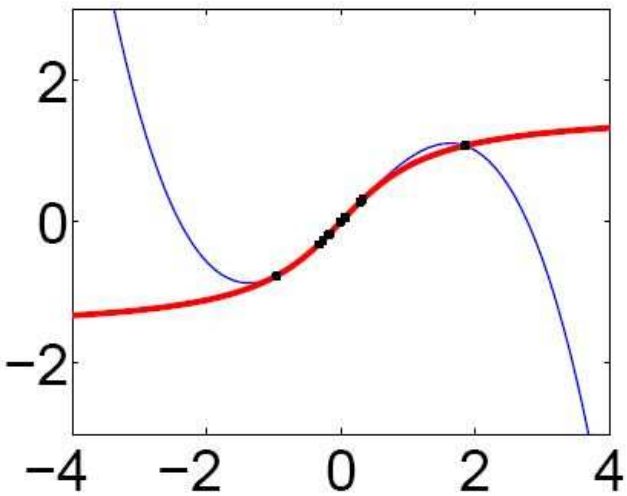
# Noise and model errors

- Noise error dominates
  - model and excitation loosely linked
  - cost function set by noise properties (ML)
- Model error dominates
  - model and excitation closely related
  - design of experiment is important

# Noise and model errors

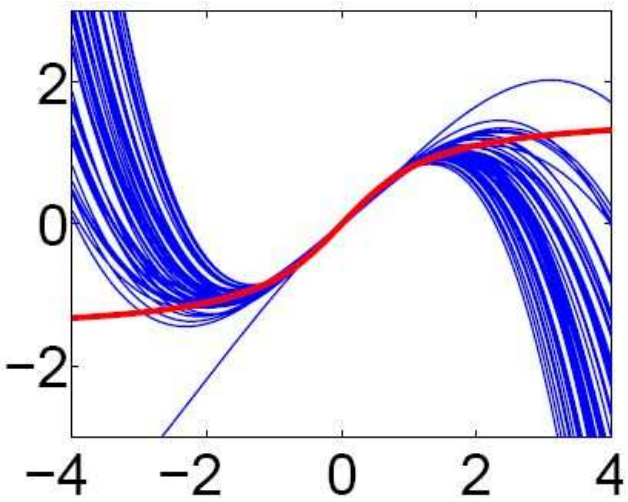
- Noise error dominates
  - model and excitation loosely linked
  - cost function set by noise properties (ML)
- Model error dominates
  - model and excitation closely related
  - design of experiment is important

# Example: model errors



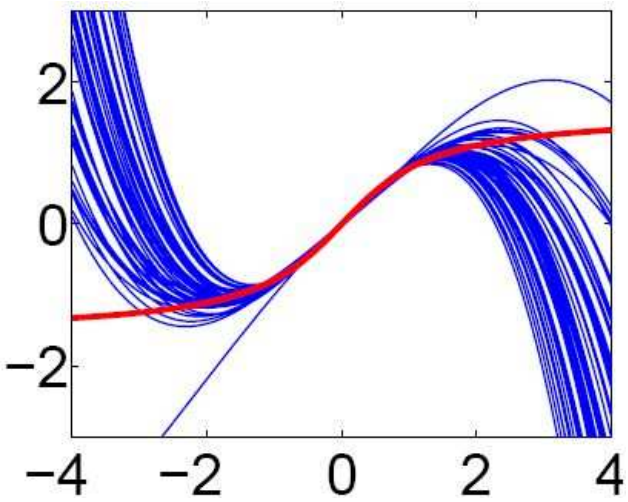

$N(0, \sigma = 1)$

# Example: model errors

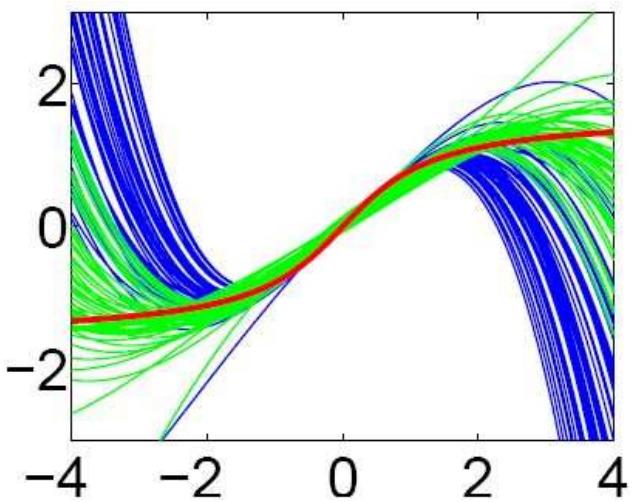



$N(0, \sigma = 1)$

# Example: model errors




$N(0, \sigma = 1)$




$N(0, \sigma = 2)$

# Time domain or frequency domain

We consider continuous/discrete time systems

$y$ is output

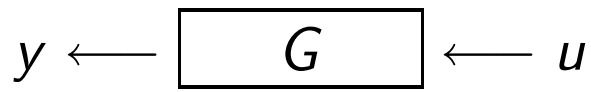
$Y(z) = \mathcal{L}(y(t))$

$Y(z) = \sum_k y_k z^{-k}$

$u$ is input

$U(z) = \mathcal{L}(u(t))$

$U(z) = \sum_k u_k z^{-k}$




$$Y(z) = G(z)U(z)$$

$G = Y/U$ transfer function

$G(z)$, $z = j\omega$ (or $z = e^{j\omega}$ for d.t.)
is FRF (frequency response function)

# Time domain or frequency domain

We consider continuous/discrete time systems

$y$ is output
$Y(z) = \mathcal{L}(y(t))$
$Y(z) = \sum_k y_k z^{-k}$

$u$ is input
$U(z) = \mathcal{L}(u(t))$
$U(z) = \sum_k u_k z^{-k}$




$$Y(z) = G(z)U(z)$$

$G = Y/U$ transfer function

$G(z)$, $z = j\omega$ (or $z = e^{j\omega}$ for d.t.)
is FRF (frequency response function)

# Time domain or frequency domain

We consider continuous/discrete time systems

$y$ is output
$Y(z) = \mathcal{L}(y(t))$
$Y(z) = \sum_k y_k z^{-k}$

$u$ is input
$U(z) = \mathcal{L}(u(t))$
$U(z) = \sum_k u_k z^{-k}$




$$Y(z) = G(z)U(z)$$

$G = Y/U$ transfer function

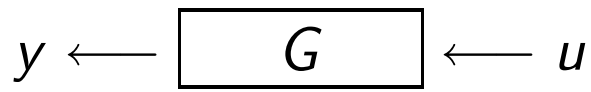
$G(z)$, $z = j\omega$ (or $z = e^{j\omega}$ for d.t.)
is FRF (frequency response function)

# Time domain or frequency domain

We consider continuous/discrete time systems

$y$ is output
$Y(z) = \mathcal{L}(y(t))$
$Y(z) = \sum_k y_k z^{-k}$

$u$ is input
$U(z) = \mathcal{L}(u(t))$
$U(z) = \sum_k u_k z^{-k}$




$$Y(z) = G(z)U(z)$$

$G = Y/U$ transfer function

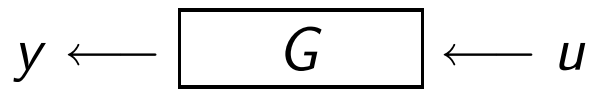
$G(z)$, $z = j\omega$ (or $z = e^{j\omega}$ for d.t.)
is FRF (frequency response function)

# Time domain or frequency domain

We consider continuous/discrete time systems

$y$ is output
$Y(z) = \mathcal{L}(y(t))$
$Y(z) = \sum_k y_k z^{-k}$

$u$ is input
$U(z) = \mathcal{L}(u(t))$
$U(z) = \sum_k u_k z^{-k}$




$$Y(z) = G(z)U(z)$$

$G = Y/U$ transfer function

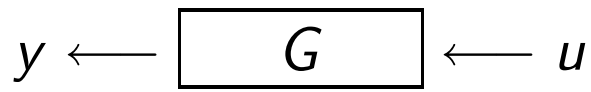
$G(z)$, $z = j\omega$ (or $z = e^{j\omega}$ for d.t.)
is FRF (frequency response function)

# Time domain or frequency domain

We consider continuous/discrete time systems

$y$ is output
$Y(z) = \mathcal{L}(y(t))$
$Y(z) = \sum_k y_k z^{-k}$

$u$ is input
$U(z) = \mathcal{L}(u(t))$
$U(z) = \sum_k u_k z^{-k}$




$$Y(z) = G(z)U(z)$$

$G = Y/U$ transfer function

$G(z)$, $z = j\omega$ (or $z = e^{j\omega}$ for d.t.)
is FRF (frequency response function)

# Time domain or frequency domain

**DATA** MODEL COST

time-frequency transform
Discrete Fourier transform (DFT)
Laplace transform

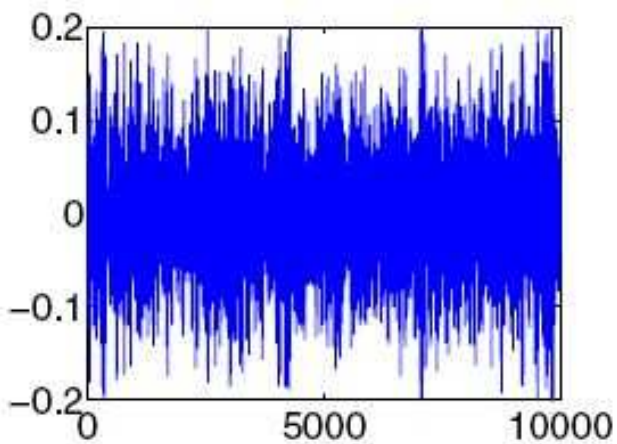
time domain

$$u(t) = u_0(t) + n_u(t)$$
$$y(t) = y_0(t) + n_y(t)$$

$\Leftrightarrow$

frequency domain

$$U(z_k) = U_0(z_k) + N_U(z_k)$$
$$Y(z_k) = Y_0(z_k) + N_Y(z_k)$$



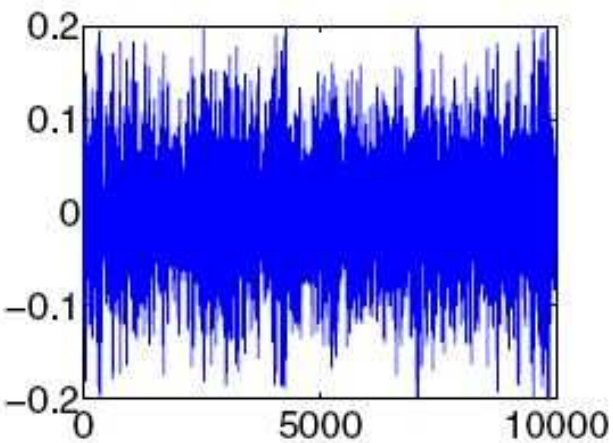
# Time domain or frequency domain

**DATA** MODEL COST

time-frequency transform
Discrete Fourier transform (DFT)
Laplace transform

time domain

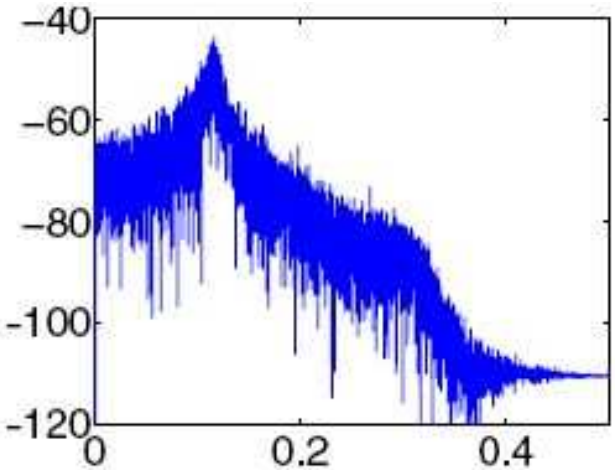
$$u(t) = u_0(t) + n_u(t)$$
$$y(t) = y_0(t) + n_y(t)$$




$\Leftrightarrow$

frequency domain

$$U(z_k) = U_0(z_k) + N_U(z_k)$$
$$Y(z_k) = Y_0(z_k) + N_Y(z_k)$$



# Time domain or frequency domain

**DATA** MODEL COST

time-frequency transform

Discrete Fourier transform (DFT)

Laplace transform

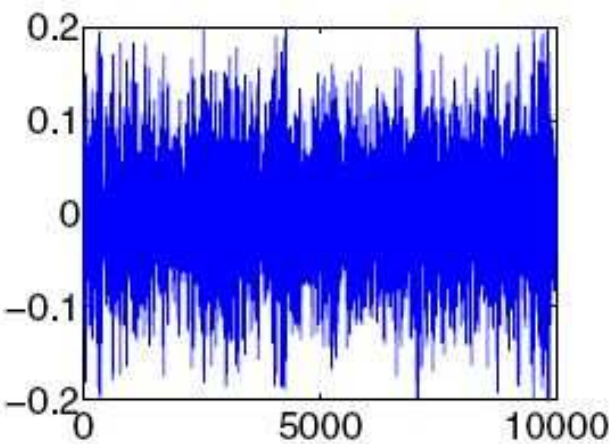
time domain

$$u(t) = u_0(t) + n_u(t)$$
$$y(t) = y_0(t) + n_y(t)$$

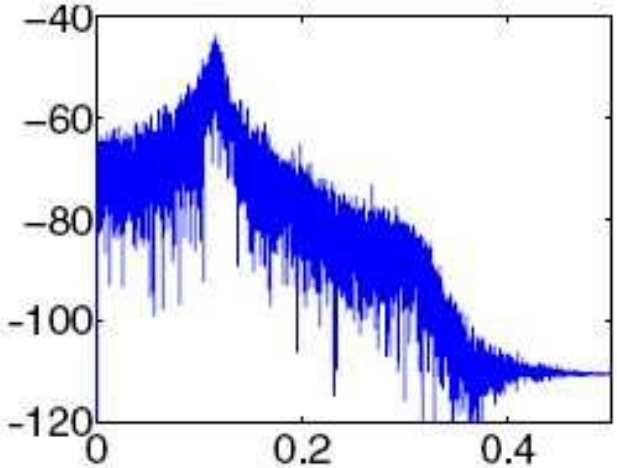
$\Leftrightarrow$

frequency domain

$$U(z_k) = U_0(z_k) + N_U(z_k)$$
$$Y(z_k) = Y_0(z_k) + N_Y(z_k)$$





# Discrete time vs continuous time

DATA **MODEL** COST

time domain

frequency domain

$A(z)Y_0(z) = B(z)U_0(z)$

ct: $A(D)y_0(t) = B(D)u_0(t)$ $\Leftrightarrow$ ct: $z = j\omega$

$D = \frac{d}{dt}$

dt: $A(\Delta)y_0(t) = B(\Delta)u_0(t)$ dt: $z = e^{j2\pi\omega/\omega_s}$

$\Delta y(t) = y(t - T_s)$

# Discrete time vs continuous time

DATA **MODEL** COST

| time domain | | frequency domain |
|---|---|---|
| | | $A(z)Y_0(z) = B(z)U_0(z)$ |
| ct: $A(D)y_0(t) = B(D)u_0(t)$<br>$D = \frac{d}{dt}$ | $\Leftrightarrow$ | ct: $z = j\omega$ |
| dt: $A(\Delta)y_0(t) = B(\Delta)u_0(t)$<br>$\Delta y(t) = y(t - T_s)$ | | dt: $z = e^{j2\pi\omega/\omega_s}$ |

# Discrete time vs continuous time

DATA **MODEL** COST

| time domain | | frequency domain |
|---|---|---|
| | | $A(z)Y_0(z) = B(z)U_0(z)$ |
| ct: $A(D)y_0(t) = B(D)u_0(t)$<br>$D = \frac{d}{dt}$ | $\Leftrightarrow$ | ct: $z = j\omega$ |
| dt: $A(\Delta)y_0(t) = B(\Delta)u_0(t)$<br>$\Delta y(t) = y(t - T_s)$ | | dt: $z = e^{j2\pi\omega/\omega_s}$ |



# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$
current: $u(k) = u_0(k) + n_u(k)$ , $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

## LEAST SQUARES

- $\mathbf{u}R = \mathbf{y}$, $\mathbf{u} = [u(1), \ldots, u(N)]^T$, $\mathbf{y} = [y(1), \ldots, y(N)]^T$
- $\mathbf{e} = \mathbf{u}R - \mathbf{y}$ (residual)
- $\mathbf{u}^T\mathbf{u}R = \mathbf{u}^T\mathbf{y}$ (normal equations minimizes $\mathbf{e}^T\mathbf{e}$)
- $R_{LS} = \dfrac{\mathbf{u}^T\mathbf{y}}{\mathbf{u}^T\mathbf{u}} = \dfrac{\frac{1}{N}\sum_k y(k)u(k)}{\frac{1}{N}\sum_k u(k)^2}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$
current: $u(k) = u_0(k) + n_u(k)$ , $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**LEAST SQUARES**

- $\mathbf{u}R = \mathbf{y}$, $\mathbf{u} = [u(1), \ldots, u(N)]^T$, $\mathbf{y} = [y(1), \ldots, y(N)]^T$
- $\mathbf{e} = \mathbf{u}R - \mathbf{y}$ (residual)
- $\mathbf{u}^T\mathbf{u}R = \mathbf{u}^T\mathbf{y}$ (normal equations minimizes $\mathbf{e}^T\mathbf{e}$)
- $R_{LS} = \dfrac{\mathbf{u}^T\mathbf{y}}{\mathbf{u}^T\mathbf{u}} = \dfrac{\frac{1}{N}\sum_k y(k)u(k)}{\frac{1}{N}\sum_k u(k)^2}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**LEAST SQUARES**

- $\mathbf{u}R = \mathbf{y}$, $\mathbf{u} = [u(1), \ldots, u(N)]^T$, $\mathbf{y} = [y(1), \ldots, y(N)]^T$
- $\mathbf{e} = \mathbf{u}R - \mathbf{y}$ (residual)
- $\mathbf{u}^T\mathbf{u}R = \mathbf{u}^T\mathbf{y}$ (normal equations minimizes $\mathbf{e}^T\mathbf{e}$)
- $R_{LS} = \dfrac{\mathbf{u}^T\mathbf{y}}{\mathbf{u}^T\mathbf{u}} = \dfrac{\frac{1}{N}\sum_k y(k)u(k)}{\frac{1}{N}\sum_k u(k)^2}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**LEAST SQUARES**

- $\mathbf{u}R = \mathbf{y}$, $\mathbf{u} = [u(1), \ldots, u(N)]^T$, $\mathbf{y} = [y(1), \ldots, y(N)]^T$
- $\mathbf{e} = \mathbf{u}R - \mathbf{y}$ (residual)
- $\mathbf{u}^T\mathbf{u}R = \mathbf{u}^T\mathbf{y}$ (normal equations minimizes $\mathbf{e}^T\mathbf{e}$)
- $R_{LS} = \dfrac{\mathbf{u}^T\mathbf{y}}{\mathbf{u}^T\mathbf{u}} = \dfrac{\frac{1}{N}\sum_k y(k)u(k)}{\frac{1}{N}\sum_k u(k)^2}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**LEAST SQUARES**

- $\mathbf{u}R = \mathbf{y}$, $\mathbf{u} = [u(1), \ldots, u(N)]^T$, $\mathbf{y} = [y(1), \ldots, y(N)]^T$
- $\mathbf{e} = \mathbf{u}R - \mathbf{y}$ (residual)
- $\mathbf{u}^T\mathbf{u}R = \mathbf{u}^T\mathbf{y}$ (normal equations minimizes $\mathbf{e}^T\mathbf{e}$)
- $R_{LS} = \dfrac{\mathbf{u}^T\mathbf{y}}{\mathbf{u}^T\mathbf{u}} = \dfrac{\frac{1}{N}\sum_k y(k)u(k)}{\frac{1}{N}\sum_k u(k)^2}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**LEAST SQUARES**

- $\mathbf{u}R = \mathbf{y}$, $\mathbf{u} = [u(1), \ldots, u(N)]^T$, $\mathbf{y} = [y(1), \ldots, y(N)]^T$
- $\mathbf{e} = \mathbf{u}R - \mathbf{y}$ (residual)
- $\mathbf{u}^T\mathbf{u}R = \mathbf{u}^T\mathbf{y}$ (normal equations minimizes $\mathbf{e}^T\mathbf{e}$)
- $R_{LS} = \dfrac{\mathbf{u}^T\mathbf{y}}{\mathbf{u}^T\mathbf{u}} = \dfrac{\frac{1}{N}\sum_k y(k)u(k)}{\frac{1}{N}\sum_k u(k)^2}$



Gauss 1809

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$
current: $u(k) = u_0(k) + n_u(k)$ , $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**ML for ERRORS IN VARIABLES**

- estimate $R, u_0, y_0$ by max $pdf(u, y|R, u_0, y_0)$, with $y_0 = u_0 R$
- $n_u = u - u_0$ and $n_y = y - y_0$ uncorrelated
- max pdf $\sim \exp(-\frac{(u-u_0)^2}{\sigma_u^2}) \exp(-\frac{(y-y_0)^2}{\sigma_y^2})$
- min $\frac{(u-u_0)^2}{\sigma_u^2} + \frac{(y-y_0)^2}{\sigma_y^2}$
- $u_0 = \frac{1}{N}\sum_k u(k)$, $y_0 = \frac{1}{N}\sum_k y(k)$, $R_{EV} = \frac{y_0}{u_0} = \frac{\frac{1}{N}\sum_k y(k)}{\frac{1}{N}\sum_k u(k)}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**ML for ERRORS IN VARIABLES**

- estimate $R, u_0, y_0$ by max $pdf(u, y|R, u_0, y_0)$, with $y_0 = u_0 R$
- $n_u = u - u_0$ and $n_y = y - y_0$ uncorrelated
- max pdf $\sim \exp(-\frac{(u-u_0)^2}{\sigma_u^2}) \exp(-\frac{(y-y_0)^2}{\sigma_y^2})$
- min $\frac{(u-u_0)^2}{\sigma_u^2} + \frac{(y-y_0)^2}{\sigma_y^2}$
- $u_0 = \frac{1}{N}\sum_k u(k)$, $y_0 = \frac{1}{N}\sum_k y(k)$, $R_{EV} = \frac{y_0}{u_0} = \frac{\frac{1}{N}\sum_k y(k)}{\frac{1}{N}\sum_k u(k)}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**ML for ERRORS IN VARIABLES**

- estimate $R, u_0, y_0$ by max $pdf(u, y|R, u_0, y_0)$, with $y_0 = u_0 R$
- $n_u = u - u_0$ and $n_y = y - y_0$ uncorrelated
- max pdf $\sim \exp(-\frac{(u-u_0)^2}{\sigma_u^2}) \exp(-\frac{(y-y_0)^2}{\sigma_y^2})$
- min $\frac{(u-u_0)^2}{\sigma_u^2} + \frac{(y-y_0)^2}{\sigma_y^2}$
- $u_0 = \frac{1}{N}\sum_k u(k)$, $y_0 = \frac{1}{N}\sum_k y(k)$, $R_{EV} = \frac{y_0}{u_0} = \frac{\frac{1}{N}\sum_k y(k)}{\frac{1}{N}\sum_k u(k)}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**ML for ERRORS IN VARIABLES**

- estimate $R, u_0, y_0$ by max $pdf(u, y|R, u_0, y_0)$, with $y_0 = u_0 R$
- $n_u = u - u_0$ and $n_y = y - y_0$ uncorrelated
- max pdf $\sim \exp(-\frac{(u-u_0)^2}{\sigma_u^2}) \exp(-\frac{(y-y_0)^2}{\sigma_y^2})$
- min $\frac{(u-u_0)^2}{\sigma_u^2} + \frac{(y-y_0)^2}{\sigma_y^2}$
- $u_0 = \frac{1}{N}\sum_k u(k)$, $y_0 = \frac{1}{N}\sum_k y(k)$, $R_{EV} = \frac{y_0}{u_0} = \frac{\frac{1}{N}\sum_k y(k)}{\frac{1}{N}\sum_k u(k)}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**ML for ERRORS IN VARIABLES**

- estimate $R, u_0, y_0$ by max $pdf(u, y|R, u_0, y_0)$, with $y_0 = u_0 R$
- $n_u = u - u_0$ and $n_y = y - y_0$ uncorrelated
- max pdf $\sim \exp(-\frac{(u-u_0)^2}{\sigma_u^2}) \exp(-\frac{(y-y_0)^2}{\sigma_y^2})$
- min $\frac{(u-u_0)^2}{\sigma_u^2} + \frac{(y-y_0)^2}{\sigma_y^2}$
- $u_0 = \frac{1}{N}\sum_k u(k)$, $y_0 = \frac{1}{N}\sum_k y(k)$, $R_{EV} = \frac{y_0}{u_0} = \frac{\frac{1}{N}\sum_k y(k)}{\frac{1}{N}\sum_k u(k)}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**ML for ERRORS IN VARIABLES**

- estimate $R, u_0, y_0$ by max $pdf(u, y|R, u_0, y_0)$, with $y_0 = u_0 R$
- $n_u = u - u_0$ and $n_y = y - y_0$ uncorrelated
- max pdf $\sim \exp(-\frac{(u-u_0)^2}{\sigma_u^2}) \exp(-\frac{(y-y_0)^2}{\sigma_y^2})$
- min $\frac{(u-u_0)^2}{\sigma_u^2} + \frac{(y-y_0)^2}{\sigma_y^2}$
- $u_0 = \frac{1}{N}\sum_k u(k)$, $y_0 = \frac{1}{N}\sum_k y(k)$, $R_{EV} = \frac{y_0}{u_0} = \frac{\frac{1}{N}\sum_k y(k)}{\frac{1}{N}\sum_k u(k)}$

# Example: a resistor



DATA MODEL **COST**

data: voltage: $y(k) = y_0(k) + n_y(k)$
current: $u(k) = u_0(k) + n_u(k)$ , $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$

**ML for ERRORS IN VARIABLES**

- estimate $R, u_0, y_0$ by max $pdf(u, y|R, u_0, y_0)$, with $y_0 = u_0 R$
- $n_u = u - u_0$ and $n_y = y - y_0$ uncorrelated
- max pdf $\sim \exp(-\frac{(u-u_0)^2}{\sigma_u^2}) \exp(-\frac{(y-y_0)^2}{\sigma_y^2})$
- min $\frac{(u-u_0)^2}{\sigma_u^2} + \frac{(y-y_0)^2}{\sigma_y^2}$
- $u_0 = \frac{1}{N}\sum_k u(k)$, $y_0 = \frac{1}{N}\sum_k y(k)$, $R_{EV} = \frac{y_0}{u_0} = \frac{\frac{1}{N}\sum_k y(k)}{\frac{1}{N}\sum_k u(k)}$



Adcock 1877

# LS vs EV=OR=TLS

DATA MODEL **COST**

**EXAMPLE LINEAR REGRESSION** $DP = Y$

**LEAST SQUARES**

Smallest $\Delta Y$ such that
$XP = Y + \Delta Y$

**TOTAL LEAST SQUARES**

Smallest $\Delta X, \Delta Y$ such that
$(X + \Delta X)P = Y + \Delta Y$

if $\sigma_y^2 = \sigma_x^2$ then OR = TLS =
ML for EV

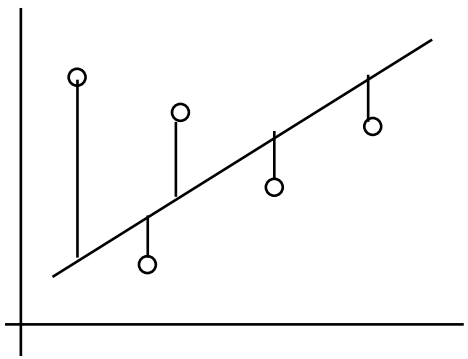
# LS vs EV=OR=TLS

DATA MODEL **COST**

## EXAMPLE LINEAR REGRESSION $DP = Y$

### LEAST SQUARES

Smallest $\Delta Y$ such that
$XP = Y + \Delta Y$



### TOTAL LEAST SQUARES

Smallest $\Delta X, \Delta Y$ such that
$(X + \Delta X)P = Y + \Delta Y$

if $\sigma_y^2 = \sigma_x^2$ then OR = TLS = ML for EV

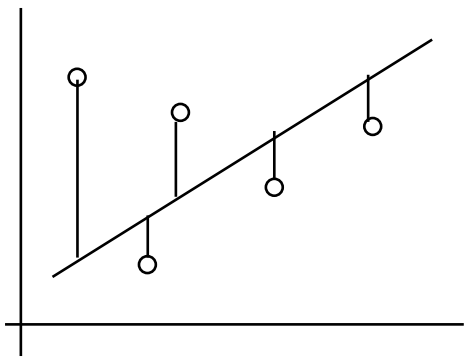
# LS vs EV=OR=TLS

DATA MODEL **COST**

**EXAMPLE LINEAR REGRESSION** $DP = Y$

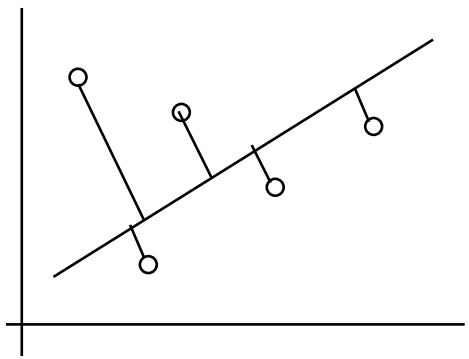
## LEAST SQUARES

Smallest $\Delta Y$ such that
$XP = Y + \Delta Y$



## TOTAL LEAST SQUARES

Smallest $\Delta X, \Delta Y$ such that
$(X + \Delta X)P = Y + \Delta Y$



if $\sigma_y^2 = \sigma_x^2$ then OR = TLS = ML for EV

# Example: a resistor



data : voltage: $y(k) = y_0(k) + n_y(k)$
current: $u(k) = u_0(k) + n_u(k)$ , $k = 1, \ldots, N$

model : $y_0(k) = Ru_0(k)$

$\hat{R}$

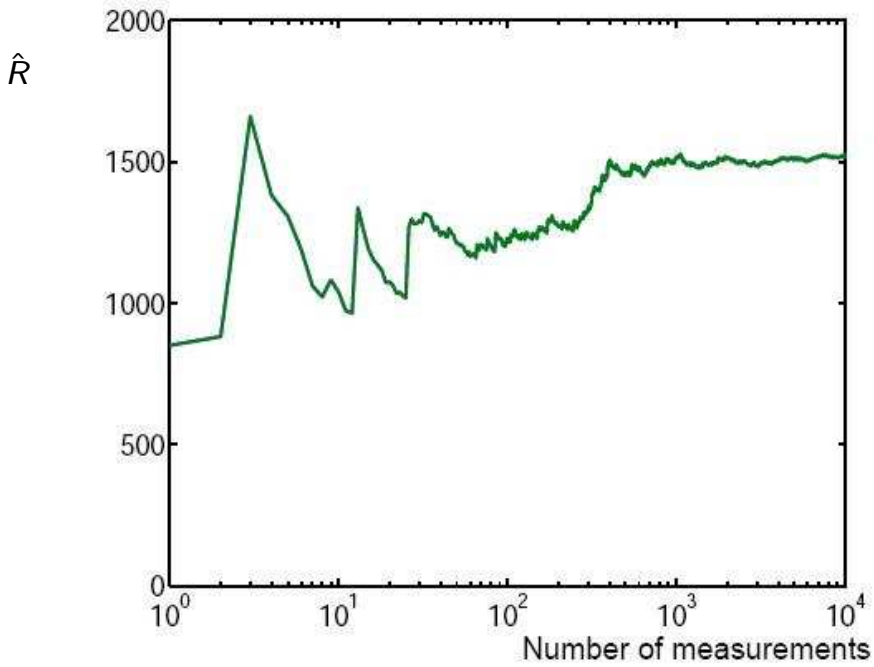
**cost**

# Example: a resistor



data : voltage: $y(k) = y_0(k) + n_y(k)$
current: $u(k) = u_0(k) + n_u(k)$ , $k = 1, \ldots, N$

model : $y_0(k) = Ru_0(k)$




**cost**

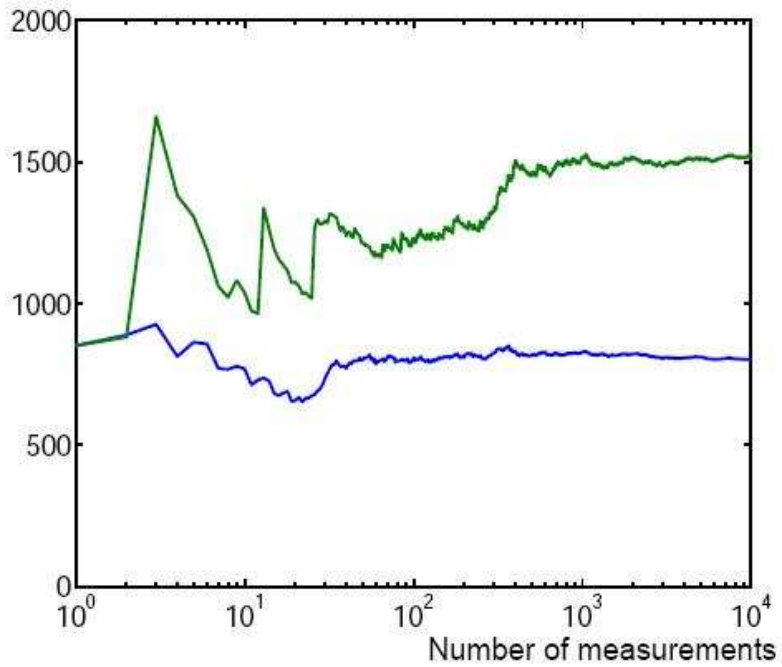
$$\hat{R}_{SA}(N) = \frac{1}{N} \sum_{k=1}^{N} \frac{y(k)}{u(k)}$$

# Example: a resistor



data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$




**cost**

$$\hat{R}_{SA}(N) = \frac{1}{N}\sum_{k=1}^{N}\frac{y(k)}{u(k)}$$

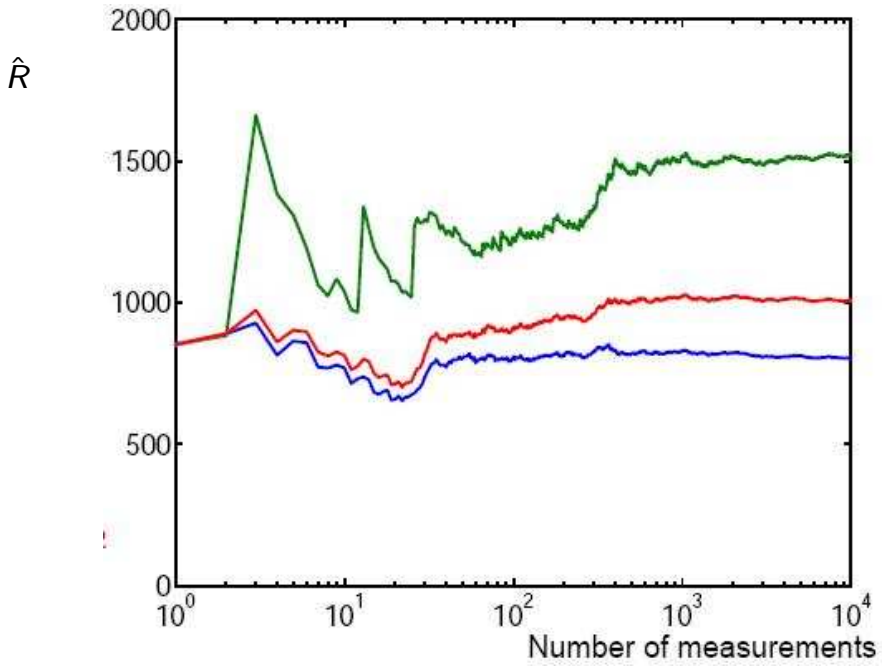
$$\hat{R}_{LS}(N) = \frac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$$

# Example: a resistor



data: voltage: $y(k) = y_0(k) + n_y(k)$, current: $u(k) = u_0(k) + n_u(k)$, $k = 1, \ldots, N$

model: $y_0(k) = Ru_0(k)$




## cost

$$\hat{R}_{SA}(N) = \frac{1}{N}\sum_{k=1}^{N}\frac{y(k)}{u(k)}$$

$$\hat{R}_{LS}(N) = \frac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$$

$$\hat{R}_{EV}(N) = \frac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$$

# Analysis of cost

- $u(k) = u_0 + n_u(k)$ and $y(k) = y_0 + n_y(k)$
- $n_u(k)$ and $n_y(k)$ zero mean i.i.d. $\sigma_u^2$, $\sigma_y^2$
- $\hat{R}_{LS}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$
- $\lim_{N\to\infty} \hat{R}_{LS}(N) = \cdots = R_0 \dfrac{1}{1+\sigma_u^2/u_0^2}$
- Biased! unless $\sigma_i = 0$
- $\hat{R}_{EV}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$
- $\lim_{N\to\infty} \hat{R}_{EV}(N) = \cdots = R_0$
- Unbiased!

# Analysis of cost

- $u(k) = u_0 + n_u(k)$ and $y(k) = y_0 + n_y(k)$
- $n_u(k)$ and $n_y(k)$ zero mean i.i.d. $\sigma_u^2$, $\sigma_y^2$
- $\hat{R}_{LS}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$
- $\lim_{N\to\infty} \hat{R}_{LS}(N) = \cdots = R_0 \dfrac{1}{1+\sigma_u^2/u_0^2}$
- Biased! unless $\sigma_i = 0$
- $\hat{R}_{EV}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$
- $\lim_{N\to\infty} \hat{R}_{EV}(N) = \cdots = R_0$
- Unbiased!

# Analysis of cost

- $u(k) = u_0 + n_u(k)$ and $y(k) = y_0 + n_y(k)$
- $n_u(k)$ and $n_y(k)$ zero mean i.i.d. $\sigma_u^2$, $\sigma_y^2$
- $\hat{R}_{LS}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$
- $\lim_{N\to\infty} \hat{R}_{LS}(N) = \cdots = R_0 \dfrac{1}{1+\sigma_u^2/u_0^2}$
- Biased! unless $\sigma_i = 0$
- $\hat{R}_{EV}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$
- $\lim_{N\to\infty} \hat{R}_{EV}(N) = \cdots = R_0$
- Unbiased!

# Analysis of cost

- $u(k) = u_0 + n_u(k)$ and $y(k) = y_0 + n_y(k)$
- $n_u(k)$ and $n_y(k)$ zero mean i.i.d. $\sigma_u^2$, $\sigma_y^2$
- $\hat{R}_{LS}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$
- $\lim_{N\to\infty} \hat{R}_{LS}(N) = \cdots = R_0 \dfrac{1}{1+\sigma_u^2/u_0^2}$
- Biased! unless $\sigma_i = 0$
- $\hat{R}_{EV}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$
- $\lim_{N\to\infty} \hat{R}_{EV}(N) = \cdots = R_0$
- Unbiased!

# Analysis of cost

- $u(k) = u_0 + n_u(k)$ and $y(k) = y_0 + n_y(k)$
- $n_u(k)$ and $n_y(k)$ zero mean i.i.d. $\sigma_u^2$, $\sigma_y^2$
- $\hat{R}_{LS}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$
- $\lim_{N\to\infty} \hat{R}_{LS}(N) = \cdots = R_0 \dfrac{1}{1+\sigma_u^2/u_0^2}$
- Biased! unless $\sigma_i = 0$
- $\hat{R}_{EV}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$
- $\lim_{N\to\infty} \hat{R}_{EV}(N) = \cdots = R_0$
- Unbiased!

# Analysis of cost

- $u(k) = u_0 + n_u(k)$ and $y(k) = y_0 + n_y(k)$
- $n_u(k)$ and $n_y(k)$ zero mean i.i.d. $\sigma_u^2$, $\sigma_y^2$
- $\hat{R}_{LS}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$
- $\lim_{N\to\infty} \hat{R}_{LS}(N) = \cdots = R_0 \dfrac{1}{1+\sigma_u^2/u_0^2}$
- Biased! unless $\sigma_i = 0$
- $\hat{R}_{EV}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$
- $\lim_{N\to\infty} \hat{R}_{EV}(N) = \cdots = R_0$
- Unbiased!

# Analysis of cost

- $u(k) = u_0 + n_u(k)$ and $y(k) = y_0 + n_y(k)$
- $n_u(k)$ and $n_y(k)$ zero mean i.i.d. $\sigma_u^2$, $\sigma_y^2$
- $\hat{R}_{LS}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$
- $\lim_{N\to\infty} \hat{R}_{LS}(N) = \cdots = R_0 \dfrac{1}{1 + \sigma_u^2/u_0^2}$
- Biased! unless $\sigma_i = 0$
- $\hat{R}_{EV}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$
- $\lim_{N\to\infty} \hat{R}_{EV}(N) = \cdots = R_0$
- Unbiased!

# Analysis of cost

- $u(k) = u_0 + n_u(k)$ and $y(k) = y_0 + n_y(k)$
- $n_u(k)$ and $n_y(k)$ zero mean i.i.d. $\sigma_u^2$, $\sigma_y^2$
- $\hat{R}_{LS}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)u(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)^2}$
- $\lim_{N\to\infty} \hat{R}_{LS}(N) = \cdots = R_0 \dfrac{1}{1 + \sigma_u^2/u_0^2}$
- Biased! unless $\sigma_i = 0$
- $\hat{R}_{EV}(N) = \dfrac{\frac{1}{N}\sum_{k=1}^{N} y(k)}{\frac{1}{N}\sum_{k=1}^{N} u(k)}$
- $\lim_{N\to\infty} \hat{R}_{EV}(N) = \cdots = R_0$
- Unbiased!

# Analysis of cost

Different estimators minimize cost $V(R)$

- $V_{SA}(R) = \frac{1}{N}\sum_{k=1}^{N}(R(k) - R)^2, \quad R(k) = \frac{y(k)}{u(k)}$
- $V_{LS}(R) = \frac{1}{N}\sum_{k=1}^{N}(y(k) - Ru(k))^2$
- $V_{EV}(R) = \frac{1}{N}\left(\sum_{k=1}^{N}\frac{(y(k) - y_0)^2}{\sigma_y^2} + \sum_{k=1}^{N}\frac{(u(k) - u_0)^2}{\sigma_u^2}\right)$

  with constraint $y_0 = Ru_0$.

# Analysis of cost

Different estimators minimize cost $V(R)$

- $V_{SA}(R) = \frac{1}{N}\sum_{k=1}^{N}(R(k) - R)^2, \quad R(k) = \frac{y(k)}{u(k)}$
- $V_{LS}(R) = \frac{1}{N}\sum_{k=1}^{N}(y(k) - Ru(k))^2$
- $V_{EV}(R) = \frac{1}{N}\left(\sum_{k=1}^{N}\frac{(y(k) - y_0)^2}{\sigma_y^2} + \sum_{k=1}^{N}\frac{(u(k) - u_0)^2}{\sigma_u^2}\right)$

  with constraint $y_0 = Ru_0$.

# Analysis of cost

Different estimators minimize cost $V(R)$

- $V_{SA}(R) = \frac{1}{N}\sum_{k=1}^{N}(R(k) - R)^2, \quad R(k) = \frac{y(k)}{u(k)}$
- $V_{LS}(R) = \frac{1}{N}\sum_{k=1}^{N}(y(k) - Ru(k))^2$
- $V_{EV}(R) = \frac{1}{N}\left(\sum_{k=1}^{N}\frac{(y(k) - y_0)^2}{\sigma_y^2} + \sum_{k=1}^{N}\frac{(u(k) - u_0)^2}{\sigma_u^2}\right)$
  with constraint $y_0 = Ru_0$.

# Other models and cost functions

$Y(z) = G_0(z)U(z) + V(z)$,
$V(z) = H_0(z)E(z)$,
$E \longrightarrow e$ zero mean, variance $\sigma_e^2$



L. Ljung

| MODEL | $G$ | $H$ |
|---|---|---|
| ARX: | $B/A$ | $1/A$ |
| ARMAX: | $B/A$ | $C/A$ |
| OE: | $B/F$ | 1 |
| FIR: | $B$ | 1 |
| BJ: | $B/F$ | $C/D$ |

model $G$ and $H$ to minimize prediction error

cost: $V = \frac{1}{N}\sum_{k=1}^{N} |y(k) - \hat{y}(k)|^2$

prediction: $\hat{y}(k) = L[(1 - H^{-1})y(k) + H^{-1}Gu(k)]$ ($L$ is prefilter)

# Other models and cost functions

$Y(z) = G_0(z)U(z) + V(z)$,
$V(z) = H_0(z)E(z)$,
$E \longrightarrow e$ zero mean, variance $\sigma_e^2$



L. Ljung

| MODEL | $G$ | $H$ |
|---|---|---|
| ARX: | $B/A$ | $1/A$ |
| ARMAX: | $B/A$ | $C/A$ |
| OE: | $B/F$ | 1 |
| FIR: | $B$ | 1 |
| BJ: | $B/F$ | $C/D$ |

model $G$ and $H$ to minimize prediction error

cost: $V = \frac{1}{N}\sum_{k=1}^{N} |y(k) - \hat{y}(k)|^2$

prediction: $\hat{y}(k) = L[(1 - H^{-1})y(k) + H^{-1}Gu(k)]$ ($L$ is prefilter)

# Other models and cost functions

$Y(z) = G_0(z)U(z) + V(z)$,
$V(z) = H_0(z)E(z)$,
$E \longrightarrow e$ zero mean, variance $\sigma_e^2$



L. Ljung

| MODEL | $G$ | $H$ |
|---|---|---|
| ARX: | $B/A$ | $1/A$ |
| ARMAX: | $B/A$ | $C/A$ |
| OE: | $B/F$ | 1 |
| FIR: | $B$ | 1 |
| BJ: | $B/F$ | $C/D$ |

model $G$ and $H$ to minimize prediction error

cost: $V = \frac{1}{N}\sum_{k=1}^{N} |y(k) - \hat{y}(k)|^2$

prediction: $\hat{y}(k) = L[(1 - H^{-1})y(k) + H^{-1}Gu(k)]$ ($L$ is prefilter)

# Other models and cost functions

$Y(z) = G_0(z)U(z) + V(z)$,
$V(z) = H_0(z)E(z)$,
$E \longrightarrow e$ zero mean, variance $\sigma_e^2$



L. Ljung

| MODEL | $G$ | $H$ |
|---|---|---|
| ARX: | $B/A$ | $1/A$ |
| ARMAX: | $B/A$ | $C/A$ |
| OE: | $B/F$ | 1 |
| FIR: | $B$ | 1 |
| BJ: | $B/F$ | $C/D$ |

model $G$ and $H$ to minimize prediction error

cost: $V = \frac{1}{N}\sum_{k=1}^{N} |y(k) - \hat{y}(k)|^2$

prediction: $\hat{y}(k) = L[(1 - H^{-1})y(k) + H^{-1}Gu(k)]$ ($L$ is prefilter)

# Other models and cost functions

$Y(z) = G_0(z)U(z) + V(z)$,
$V(z) = H_0(z)E(z)$,
$E \longrightarrow e$ zero mean, variance $\sigma_e^2$



L. Ljung

| MODEL | $G$ | $H$ |
|---|---|---|
| ARX: | $B/A$ | $1/A$ |
| ARMAX: | $B/A$ | $C/A$ |
| OE: | $B/F$ | 1 |
| FIR: | $B$ | 1 |
| BJ: | $B/F$ | $C/D$ |

model $G$ and $H$ to minimize prediction error

cost: $V = \frac{1}{N}\sum_{k=1}^{N} |y(k) - \hat{y}(k)|^2$

prediction: $\hat{y}(k) = L[(1 - H^{-1})y(k) + H^{-1}Gu(k)]$ ($L$ is prefilter)

# Other models and cost functions

$Y(z) = G_0(z)U(z) + V(z)$,
$V(z) = H_0(z)E(z)$,
$E \longrightarrow e$ zero mean, variance $\sigma_e^2$



L. Ljung

| MODEL | $G$ | $H$ |
|---|---|---|
| ARX: | $B/A$ | $1/A$ |
| ARMAX: | $B/A$ | $C/A$ |
| OE: | $B/F$ | 1 |
| FIR: | $B$ | 1 |
| BJ: | $B/F$ | $C/D$ |



G. Box



G. Jenkins

model $G$ and $H$ to minimize prediction error

cost: $V = \frac{1}{N}\sum_{k=1}^{N} |y(k) - \hat{y}(k)|^2$

prediction: $\hat{y}(k) = L[(1 - H^{-1})y(k) + H^{-1}Gu(k)]$ ($L$ is prefilter)

# Other models and cost functions

$Y(z) = G_0(z)U(z) + V(z)$,
$V(z) = H_0(z)E(z)$,
$E \longrightarrow e$ zero mean, variance $\sigma_e^2$



L. Ljung

| MODEL | $G$ | $H$ |
|---|---|---|
| ARX: | $B/A$ | $1/A$ |
| ARMAX: | $B/A$ | $C/A$ |
| OE: | $B/F$ | 1 |
| FIR: | $B$ | 1 |
| BJ: | $B/F$ | $C/D$ |



G. Box



G. Jenkins

model $G$ and $H$ to minimize prediction error

cost: $V = \frac{1}{N}\sum_{k=1}^{N} |y(k) - \hat{y}(k)|^2$

prediction: $\hat{y}(k) = L[(1 - H^{-1})y(k) + H^{-1}Gu(k)]$ ($L$ is prefilter)

# Identification steps

- Collect data (experimental setup)
- Choose a model (linear/parametric/black box/...)
- Choose a cost function (LS/WLS/ML/EV/Bayes/...)
- Minimize the cost function
- Validation of the model

# Identification steps

- Collect data (experimental setup)
- Choose a model (linear/parametric/black box/...)
- Choose a cost function (LS/WLS/ML/EV/Bayes/...)
- Minimize the cost function
- Validation of the model

# Identification steps

- Collect data (experimental setup)
- Choose a model (linear/parametric/black box/...)
- Choose a cost function (LS/WLS/ML/EV/Bayes/...)
- Minimize the cost function
- Validation of the model

# Identification steps

- Collect data (experimental setup)
- Choose a model (linear/parametric/black box/...)
- Choose a cost function (LS/WLS/ML/EV/Bayes/...)
- Minimize the cost function
- Validation of the model

# Identification steps

- Collect data (experimental setup)
- Choose a model (linear/parametric/black box/...)
- Choose a cost function (LS/WLS/ML/EV/Bayes/...)
- Minimize the cost function
- Validation of the model

# MODEL = LTI system

MODEL = parametric linear time-invariant system

$$G(z,\theta) = \frac{B(z)}{A(z)} = \frac{\sum_{r=0}^{n_b} b_r z^{-r}}{\sum_{r=0}^{n_a} a_r z^{-r}}, \quad \theta = (a_0, \ldots, a_{n_a}, b_0, \ldots, b_{n_b})$$

DATA in frequency domain by DFT

$$U(k) = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} u(rT_s) z_k^{-1}, \quad Y(k) = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} y(rT_s) z_k^{-1},$$

$$z_k = e^{j2\pi k/N}$$

constraint $\quad Y(k) = G(z_k, \theta) U(k)$

# MODEL = LTI system

MODEL = parametric linear time-invariant system

$$G(z,\theta) = \frac{B(z)}{A(z)} = \frac{\sum_{r=0}^{n_b} b_r z^{-r}}{\sum_{r=0}^{n_a} a_r z^{-r}}, \quad \theta = (a_0, \ldots, a_{n_a}, b_0, \ldots, b_{n_b})$$

DATA in frequency domain by DFT

$$U(k) = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} u(rT_s) z_k^{-1}, \quad Y(k) = \frac{1}{\sqrt{N}} \sum_{r=0}^{N-1} y(rT_s) z_k^{-1},$$

$$z_k = e^{j2\pi k/N}$$

$$\text{constraint} \qquad Y(k) = G(z_k,\theta)U(k)$$

# Identification

- Given: measurements

$$\{G(z_k)\}_{k=1}^N \quad \text{or} \quad \{U(z_k), Y(z_k)\}_{k=1}^N, \quad z_k \in j\mathbb{R} \text{ or } \mathbb{T}$$

+ noise/uncertainty information $(\sigma_u^2, \sigma_y^2)$.

- Find estimate $\hat{G}$ or $\hat{Y} = \hat{G}U$ such that ...

$$\|G - \hat{G}\|_{w_g} \text{ or } \|AY - BU\|_{w_r} = \min$$

for some weights $w_g$ or $w_r$.

# Identification

- Given: measurements

$$\{G(z_k)\}_{k=1}^N \quad \text{or} \quad \{U(z_k), Y(z_k)\}_{k=1}^N, \quad z_k \in j\mathbb{R} \text{ or } \mathbb{T}$$

  + noise/uncertainty information $(\sigma_u^2, \sigma_y^2)$.

- Find estimate $\hat{G}$ or $\hat{Y} = \hat{G}U$ such that ...

$$\|G - \hat{G}\|_{w_g} \text{ or } \|AY - BU\|_{w_r} = \min$$

  for some weights $w_g$ or $w_r$.

# Norms

- Least squares and friends

  LS = Least Squares = RMS = Root Mean Squares

  WLS = Weighted Least Squares

  ML = Maximum Likelihood

  EV = Errors in Variables

  TLS = Total LS

  OR = Orthogonal regression

  PCA = Principal Component Analysis



- $H_\infty$, $L_1$ and the like
- Hankel norm

# Norms

- Least squares and friends

  LS = Least Squares = RMS = Root Mean Squares

  WLS = Weighted Least Squares

  ML = Maximum Likelihood

  EV = Errors in Variables

  TLS = Total LS

  OR = Orthogonal regression

  PCA = Principal Component Analysis



- $H_\infty$, $L_1$ and the like
- Hankel norm

# Norms

- Least squares and friends

  LS = Least Squares = RMS = Root Mean Squares

  WLS = Weighted Least Squares

  ML = Maximum Likelihood

  EV = Errors in Variables

  TLS = Total LS

  OR = Orthogonal regression

  PCA = Principal Component Analysis



- $H_\infty$, $L_1$ and the like
- Hankel norm

# Weighted least Squares

Continuous

$$\|f\|_w^2 = \int |w(z)f(z)|^2 d\omega.$$

Discrete

$$\|f\|_w^2 = \sum_{k=1}^{N} |w_k f(z_k)|^2.$$

mathematician
polynomials in $z$

engineer
polynomials in $z^{-1}$

# Weighted least Squares

Continuous

$$\|f\|_w^2 = \int |w(z)f(z)|^2 d\omega.$$

Discrete

$$\|f\|_w^2 = \sum_{k=1}^{N} |w_k f(z_k)|^2.$$

mathematician
polynomials in $z$

engineer
polynomials in $z^{-1}$

# Weighted least Squares

Continuous

$$\|f\|_w^2 = \int |w(z)f(z)|^2 d\omega.$$

Discrete

$$\|f\|_w^2 = \sum_{k=1}^{N} |w_k f(z_k)|^2.$$

mathematician
polynomials in $z$

engineer
polynomials in $z^{-1}$

# Weighted least Squares

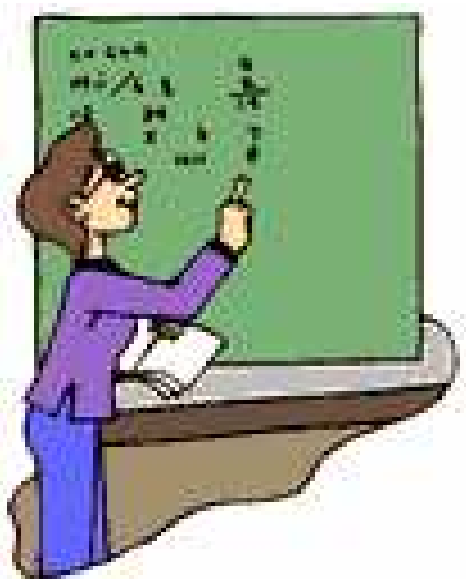
Continuous

$$\|f\|_w^2 = \int |w(z)f(z)|^2 d\omega.$$

Discrete

$$\|f\|_w^2 = \sum_{k=1}^{N} |w_k f(z_k)|^2.$$

mathematician
polynomials in $z$



engineer
polynomials in $z^{-1}$



# Weights

Data dependent

Measurements $\{G(z_k)\}$, with variance $\{\sigma_k\}$:

$$\|G - \hat{G}\|_w^2 = \sum_{k=1}^{N} \frac{|G(z_k) - \hat{G}(z_k)|^2}{\sigma_k^2}, \quad w_k = \sigma_k^{-1}$$

$$\begin{aligned} \|Y - \hat{Y}\|_{w_y}^2 &= \|(G - \hat{G})U\|_{w_y}^2 = \sum_k |G(k) - \hat{G}(k)|^2 \underbrace{|U(k)|^2 w_y^2(k)}_{w_g^2} \\ &= \sum_k |G(k) - \hat{G}(k)|^2 w_g^2(k), \quad w_g^2 = |U|^2 w_y^2 \\ &= \|G - \hat{G}\|_{w_g}^2, \quad w_g^2 = w_y^2 |U|^2. \end{aligned}$$

# Weights

Data dependent

Measurements $\{G(z_k)\}$, with variance $\{\sigma_k\}$:

$$\|G - \hat{G}\|_w^2 = \sum_{k=1}^{N} \frac{|G(z_k) - \hat{G}(z_k)|^2}{\sigma_k^2}, \quad w_k = \sigma_k^{-1}$$

$$\begin{aligned} \|Y - \hat{Y}\|_{w_y}^2 &= \|(G - \hat{G})U\|_{w_y}^2 = \sum_k |G(k) - \hat{G}(k)|^2 \underbrace{|U(k)|^2 w_y^2(k)}_{w_g^2} \\ &= \sum_k |G(k) - \hat{G}(k)|^2 w_g^2(k), \quad w_g^2 = |U|^2 w_y^2 \\ &= \|G - \hat{G}\|_{w_g}^2, \quad w_g^2 = w_y^2 |U|^2. \end{aligned}$$

# Weights

Solution dependent

$$\|G - \hat{G}\|_{w_g} = \left\| \frac{Y}{U} - \frac{B}{A} \right\|_{w_g}$$

$$= \left\| \frac{YA - UB}{AU} \right\|_{w_g}$$

$$= \|YA - UB\|_{\tilde{w}_g}, \quad \tilde{w}_g = \frac{w_g}{|AU|}$$

ML cost is similar

$$\|YA - UB\|_{w_{ML}} = \|R\|_{w_{ML}}, \quad w_{ML}^2 = \frac{1}{\sigma_R^2}$$

# Weights

Solution dependent

$$\begin{aligned} \|G - \hat{G}\|_{w_g} &= \left\| \frac{Y}{U} - \frac{B}{A} \right\|_{w_g} \\ &= \left\| \frac{YA - UB}{AU} \right\|_{w_g} \\ &= \|YA - UB\|_{\tilde{w}_g}, \quad \tilde{w}_g = \frac{w_g}{|AU|} \end{aligned}$$

ML cost is similar

$$\|YA - UB\|_{w_{ML}} = \|R\|_{w_{ML}}, \quad w_{ML}^2 = \frac{1}{\sigma_R^2}$$

# Weights

Solution dependent

$$\begin{aligned} \|G - \hat{G}\|_{w_g} &= \left\| \frac{Y}{U} - \frac{B}{A} \right\|_{w_g} \\ &= \left\| \frac{YA - UB}{AU} \right\|_{w_g} \\ &= \|YA - UB\|_{\tilde{w}_g}, \quad \tilde{w}_g = \frac{w_g}{|AU|} \end{aligned}$$

ML cost is similar

$$\|YA - UB\|_{w_{ML}} = \|R\|_{w_{ML}}, \quad w_{ML}^2 = \frac{1}{\sigma_R^2}$$

# Weights

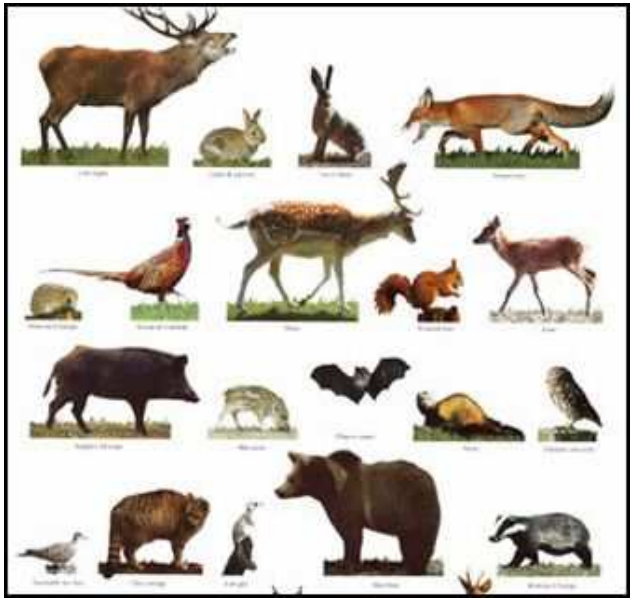
Solution dependent

$$\begin{aligned} \|G - \hat{G}\|_{w_g} &= \left\| \frac{Y}{U} - \frac{B}{A} \right\|_{w_g} \\ &= \left\| \frac{YA - UB}{AU} \right\|_{w_g} \\ &= \|YA - UB\|_{\tilde{w}_g}, \quad \tilde{w}_g = \frac{w_g}{|AU|} \end{aligned}$$

ML cost is similar

$$\|YA - UB\|_{w_{ML}} = \|R\|_{w_{ML}}, \quad w_{ML}^2 = \frac{1}{\sigma_R^2}$$

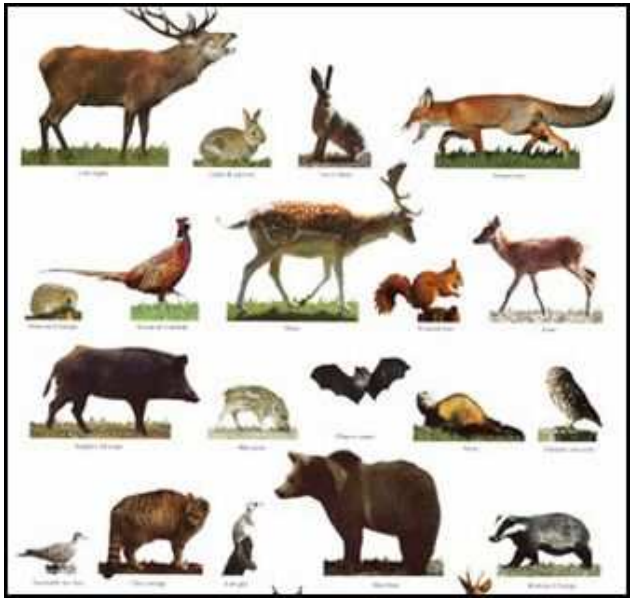
# A zoo of models



A jungle of costs

# Models and cost

## A zoo of models





## A jungle of costs

# Summary

- Identification problem
- Orthogonal Rational Functions
- Orthogonal Vector Polynomials

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k / A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\boldsymbol{\lambda}$ (ill posed problem)

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k / A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\boldsymbol{\lambda}$ (ill posed problem)

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k/A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\boldsymbol{\lambda}$ (ill posed problem)

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k/A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\boldsymbol{\lambda}$ (ill posed problem)

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k/A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\lambda$ (ill posed problem)

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k/A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\lambda$ (ill posed problem)

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k/A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\lambda$ (ill posed problem)

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k/A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\boldsymbol{\lambda}$ (ill posed problem)

# Rational model

- $G$ is a rational from $\mathcal{L}_n = \operatorname{span}\{\phi_0, \ldots, \phi_n\}$
- $G(z) = \sum_{k=0}^{n} \phi_k(z)\lambda_k = [\phi_0, \ldots, \phi_n]\boldsymbol{\lambda} = \Phi(z)\boldsymbol{\lambda}$
- E.g. $\phi_k(z) = z^k/A(z)$
- linear parameters $\boldsymbol{\lambda}$, nonlinear parameters in $A$.
- LS solution satisfies orthogonality relations $G - \hat{G} \perp \mathcal{L}_n$
- $\sum_{j=0}^{n} \langle \phi_k, \phi_j \rangle \lambda_j = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$. (normal eqs)
- if orthonormal basis then $\lambda_k = \langle \phi_k, G \rangle$, $k = 0, \ldots, n$.
- linear parameters depend on the choice of the nonlinear parameters $A$
- wrong $A$ can be partly corrected by $\boldsymbol{\lambda}$ (ill posed problem)

# Laguerre basis



E. Laguerre

- Laguerre polynomials $\int_0^\infty e^{-x} L_p(x) L_q(x) dx = \delta_{pq}$
- Laguerre functions $\psi_n(x) = e^{-x/2} L_n(x)$ orthogonal with weight 1
- in the frequency domain (c.t.) $x = 2\alpha t$

$$\phi_k(z) = \frac{\sqrt{2\alpha}}{z+\alpha} \left( \frac{z-\alpha}{z+\alpha} \right)^n$$

- d.t. analog

$$\phi_k(z) = \frac{\sqrt{1-\xi^2}}{z-\xi} \left( \frac{\xi z - 1}{z-\xi} \right)^n, \quad \xi = \frac{1-\alpha}{1+\alpha}.$$

- Still orthogonal, weight $= 1$ (Parseval)
- Only one pole

# Laguerre basis



E. Laguerre

- Laguerre polynomials $\int_0^\infty e^{-x} L_p(x) L_q(x) dx = \delta_{pq}$
- Laguerre functions $\psi_n(x) = e^{-x/2} L_n(x)$ orthogonal with weight 1
- in the frequency domain (c.t.) $x = 2\alpha t$

$$\phi_k(z) = \frac{\sqrt{2\alpha}}{z+\alpha} \left( \frac{z-\alpha}{z+\alpha} \right)^n$$

- d.t. analog

$$\phi_k(z) = \frac{\sqrt{1-\xi^2}}{z-\xi} \left( \frac{\xi z - 1}{z-\xi} \right)^n, \quad \xi = \frac{1-\alpha}{1+\alpha}.$$

- Still orthogonal, weight $= 1$ (Parseval)
- Only one pole

# Laguerre basis



E. Laguerre

- Laguerre polynomials $\int_0^\infty e^{-x} L_p(x) L_q(x) dx = \delta_{pq}$
- Laguerre functions $\psi_n(x) = e^{-x/2} L_n(x)$ orthogonal with weight 1
- in the frequency domain (c.t.) $x = 2\alpha t$

$$\phi_k(z) = \frac{\sqrt{2\alpha}}{z+\alpha} \left( \frac{z-\alpha}{z+\alpha} \right)^n$$

- d.t. analog

$$\phi_k(z) = \frac{\sqrt{1-\xi^2}}{z-\xi} \left( \frac{\xi z - 1}{z-\xi} \right)^n, \quad \xi = \frac{1-\alpha}{1+\alpha}.$$

- Still orthogonal, weight $= 1$ (Parseval)
- Only one pole

# Laguerre basis



E. Laguerre

- Laguerre polynomials $\int_0^\infty e^{-x} L_p(x) L_q(x) dx = \delta_{pq}$
- Laguerre functions $\psi_n(x) = e^{-x/2} L_n(x)$ orthogonal with weight 1
- in the frequency domain (c.t.) $x = 2\alpha t$

$$\phi_k(z) = \frac{\sqrt{2\alpha}}{z+\alpha} \left( \frac{z-\alpha}{z+\alpha} \right)^n$$



Y.W. Lee



N. Wiener 1933

- d.t. analog

$$\phi_k(z) = \frac{\sqrt{1-\xi^2}}{z-\xi} \left( \frac{\xi z - 1}{z-\xi} \right)^n, \quad \xi = \frac{1-\alpha}{1+\alpha}.$$

- Still orthogonal, weight = 1 (Parseval)
- Only one pole

# Laguerre basis



E. Laguerre

- Laguerre polynomials $\int_0^\infty e^{-x} L_p(x) L_q(x) dx = \delta_{pq}$
- Laguerre functions $\psi_n(x) = e^{-x/2} L_n(x)$ orthogonal with weight 1
- in the frequency domain (c.t.) $x = 2\alpha t$

$$\phi_k(z) = \frac{\sqrt{2\alpha}}{z+\alpha} \left( \frac{z-\alpha}{z+\alpha} \right)^n$$



Y.W. Lee



N. Wiener 1933

- d.t. analog

$$\phi_k(z) = \frac{\sqrt{1-\xi^2}}{z-\xi} \left( \frac{\xi z - 1}{z-\xi} \right)^n, \quad \xi = \frac{1-\alpha}{1+\alpha}.$$

- Still orthogonal, weight = 1 (Parseval)
- Only one pole

# Laguerre basis



E. Laguerre

- Laguerre polynomials $\int_0^\infty e^{-x} L_p(x) L_q(x) dx = \delta_{pq}$
- Laguerre functions $\psi_n(x) = e^{-x/2} L_n(x)$ orthogonal with weight 1
- in the frequency domain (c.t.) $x = 2\alpha t$

$$\phi_k(z) = \frac{\sqrt{2\alpha}}{z+\alpha} \left( \frac{z-\alpha}{z+\alpha} \right)^n$$



Y.W. Lee



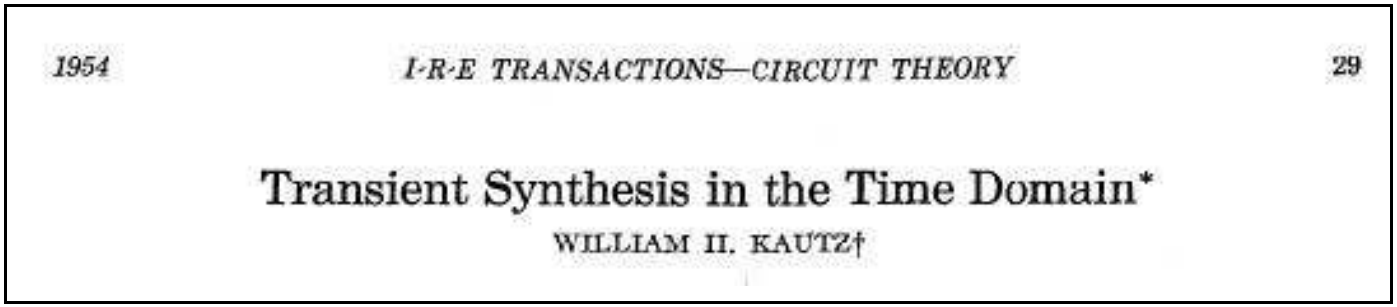
N. Wiener 1933

- d.t. analog

$$\phi_k(z) = \frac{\sqrt{1-\xi^2}}{z-\xi} \left( \frac{\xi z - 1}{z-\xi} \right)^n, \quad \xi = \frac{1-\alpha}{1+\alpha}.$$

- Still orthogonal, weight = 1 (Parseval)
- Only one pole

# Kautz basis



1954 I-R-E TRANSACTIONS—CIRCUIT THEORY 29

Transient Synthesis in the Time Domain*

WILLIAM H. KAUTZ†

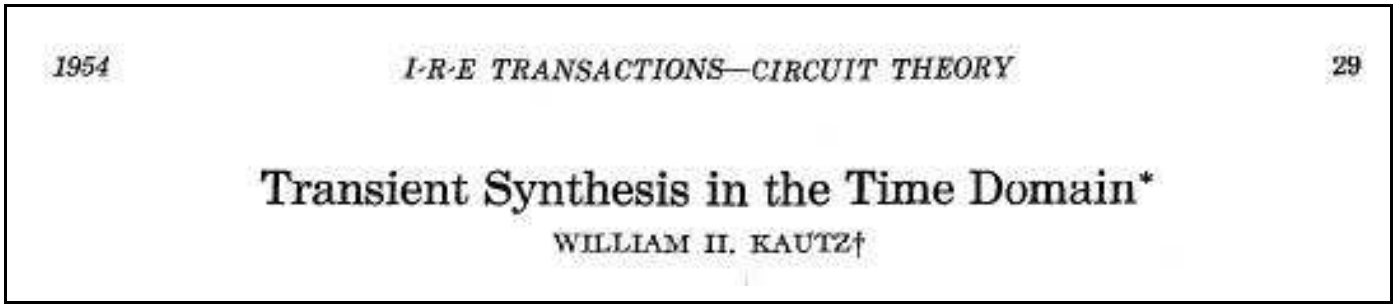
- 1954 W.H. Kautz (circuit Th.) introduces more (complex conjugate pairs of) poles
- c.t. time domain: $a_{k1}e^{-\alpha_1 t} + \cdots + a_{kk}e^{-\alpha_n t}$
- Laplace transform: $\frac{a_{k1}}{z+\alpha_1} + \cdots + \frac{a_{kk}}{z+\alpha_k}$
-

$$\phi_n(z) = \frac{\sqrt{2\alpha_n}}{z+\alpha_n} \prod_{k=1}^{n-1} \left( \frac{z-\alpha_k}{z+\alpha_k} \right)$$

- $\alpha_k = -\alpha$ for all $k \rightarrow$ Laguerre
- d.t analog

$$\phi_n(z) = \frac{\sqrt{1-\xi_n^2}}{z-\xi_n} \prod_{k=1}^{n-1} \left( \frac{1-\xi_k z}{z-\xi_k} \right), \quad \xi_k = \frac{1-\alpha_k}{1+\alpha_k}$$

# Kautz basis



1954 I-R-E TRANSACTIONS—CIRCUIT THEORY 29

Transient Synthesis in the Time Domain*

WILLIAM H. KAUTZ†

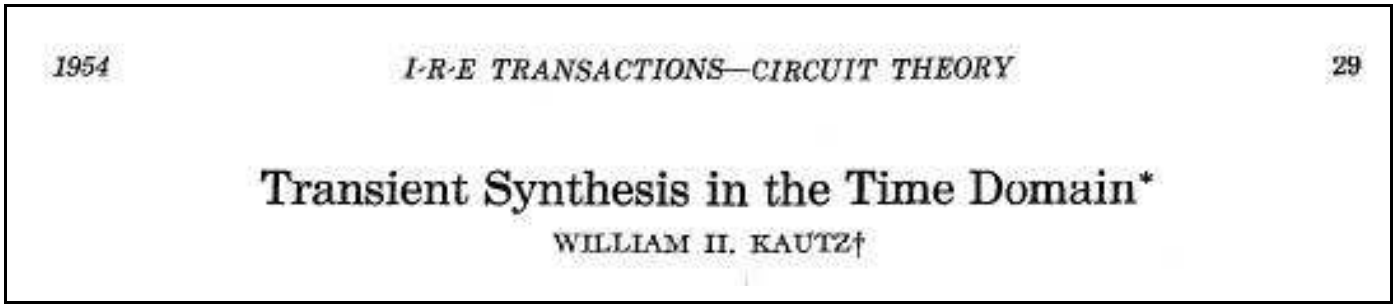
- 1954 W.H. Kautz (circuit Th.) introduces more (complex conjugate pairs of) poles
- c.t. time domain: $a_{k1}e^{-\alpha_1 t} + \cdots + a_{kk}e^{-\alpha_n t}$
- Laplace transform: $\frac{a_{k1}}{z+\alpha_1} + \cdots + \frac{a_{kk}}{z+\alpha_k}$
-

$$\phi_n(z) = \frac{\sqrt{2\alpha_n}}{z+\alpha_n}\prod_{k=1}^{n-1}\left(\frac{z-\alpha_k}{z+\alpha_k}\right)$$

- $\alpha_k = -\alpha$ for all $k \rightarrow$ Laguerre
- d.t analog

$$\phi_n(z) = \frac{\sqrt{1-\xi_n^2}}{z-\xi_n}\prod_{k=1}^{n-1}\left(\frac{1-\xi_k z}{z-\xi_k}\right), \quad \xi_k = \frac{1-\alpha_k}{1+\alpha_k}$$

# Kautz basis



1954 I·R·E TRANSACTIONS—CIRCUIT THEORY 29

Transient Synthesis in the Time Domain*

WILLIAM H. KAUTZ†

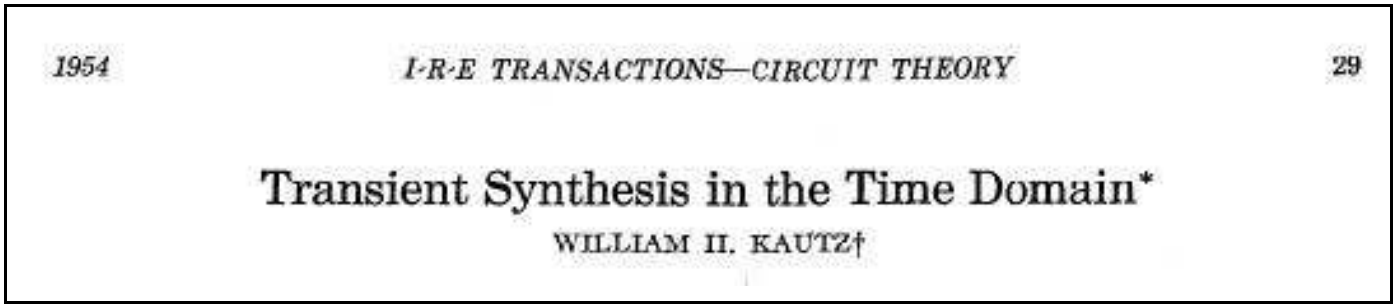
- 1954 W.H. Kautz (circuit Th.) introduces more (complex conjugate pairs of) poles
- c.t. time domain: $a_{k1}e^{-\alpha_1 t} + \cdots + a_{kk}e^{-\alpha_n t}$
- Laplace transform: $\frac{a_{k1}}{z+\alpha_1} + \cdots + \frac{a_{kk}}{z+\alpha_k}$
-

$$\phi_n(z) = \frac{\sqrt{2\alpha_n}}{z+\alpha_n}\prod_{k=1}^{n-1}\left(\frac{z-\alpha_k}{z+\alpha_k}\right)$$

- $\alpha_k = -\alpha$ for all $k \rightarrow$ Laguerre
- d.t analog

$$\phi_n(z) = \frac{\sqrt{1-\xi_n^2}}{z-\xi_n}\prod_{k=1}^{n-1}\left(\frac{1-\xi_k z}{z-\xi_k}\right), \quad \xi_k = \frac{1-\alpha_k}{1+\alpha_k}$$

# Kautz basis



1954 I·R·E TRANSACTIONS—CIRCUIT THEORY 29

Transient Synthesis in the Time Domain*

WILLIAM H. KAUTZ†

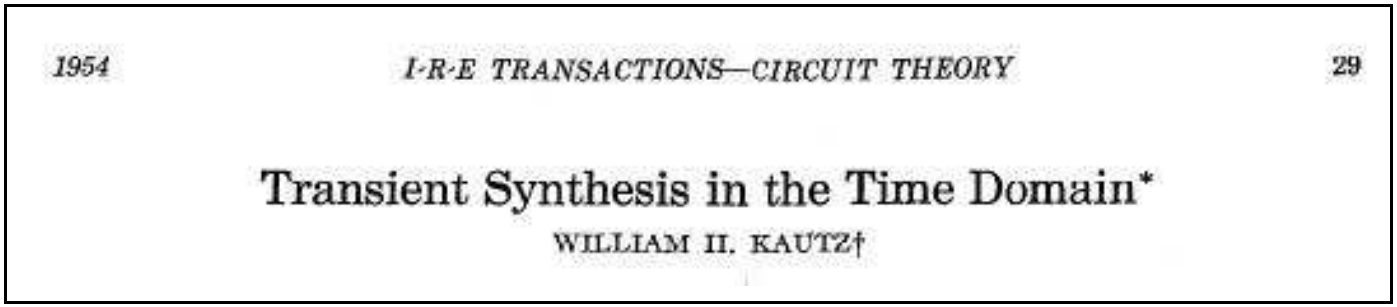
- 1954 W.H. Kautz (circuit Th.) introduces more (complex conjugate pairs of) poles
- c.t. time domain: $a_{k1}e^{-\alpha_1 t} + \cdots + a_{kk}e^{-\alpha_n t}$
- Laplace transform: $\frac{a_{k1}}{z+\alpha_1} + \cdots + \frac{a_{kk}}{z+\alpha_k}$
-

$$\phi_n(z) = \frac{\sqrt{2\alpha_n}}{z+\alpha_n} \prod_{k=1}^{n-1} \left( \frac{z-\alpha_k}{z+\alpha_k} \right)$$

- $\alpha_k = -\alpha$ for all $k \rightarrow$ Laguerre
- d.t analog

$$\phi_n(z) = \frac{\sqrt{1-\xi_n^2}}{z-\xi_n} \prod_{k=1}^{n-1} \left( \frac{1-\xi_k z}{z-\xi_k} \right), \quad \xi_k = \frac{1-\alpha_k}{1+\alpha_k}$$

# Kautz basis



1954 I-R-E TRANSACTIONS—CIRCUIT THEORY 29

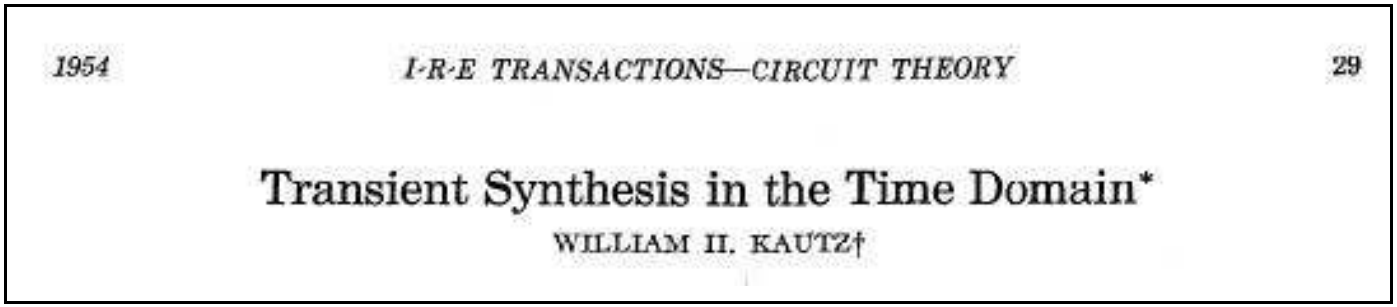
Transient Synthesis in the Time Domain*

WILLIAM H. KAUTZ†

- 1954 W.H. Kautz (circuit Th.) introduces more (complex conjugate pairs of) poles
- c.t. time domain: $a_{k1}e^{-\alpha_1 t} + \cdots + a_{kk}e^{-\alpha_n t}$
- Laplace transform: $\frac{a_{k1}}{z+\alpha_1} + \cdots + \frac{a_{kk}}{z+\alpha_k}$
- $$\phi_n(z) = \frac{\sqrt{2\alpha_n}}{z+\alpha_n}\prod_{k=1}^{n-1}\left(\frac{z-\alpha_k}{z+\alpha_k}\right)$$
- $\alpha_k = -\alpha$ for all $k \rightarrow$ Laguerre
- d.t analog

$$\phi_n(z) = \frac{\sqrt{1-\xi_n^2}}{z-\xi_n}\prod_{k=1}^{n-1}\left(\frac{1-\xi_k z}{z-\xi_k}\right), \quad \xi_k = \frac{1-\alpha_k}{1+\alpha_k}$$

# Kautz basis



1954 I-R-E TRANSACTIONS—CIRCUIT THEORY 29

Transient Synthesis in the Time Domain*

WILLIAM H. KAUTZ†

- 1954 W.H. Kautz (circuit Th.) introduces more (complex conjugate pairs of) poles
- c.t. time domain: $a_{k1}e^{-\alpha_1 t} + \cdots + a_{kk}e^{-\alpha_n t}$
- Laplace transform: $\frac{a_{k1}}{z+\alpha_1} + \cdots + \frac{a_{kk}}{z+\alpha_k}$
- $$\phi_n(z) = \frac{\sqrt{2\alpha_n}}{z+\alpha_n}\prod_{k=1}^{n-1}\left(\frac{z-\alpha_k}{z+\alpha_k}\right)$$
- $\alpha_k = -\alpha$ for all $k \rightarrow$ Laguerre
- d.t analog

$$\phi_n(z) = \frac{\sqrt{1-\xi_n^2}}{z-\xi_n}\prod_{k=1}^{n-1}\left(\frac{1-\xi_k z}{z-\xi_k}\right), \quad \xi_k = \frac{1-\alpha_k}{1+\alpha_k}$$

# Kautz = Takenaka-Malmquist = Džrbašian basis

- J.L. Walsh: (1960) refers to
  S. Takenaka (1925) Japanese J. Math.
  F. Malmquist (1926) Congress of Scand. Mathematics
  J.L. Walsh (1932) Trans AMS
- M.M. Džrbašian: (1962) ORFUC

# Kautz = Takenaka-Malmquist = Džrbašian basis

- J.L. Walsh: (1960) refers to
  S. Takenaka (1925) Japanese J. Math.
  F. Malmquist (1926) Congress of Scand. Mathematics
  J.L. Walsh (1932) Trans AMS
- M.M. Džrbašian: (1962) ORFUC

# Orthogonal Rational Functions

- Rational approximant $\hat{G} \Rightarrow$ use rational basis
- $\hat{G}(z) = \sum_{k=0}^{n} \phi_k \lambda_k \in \mathcal{L}_n$
- $\mathcal{L}_n = \dfrac{p_n(z)}{(1-\alpha_1/z)\cdots(1-\alpha_n/z)}$
- $\phi_k \in \mathcal{L}_k \setminus \mathcal{L}_{k-1}$, $\phi_k \perp \mathcal{L}_{k-1}$
- Use Gram-Schmidt procedure on

$$\{B_0, B_1, \ldots, B_n\}, \quad B_0 = 1, B_k = \prod_{l=1}^{k} \frac{1}{1-\alpha_l/z}, k \geq 1$$

- Choice of poles $\boldsymbol{\alpha} = \{\alpha_k\} \Rightarrow$ linear problem

# Orthogonal Rational Functions

- Rational approximant $\hat{G} \Rightarrow$ use rational basis
- $\hat{G}(z) = \sum_{k=0}^{n} \phi_k \lambda_k \in \mathcal{L}_n$
- $\mathcal{L}_n = \dfrac{p_n(z)}{(1-\alpha_1/z)\cdots(1-\alpha_n/z)}$
- $\phi_k \in \mathcal{L}_k \setminus \mathcal{L}_{k-1}$, $\phi_k \perp \mathcal{L}_{k-1}$
- Use Gram-Schmidt procedure on

$$\{B_0, B_1, \ldots, B_n\}, \quad B_0 = 1, B_k = \prod_{l=1}^{k} \frac{1}{1-\alpha_l/z}, k \geq 1$$

- Choice of poles $\alpha = \{\alpha_k\} \Rightarrow$ linear problem

# Orthogonal Rational Functions

- Rational approximant $\hat{G} \Rightarrow$ use rational basis
- $\hat{G}(z) = \sum_{k=0}^{n} \phi_k \lambda_k \in \mathcal{L}_n$
- $\mathcal{L}_n = \dfrac{p_n(z)}{(1-\alpha_1/z)\cdots(1-\alpha_n/z)}$
- $\phi_k \in \mathcal{L}_k \setminus \mathcal{L}_{k-1}$, $\phi_k \perp \mathcal{L}_{k-1}$
- Use Gram-Schmidt procedure on

$$\{B_0, B_1, \ldots, B_n\}, \quad B_0 = 1, B_k = \prod_{l=1}^{k} \frac{1}{1-\alpha_l/z}, k \geq 1$$

- Choice of poles $\alpha = \{\alpha_k\} \Rightarrow$ linear problem

# Orthogonal Rational Functions

- Rational approximant $\hat{G} \Rightarrow$ use rational basis
- $\hat{G}(z) = \sum_{k=0}^{n} \phi_k \lambda_k \in \mathcal{L}_n$
- $\mathcal{L}_n = \dfrac{p_n(z)}{(1-\alpha_1/z)\cdots(1-\alpha_n/z)}$
- $\phi_k \in \mathcal{L}_k \setminus \mathcal{L}_{k-1}$, $\phi_k \perp \mathcal{L}_{k-1}$
- Use Gram-Schmidt procedure on

$$\{B_0, B_1, \ldots, B_n\}, \quad B_0 = 1, B_k = \prod_{l=1}^{k} \frac{1}{1-\alpha_l/z}, k \geq 1$$

- Choice of poles $\boldsymbol{\alpha} = \{\alpha_k\} \Rightarrow$ linear problem

# Orthogonal Rational Functions

- Rational approximant $\hat{G} \Rightarrow$ use rational basis
- $\hat{G}(z) = \sum_{k=0}^{n} \phi_k \lambda_k \in \mathcal{L}_n$
- $\mathcal{L}_n = \dfrac{p_n(z)}{(1-\alpha_1/z)\cdots(1-\alpha_n/z)}$
- $\phi_k \in \mathcal{L}_k \setminus \mathcal{L}_{k-1}$, $\phi_k \perp \mathcal{L}_{k-1}$
- Use Gram-Schmidt procedure on

$$\{B_0, B_1, \ldots, B_n\}, \quad B_0 = 1, B_k = \prod_{l=1}^{k} \frac{1}{1-\alpha_l/z}, k \geq 1$$

- Choice of poles $\boldsymbol{\alpha} = \{\alpha_k\} \Rightarrow$ linear problem

# Orthogonal Rational Functions

- Rational approximant $\hat{G} \Rightarrow$ use rational basis
- $\hat{G}(z) = \sum_{k=0}^{n} \phi_k \lambda_k \in \mathcal{L}_n$
- $\mathcal{L}_n = \dfrac{p_n(z)}{(1-\alpha_1/z)\cdots(1-\alpha_n/z)}$
- $\phi_k \in \mathcal{L}_k \setminus \mathcal{L}_{k-1}$, $\phi_k \perp \mathcal{L}_{k-1}$
- Use Gram-Schmidt procedure on

$$\{B_0, B_1, \ldots, B_n\}, \quad B_0 = 1, B_k = \prod_{l=1}^{k} \frac{1}{1-\alpha_l/z}, k \geq 1$$

- Choice of poles $\boldsymbol{\alpha} = \{\alpha_k\} \Rightarrow$ linear problem

# Takenaka-Malmquist basis

- For discrete systems:

$$\phi_k = \frac{\sqrt{1-|\alpha_k|^2}z}{z-\alpha_k}\prod_{l=1}^{n-1}\frac{1-\overline{\alpha}_l z}{z-\alpha_l}$$

  is orthonormal w.r.t.

$$\langle f, g\rangle = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(e^{i\omega})\overline{g(e^{i\omega})}d\omega$$

- Good for solving

$$\min \frac{1}{2\pi}\int_{-\pi}^{\pi} |G(e^{i\omega}) - \hat{G}(e^{i\omega})|^2 d\omega$$

# Takenaka-Malmquist basis

- For discrete systems:

$$\phi_k = \frac{\sqrt{1-|\alpha_k|^2}z}{z-\alpha_k}\prod_{l=1}^{n-1}\frac{1-\overline{\alpha}_l z}{z-\alpha_l}$$

  is orthonormal w.r.t.

$$\langle f, g\rangle = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(e^{i\omega})\overline{g(e^{i\omega})}d\omega$$

- Good for solving

$$\min \frac{1}{2\pi}\int_{-\pi}^{\pi} |G(e^{i\omega}) - \hat{G}(e^{i\omega})|^2 d\omega$$

# Orthogonality w.r.t. summation

- BUT
  given $G(z_k)$ and $w_k$,
  find $\min \sum_{k=1}^{N} w_k |G(z_k) - \hat{G}(z_k)|^2$
- thus need orthogonality w.r.t.

$$\langle f, g \rangle_w = \sum_{k=1}^{N} f(z_k) w_k \overline{g(z_k)}$$

# Orthogonality w.r.t. summation

- BUT
  given $G(z_k)$ and $w_k$,
  find $\min \sum_{k=1}^{N} w_k |G(z_k) - \hat{G}(z_k)|^2$
- thus need orthogonality w.r.t.

$$\langle f, g \rangle_w = \sum_{k=1}^{N} f(z_k) w_k \overline{g(z_k)}$$

# Nonlinear problem

- if $\alpha_k$ given, finding $\frac{B}{A} = \sum_{l=0}^{n} \phi_l \lambda_l$ is linear

  $$\sum \phi_l \lambda_l = [\phi_0 \dots \phi_n]\boldsymbol{\lambda} = \Phi(\cdot; \boldsymbol{\alpha})\boldsymbol{\lambda}$$

$$\begin{bmatrix} \phi_0(z_1) & \cdots & \phi_n(z_1) \\ \vdots & & \vdots \\ \phi_0(z_N) & \cdots & \phi_n(z_N) \end{bmatrix} \begin{bmatrix} \lambda_0 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} G(z_1) \\ \vdots \\ G(z_N) \end{bmatrix}$$

- min $C(\boldsymbol{\alpha})$,

$$C(\boldsymbol{\alpha}) \quad = \quad \|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}(\boldsymbol{\alpha})\|_W$$

- nonconvex, needs good starting point for global min
- stability, needs constraints $|\alpha_k| < 1$.

# Nonlinear problem

- if $\alpha_k$ given, finding $\frac{B}{A} = \sum_{l=0}^{n} \phi_l \lambda_l$ is linear
- $\|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}\|_W$
- min $C(\boldsymbol{\alpha})$,

$$C(\boldsymbol{\alpha}) \quad = \quad \|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}(\boldsymbol{\alpha})\|_W$$

- nonconvex, needs good starting point for global min
- stability, needs constraints $|\alpha_k| < 1$.

# Nonlinear problem

- if $\alpha_k$ given, finding $\frac{B}{A} = \sum_{l=0}^{n} \phi_l \lambda_l$ is linear
- $\|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}\|_W$
- $\boldsymbol{\lambda}(\boldsymbol{\alpha}) = [\boldsymbol{\Phi}(\boldsymbol{\alpha})^H \mathbf{w} \boldsymbol{\Phi}(\boldsymbol{\alpha})]^{-1} \boldsymbol{\Phi}(\boldsymbol{\alpha})^H \mathbf{W}^H \mathbf{W} G(\mathbf{z})$
- min $C(\boldsymbol{\alpha})$,

$$C(\boldsymbol{\alpha}) \quad = \quad \|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}(\boldsymbol{\alpha})\|_W$$

- nonconvex, needs good starting point for global min
- stability, needs constraints $|\alpha_k| < 1$.

# Nonlinear problem

- if $\alpha_k$ given, finding $\frac{B}{A} = \sum_{l=0}^{n} \phi_l \lambda_l$ is linear
- $\|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}\|_W$
- $\boldsymbol{\lambda}(\boldsymbol{\alpha}) = [\boldsymbol{\Phi}(\boldsymbol{\alpha})^H \mathbf{W}^H \mathbf{W} \boldsymbol{\Phi}(\boldsymbol{\alpha})]^{-1} \boldsymbol{\Phi}(\boldsymbol{\alpha})^H \mathbf{W}^H \mathbf{W} G(\mathbf{z})$
- $\min C(\boldsymbol{\alpha})$,

$$C(\boldsymbol{\alpha}) \quad = \quad \|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}(\boldsymbol{\alpha})\|_W$$

- nonconvex, needs good starting point for global min
- stability, needs constraints $|\alpha_k| < 1$.

# Nonlinear problem

- if $\alpha_k$ given, finding $\frac{B}{A} = \sum_{l=0}^{n} \phi_l \lambda_l$ is linear
- $\|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}\|_W$
- $\boldsymbol{\lambda}(\boldsymbol{\alpha}) = \boldsymbol{\Phi}(\boldsymbol{\alpha})^H \mathbf{W}^H \mathbf{W} G(\mathbf{z})$
- min $C(\boldsymbol{\alpha})$,

$$C(\boldsymbol{\alpha}) \quad = \quad \|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}(\boldsymbol{\alpha})\|_W$$

- nonconvex, needs good starting point for global min
- stability, needs constraints $|\alpha_k| < 1$.

# Nonlinear problem

- if $\alpha_k$ given, finding $\frac{B}{A} = \sum_{l=0}^{n} \phi_l \lambda_l$ is linear
- $\|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}\|_W$
- $\boldsymbol{\lambda}(\boldsymbol{\alpha}) = \boldsymbol{\Phi}(\boldsymbol{\alpha})^H \mathbf{W}^H \mathbf{W} G(\mathbf{z})$
- min $C(\boldsymbol{\alpha})$,

$$C(\boldsymbol{\alpha}) \quad = \quad \|G(\mathbf{z}) - \Phi(\mathbf{z}; \boldsymbol{\alpha})\boldsymbol{\lambda}(\boldsymbol{\alpha})\|_W$$

- nonconvex, needs good starting point for global min
- stability, needs constraints $|\alpha_k| < 1$.

# Recurrence for ORF

continuous time

- OPS: $z^{-1}\phi_k(z) \in \mathrm{span}\{\phi_l : l = k-1, k, k+1\}$
- ORF:(generic case = regular system: $\phi_k(\alpha_{k-1}) \neq 0$)

$$z^{-1}\phi_k(z) \in \mathrm{span}\{[1 - \alpha_l/z]\phi_l(z) : l = k-1, k, k+1\}$$

- $\Phi = \Phi(zI - \mathbf{A})\mathbf{T} + [0 \ldots 0 \ (z - \alpha_{n+1})\phi_{n+1} b_{n+1}]$

$$\Phi = [\phi_0, \ldots, \phi_n], \mathbf{A} = \mathrm{diag}(1, \alpha_1, \ldots, \alpha_n),$$

$$\mathbf{T} = \begin{bmatrix} a_0 & b_0 & & & \\ b_0 & a_1 & b_1 & & \\ & b_1 & \ddots & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{bmatrix}$$

# Recurrence for ORF

continuous time

- OPS: $z^{-1}\phi_k(z) \in \mathrm{span}\{\phi_l : l = k-1, k, k+1\}$
- ORF:(generic case = regular system: $\phi_k(\alpha_{k-1}) \neq 0$)

$$z^{-1}\phi_k(z) \in \mathrm{span}\{[1-\alpha_l/z]\phi_l(z) : l = k-1, k, k+1\}$$

- $\Phi = \Phi(zI - \mathbf{A})\mathbf{T} + [0 \ldots 0 \ (z-\alpha_{n+1})\phi_{n+1} b_{n+1}]$

$$\Phi = [\phi_0, \ldots, \phi_n], \mathbf{A} = \mathrm{diag}(1, \alpha_1, \ldots, \alpha_n),$$

$$\mathbf{T} = \begin{bmatrix} a_0 & b_0 & & & \\ b_0 & a_1 & b_1 & & \\ & b_1 & \ddots & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{bmatrix}$$

# Recurrence for ORF

continuous time

- OPS: $z^{-1}\phi_k(z) \in \mathrm{span}\{\phi_l : l = k-1, k, k+1\}$
- ORF:(generic case = regular system: $\phi_k(\alpha_{k-1}) \neq 0$)

$$z^{-1}\phi_k(z) \in \mathrm{span}\{[1-\alpha_l/z]\phi_l(z) : l = k-1, k, k+1\}$$

- $\Phi = \Phi(zI - \mathbf{A})\mathbf{T} + [0 \ldots 0 \ (z-\alpha_{n+1})\phi_{n+1} b_{n+1}]$

$$\Phi = [\phi_0, \ldots, \phi_n], \mathbf{A} = \mathrm{diag}(1, \alpha_1, \ldots, \alpha_n),$$

$$\mathbf{T} = \begin{bmatrix} a_0 & b_0 & & & \\ b_0 & a_1 & b_1 & & \\ & b_1 & \ddots & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{bmatrix}$$

# Recurrence for ORF

discrete time

- OPS: $z^{-1}\phi_k(z) \in \mathrm{span}\{\phi_l : 0, 1, \ldots, k+1\}$
- ORF:(generic case = regular system: $\phi_k(\alpha_{k-1}) \neq 0$)

$$z^{-1}\phi_k(z) \in \mathrm{span}\{[1 - \alpha_l/z]\phi_l(z) : 0, 1, \ldots, k+1\}$$

- $\Phi = \Phi(zI - \mathbf{A})\mathbf{H} + [0 \ldots 0 \ (z - \alpha_{n+1})\phi_{n+1}\eta_{n+1,n+1}]$

$$\Phi = [\phi_0, \ldots, \phi_n], \mathbf{A} = \mathrm{diag}(1, \alpha_1, \ldots, \alpha_n),$$

$$\mathbf{H} = \begin{bmatrix} \eta_{01} & \eta_{02} & \eta_{03} & \cdots & \eta_{0,n+1} \\ \eta_{11} & \eta_{12} & \eta_{13} & & \eta_{1,n+1} \\ & \eta_{22} & \ddots & \ddots & \vdots \\ & & \ddots & \ddots & \eta_{n-1,n} \\ & & & \eta_{n,n} & \eta_{n,n+1} \end{bmatrix}$$

# Algorithm

Discrete time case:

- Szegő-type recurrence:
$$\begin{aligned} \phi_n \sigma_n &= z^{-1}\phi_{n-1} + \phi_{n-1}^* \gamma_n \\ \phi_n^* \sigma_n^H &= z^{-1}\phi_{n-1}\gamma_n^H + \phi_{n-1}^* \end{aligned}$$
$\sigma_n, \gamma_n$ (Schur parameters)
- Store the upper Hessenberg matrix in factored form $H = G_1 G_2 \cdots G_m$

$$G_k = \begin{bmatrix} I_{k-1} & & & \\ & -\gamma_k & \hat{\sigma}_k & \\ & \sigma_k & \hat{\gamma}_k & \\ & & & I_{\ldots} \end{bmatrix}$$

- Hessenberg "as simple" as tridiagonal.
- recursive update is possible (see later)
- loss of orthogonality may require reorthogonalization

# Algorithm

Discrete time case:

- Szegő-type recurrence:
$$\begin{aligned} \phi_n \sigma_n &= z^{-1}\phi_{n-1} + \phi^*_{n-1}\gamma_n \\ \phi^*_n \sigma^H_n &= z^{-1}\phi_{n-1}\gamma^H_n + \phi^*_{n-1} \end{aligned}$$
$\sigma_n, \gamma_n$ (Schur parameters)
- Store the upper Hessenberg matrix in factored form $H = G_1 G_2 \cdots G_m$

$$G_k = \begin{bmatrix} I_{k-1} & & & \\ & -\gamma_k & \hat{\sigma}_k & \\ & \sigma_k & \hat{\gamma}_k & \\ & & & I_{\ldots} \end{bmatrix}$$

- Hessenberg "as simple" as tridiagonal.
- recursive update is possible (see later)
- loss of orthogonality may require reorthogonalization

# Algorithm

Discrete time case:

- Szegő-type recurrence:
$$\begin{aligned} \phi_n \sigma_n &= z^{-1}\phi_{n-1} + \phi_{n-1}^* \gamma_n \\ \phi_n^* \sigma_n^H &= z^{-1}\phi_{n-1}\gamma_n^H + \phi_{n-1}^* \end{aligned}$$
$\sigma_n, \gamma_n$ (Schur parameters)
- Store the upper Hessenberg matrix in factored form $H = G_1 G_2 \cdots G_m$

$$G_k = \begin{bmatrix} I_{k-1} & & & \\ & -\gamma_k & \hat{\sigma}_k & \\ & \sigma_k & \hat{\gamma}_k & \\ & & & I_{\ldots} \end{bmatrix}$$

- Hessenberg "as simple" as tridiagonal.
- recursive update is possible (see later)
- loss of orthogonality may require reorthogonalization

# Algorithm

Discrete time case:

- Szegő-type recurrence:
$$\begin{aligned} \phi_n \sigma_n &= z^{-1}\phi_{n-1} + \phi^*_{n-1}\gamma_n \\ \phi^*_n \sigma^H_n &= z^{-1}\phi_{n-1}\gamma^H_n + \phi^*_{n-1} \end{aligned}$$
$\sigma_n, \gamma_n$ (Schur parameters)
- Store the upper Hessenberg matrix in factored form $H = G_1 G_2 \cdots G_m$

$$G_k = \begin{bmatrix} I_{k-1} & & & \\ & -\gamma_k & \hat{\sigma}_k & \\ & \sigma_k & \hat{\gamma}_k & \\ & & & I_{\ldots} \end{bmatrix}$$

- Hessenberg "as simple" as tridiagonal.
- recursive update is possible (see later)
- loss of orthogonality may require reorthogonalization

# Algorithm

Discrete time case:

- Szegő-type recurrence:
$$\begin{aligned} \phi_n \sigma_n &= z^{-1}\phi_{n-1} + \phi^*_{n-1}\gamma_n \\ \phi^*_n \sigma^H_n &= z^{-1}\phi_{n-1}\gamma^H_n + \phi^*_{n-1} \end{aligned}$$
$\sigma_n, \gamma_n$ (Schur parameters)
- Store the upper Hessenberg matrix in factored form $H = G_1 G_2 \cdots G_m$

$$G_k = \begin{bmatrix} I_{k-1} & & & \\ & -\gamma_k & \hat{\sigma}_k & \\ & \sigma_k & \hat{\gamma}_k & \\ & & & I_{\ldots} \end{bmatrix}$$

- Hessenberg "as simple" as tridiagonal.
- recursive update is possible (see later)
- loss of orthogonality may require reorthogonalization

# Compact notation

- $\Phi(z) = \Phi(z)(z\mathbf{I} - \mathbf{A})\mathbf{H} + [0, \ldots, 0, *]$

  $\Phi(z) = [\phi_0, \ldots, \phi_n(z)]$, $\mathbf{A} = \mathrm{diag}(1, \alpha_1, \ldots, \alpha_n)$,
  $\mathbf{H}$ Hessenberg
- for all $\mathbf{z} = (z_1, \ldots, z_N)$ gives

  $\mathbf{\Phi} = (\mathbf{Z\Phi} - \mathbf{\Phi A})\mathbf{H} + 0$

  with $\mathbf{Z} = \mathrm{diag}(\mathbf{z})$ and $\mathbf{\Phi} = \Phi(\mathbf{z}) = [\phi_i(z_k)]_{k=1:N}^{i=0:n}$
- $\mathbf{W\Phi} = (\mathbf{ZW\Phi} - \mathbf{W\Phi A})\mathbf{H}$
  $\mathbf{W} = \mathrm{diag}(w_1, \ldots, w_N)$, $\mathbf{WZ} = \mathbf{ZW} \Leftarrow$ diagonals commute
- set $\mathbf{W\Phi} = \mathbf{Q} \Rightarrow \mathbf{Q}^H\mathbf{Q} = \mathbf{I}$

$$\boxed{\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{H}}$$

# Compact notation

- $\Phi(z) = \Phi(z)(z\mathbf{I} - \mathbf{A})\mathbf{H} + [0, \ldots, 0, *]$

  $\Phi(z) = [\phi_0, \ldots, \phi_n(z)]$, $\mathbf{A} = \mathrm{diag}(1, \alpha_1, \ldots, \alpha_n)$,
  $\mathbf{H}$ Hessenberg
- for all $\mathbf{z} = (z_1, \ldots, z_N)$ gives

  $\mathbf{\Phi} = (\mathbf{Z\Phi} - \mathbf{\Phi A})\mathbf{H} + 0$

  with $\mathbf{Z} = \mathrm{diag}(\mathbf{z})$ and $\mathbf{\Phi} = \Phi(\mathbf{z}) = [\phi_i(z_k)]_{k=1:N}^{i=0:n}$
- $\mathbf{W\Phi} = (\mathbf{ZW\Phi} - \mathbf{W\Phi A})\mathbf{H}$

  $\mathbf{W} = \mathrm{diag}(w_1, \ldots, w_N)$, $\mathbf{WZ} = \mathbf{ZW} \Leftarrow$ diagonals commute
- set $\mathbf{W\Phi} = \mathbf{Q} \Rightarrow \mathbf{Q}^H\mathbf{Q} = \mathbf{I}$

$$\boxed{\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{H}}$$

# Compact notation

- $\Phi(z) = \Phi(z)(z\mathbf{I} - \mathbf{A})\mathbf{H} + [0, \ldots, 0, *]$

  $\Phi(z) = [\phi_0, \ldots, \phi_n(z)]$, $\mathbf{A} = \mathrm{diag}(1, \alpha_1, \ldots, \alpha_n)$,
  $\mathbf{H}$ Hessenberg
- for all $\mathbf{z} = (z_1, \ldots, z_N)$ gives

  $\mathbf{\Phi} = (\mathbf{Z\Phi} - \mathbf{\Phi A})\mathbf{H} + 0$

  with $\mathbf{Z} = \mathrm{diag}(\mathbf{z})$ and $\mathbf{\Phi} = \Phi(\mathbf{z}) = [\phi_i(z_k)]_{k=1:N}^{i=0:n}$
- $\mathbf{W\Phi} = (\mathbf{ZW\Phi} - \mathbf{W\Phi A})\mathbf{H}$
  $\mathbf{W} = \mathrm{diag}(w_1, \ldots, w_N)$, $\mathbf{WZ} = \mathbf{ZW} \Leftarrow$ diagonals commute
- set $\mathbf{W\Phi} = \mathbf{Q} \Rightarrow \mathbf{Q}^H\mathbf{Q} = \mathbf{I}$

$$\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{H}$$

# Compact notation

- $\Phi(z) = \Phi(z)(z\mathbf{I} - \mathbf{A})\mathbf{H} + [0, \ldots, 0, *]$

  $\Phi(z) = [\phi_0, \ldots, \phi_n(z)]$, $\mathbf{A} = \mathrm{diag}(1, \alpha_1, \ldots, \alpha_n)$,
  $\mathbf{H}$ Hessenberg
- for all $\mathbf{z} = (z_1, \ldots, z_N)$ gives

  $\mathbf{\Phi} = (\mathbf{Z\Phi} - \mathbf{\Phi A})\mathbf{H} + 0$

  with $\mathbf{Z} = \mathrm{diag}(\mathbf{z})$ and $\mathbf{\Phi} = \Phi(\mathbf{z}) = [\phi_i(z_k)]_{k=1:N}^{i=0:n}$
- $\mathbf{W\Phi} = (\mathbf{ZW\Phi} - \mathbf{W\Phi A})\mathbf{H}$
  $\mathbf{W} = \mathrm{diag}(w_1, \ldots, w_N)$, $\mathbf{WZ} = \mathbf{ZW} \Leftarrow$ diagonals commute
- set $\mathbf{W\Phi} = \mathbf{Q} \Rightarrow \mathbf{Q}^H\mathbf{Q} = \mathbf{I}$

$$\boxed{\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{H}}$$

# Example condition number

- Laguerre basis, $\alpha = 0.5$, $u(t) = 1$, $\omega_k \in [-\pi/6, \pi/6]$
- Condition number $N = 50$, $n = 20$, $\Rightarrow \kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = O(10^9)$
- Condition number $N = 300$, $n = 33$, $\Rightarrow \kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = O(10^{16})$
- Condition number appropriate ORF $\kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = \kappa(I) = 1$.

# Example condition number

- Laguerre basis, $\alpha = 0.5$, $u(t) = 1$, $\omega_k \in [-\pi/6, \pi/6]$
- Condition number $N = 50$, $n = 20$, $\Rightarrow \kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = O(10^9)$
- Condition number $N = 300$, $n = 33$, $\Rightarrow \kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = O(10^{16})$
- Condition number appropriate ORF $\kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = \kappa(I) = 1$.

# Example condition number

- Laguerre basis, $\alpha = 0.5$, $u(t) = 1$, $\omega_k \in [-\pi/6, \pi/6]$
- Condition number $N = 50$, $n = 20$, $\Rightarrow \kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = O(10^9)$
- Condition number $N = 300$, $n = 33$, $\Rightarrow \kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = O(10^{16})$
- Condition number appropriate ORF $\kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = \kappa(I) = 1$.

# Example condition number

- Laguerre basis, $\alpha = 0.5$, $u(t) = 1$, $\omega_k \in [-\pi/6, \pi/6]$
- Condition number $N = 50$, $n = 20$, $\Rightarrow \kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = O(10^9)$
- Condition number $N = 300$, $n = 33$, $\Rightarrow \kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = O(10^{16})$
- Condition number appropriate ORF $\kappa(\mathbf{\Phi}^H \mathbf{\Phi}) = \kappa(I) = 1$.

# Statistics

- $Y(z) = G_0(z)U(z) + V(z),\ V(z) = H(z)E(z)$
- Let $G_0(z) = \sum_{k=0}^{n} \lambda_k^* \phi_k(z) + Z(z),\ Z(z) = \sum_{k>n} \lambda_k^* \phi_k(z)$
- $\mathrm{plim}_{N\to\infty} \boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} = \boldsymbol{\lambda}^* + \boldsymbol{\Phi}^H \mathbf{w}\mathbf{Z}$
- $\sqrt{N}(\boldsymbol{\lambda} - \hat{\boldsymbol{\lambda}}) \sim \mathcal{N}(0, P)$
- $P_{ik} = \sum_l \overline{\phi_{i-1}(z_l)} \phi_{k-1}(z_l) |w_k|^2 \frac{F_v(z_l)}{F_u(z_l)}$

# Statistics

- $Y(z) = G_0(z)U(z) + V(z),\ V(z) = H(z)E(z)$
- Let $G_0(z) = \sum_{k=0}^{n} \lambda_k^* \phi_k(z) + Z(z),\ Z(z) = \sum_{k>n} \lambda_k^* \phi_k(z)$
- $\mathrm{plim}_{N\to\infty} \boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} = \boldsymbol{\lambda}^* + \boldsymbol{\Phi}^H \mathbf{w}\mathbf{Z}$
- $\sqrt{N}(\boldsymbol{\lambda} - \hat{\boldsymbol{\lambda}}) \sim \mathcal{N}(0, P)$
- $P_{ik} = \sum_l \overline{\phi_{i-1}(z_l)} \phi_{k-1}(z_l) |w_k|^2 \frac{F_v(z_l)}{F_u(z_l)}$

# Statistics

- $Y(z) = G_0(z)U(z) + V(z),\ V(z) = H(z)E(z)$
- Let $G_0(z) = \sum_{k=0}^{n} \lambda_k^* \phi_k(z) + Z(z),\ Z(z) = \sum_{k>n} \lambda_k^* \phi_k(z)$
- $\mathrm{plim}_{N\to\infty} \boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} = \boldsymbol{\lambda}^* + \mathbf{\Phi}^H \mathbf{w} \mathbf{Z}$
- $\sqrt{N}(\boldsymbol{\lambda} - \hat{\boldsymbol{\lambda}}) \sim \mathcal{N}(0, P)$
- $P_{ik} = \sum_l \overline{\phi_{i-1}(z_l)} \phi_{k-1}(z_l) |w_k|^2 \frac{F_v(z_l)}{F_u(z_l)}$

# Statistics

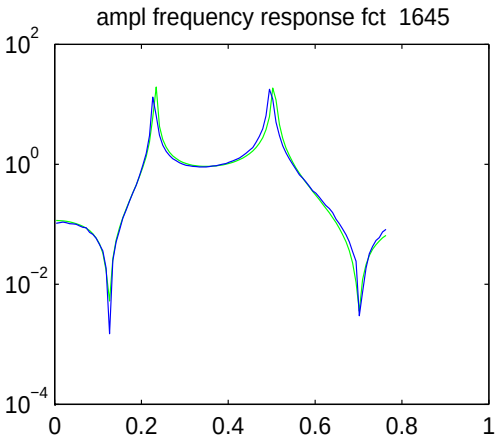
- $Y(z) = G_0(z)U(z) + V(z),\ V(z) = H(z)E(z)$
- Let $G_0(z) = \sum_{k=0}^{n} \lambda_k^* \phi_k(z) + Z(z),\ Z(z) = \sum_{k>n} \lambda_k^* \phi_k(z)$
- $\mathrm{plim}_{N\to\infty} \boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} = \boldsymbol{\lambda}^* + \mathbf{\Phi}^H \mathbf{w} \mathbf{Z}$
- $\sqrt{N}(\boldsymbol{\lambda} - \hat{\boldsymbol{\lambda}}) \sim \mathcal{N}(0, P)$
- $P_{ik} = \sum_l \overline{\phi_{i-1}(z_l)} \phi_{k-1}(z_l) |w_k|^2 \frac{F_v(z_l)}{F_u(z_l)}$

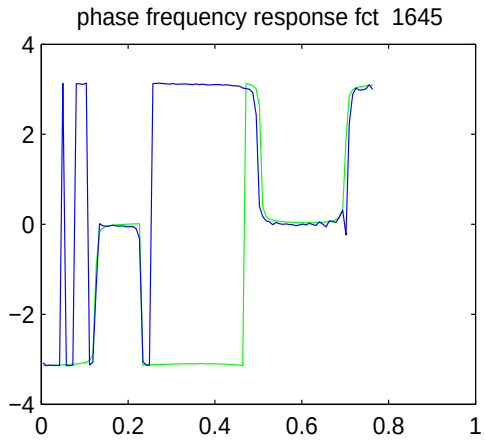
# Statistics

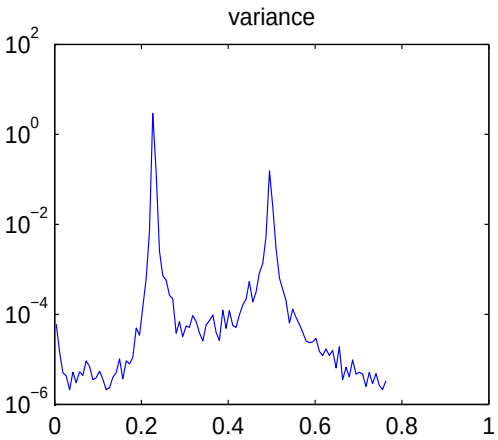
- $Y(z) = G_0(z)U(z) + V(z),\ V(z) = H(z)E(z)$
- Let $G_0(z) = \sum_{k=0}^{n} \lambda_k^* \phi_k(z) + Z(z),\ Z(z) = \sum_{k>n} \lambda_k^* \phi_k(z)$
- $\mathrm{plim}_{N\to\infty} \boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} = \boldsymbol{\lambda}^* + \mathbf{\Phi}^H \mathbf{w} \mathbf{Z}$
- $\sqrt{N}(\boldsymbol{\lambda} - \hat{\boldsymbol{\lambda}}) \sim \mathcal{N}(0, P)$
- $P_{ik} = \sum_l \overline{\phi_{i-1}(z_l)} \phi_{k-1}(z_l) |w_k|^2 \frac{F_v(z_l)}{F_u(z_l)}$

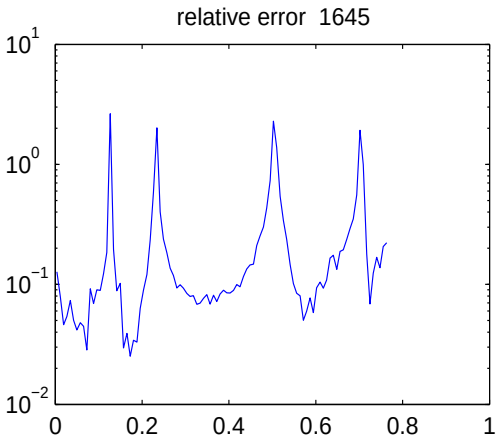
# Robotarm

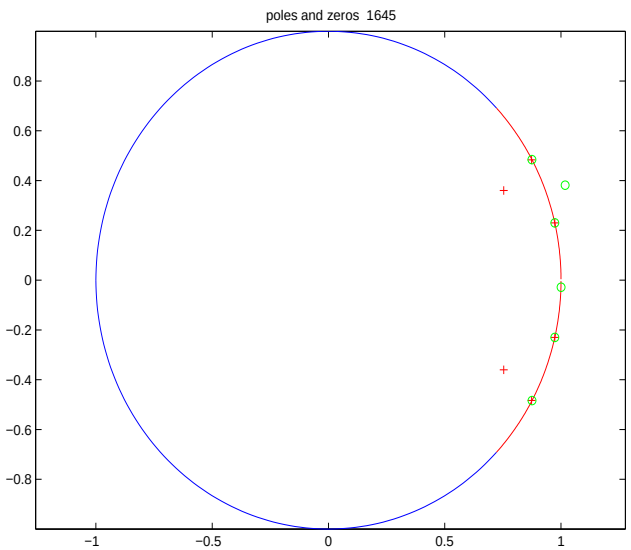
order [6/6], 100 data points, 200Hz, discrete time



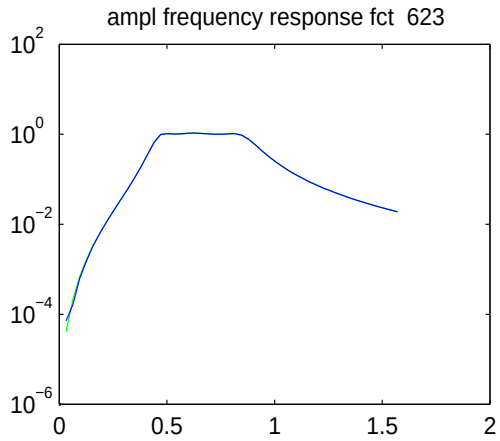



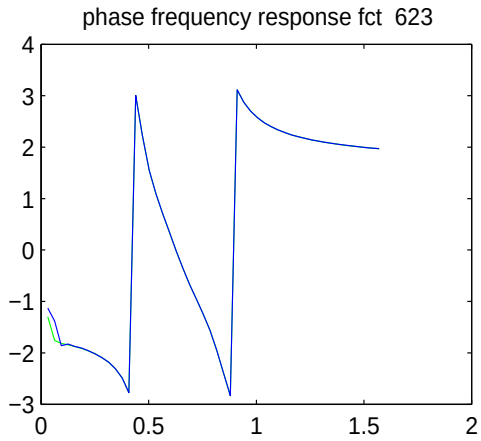



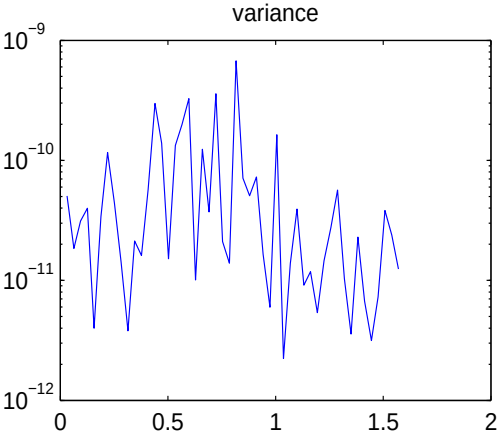



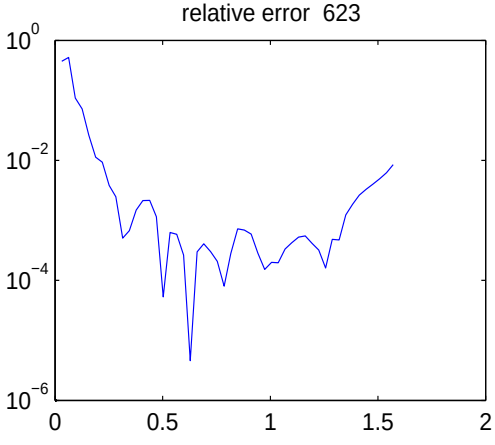
# Bandpass filter

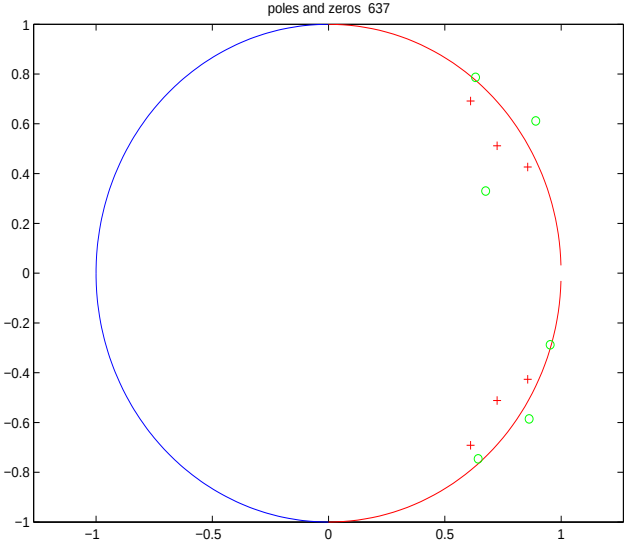
order [6/6], 50 data points, 20KHz, discrete time



# References ORF



A. Bultheel, P. González-Vera, E. Hendriksen, and O. Njåstad. *Orthogonal rational functions*. Cambridge University Press, 1999.


A. Bultheel, M. Van Barel, P. Van gucht. *Orthogonal bases in discrete least squares rational approximation.* *J. Comput. Appl. Math.*, 164-165 (2004) 175-194.


P. Van gucht, A. Bultheel. *Using orthogonal rational functions for system identification: numerical aspects*, *IEEE Trans. Automat. Control*, 48 (2003) 705-709.


A. Bultheel, P. González-Vera, E. Hendriksen, and O. Njåstad. *Orthogonal rational functions and tridiagonal matrices*, *J. Comput. Appl. Math.* 153 (2003) 89-97.

# Summary

- Identification problem
- Orthogonal Rational Functions
- Orthogonal Vector Polynomials

# System

- in: $u(k) = u_0(k) + n_u(k), \quad n_u$ zero mean i.i.d., var$=\sigma_u^2$
- out: $y(k) = y_0(k) + n_y(k), \quad n_y$ zero mean i.i.d., var$=\sigma_y^2$
- model : $G = B/A, \quad B$ and $A$ polynomials (in $z^{-1}$)
- data : in frequency domain by DFT
- cost : ML for EV (resistor example)

$$V_{ML} = \sum_{k=1}^{N} \frac{(Y(k) - Y_0(k))^2}{\sigma_Y^2(k)} + \frac{(U(k) - U_0(k))^2}{\sigma_U^2(k)}$$

constraint $A(z_k)Y_0(k) = B(z_k)U_0(k),\ k = 1, \ldots, N$

- minimize w.r.t. $A, B, Y_0, U_0$

# System

- in: $u(k) = u_0(k) + n_u(k)$, $n_u$ zero mean i.i.d., var$=\sigma_u^2$
- out: $y(k) = y_0(k) + n_y(k)$, $n_y$ zero mean i.i.d., var$=\sigma_y^2$
- model : $G = B/A$, $B$ and $A$ polynomials (in $z^{-1}$)
- data : in frequency domain by DFT
- cost : ML for EV (resistor example)

$$V_{ML} = \sum_{k=1}^{N} \frac{(Y(k) - Y_0(k))^2}{\sigma_Y^2(k)} + \frac{(U(k) - U_0(k))^2}{\sigma_U^2(k)}$$

constraint $A(z_k)Y_0(k) = B(z_k)U_0(k)$, $k = 1, \ldots, N$

- minimize w.r.t. $A, B, Y_0, U_0$

# System

- in: $u(k) = u_0(k) + n_u(k)$, $n_u$ zero mean i.i.d., var$=\sigma_u^2$
- out: $y(k) = y_0(k) + n_y(k)$, $n_y$ zero mean i.i.d., var$=\sigma_y^2$
- model : $G = B/A$, $B$ and $A$ polynomials (in $z^{-1}$)
- data : in frequency domain by DFT
- cost : ML for EV (resistor example)

$$V_{ML} = \sum_{k=1}^{N} \frac{(Y(k) - Y_0(k))^2}{\sigma_Y^2(k)} + \frac{(U(k) - U_0(k))^2}{\sigma_U^2(k)}$$

constraint $A(z_k)Y_0(k) = B(z_k)U_0(k)$, $k = 1, \ldots, N$

- minimize w.r.t. $A, B, Y_0, U_0$

# System

- in: $u(k) = u_0(k) + n_u(k)$, $n_u$ zero mean i.i.d., var$=\sigma_u^2$
- out: $y(k) = y_0(k) + n_y(k)$, $n_y$ zero mean i.i.d., var$=\sigma_y^2$
- model : $G = B/A$, $B$ and $A$ polynomials (in $z^{-1}$)
- data : in frequency domain by DFT
- cost : ML for EV (resistor example)

$$V_{ML} = \sum_{k=1}^{N} \frac{(Y(k) - Y_0(k))^2}{\sigma_Y^2(k)} + \frac{(U(k) - U_0(k))^2}{\sigma_U^2(k)}$$

constraint $A(z_k)Y_0(k) = B(z_k)U_0(k)$, $k = 1, \ldots, N$

- minimize w.r.t. $A, B, Y_0, U_0$

# System

- in: $u(k) = u_0(k) + n_u(k)$, $n_u$ zero mean i.i.d., var=$\sigma_u^2$
- out: $y(k) = y_0(k) + n_y(k)$, $n_y$ zero mean i.i.d., var=$\sigma_y^2$
- model : $G = B/A$, $B$ and $A$ polynomials (in $z^{-1}$)
- data : in frequency domain by DFT
- cost : ML for EV (resistor example)

$$V_{ML} = \sum_{k=1}^{N} \frac{(Y(k) - Y_0(k))^2}{\sigma_Y^2(k)} + \frac{(U(k) - U_0(k))^2}{\sigma_U^2(k)}$$

constraint $A(z_k)Y_0(k) = B(z_k)U_0(k)$, $k = 1, \ldots, N$

- minimize w.r.t. $A, B, Y_0, U_0$

# System

- in: $u(k) = u_0(k) + n_u(k)$, $n_u$ zero mean i.i.d., var$=\sigma_u^2$
- out: $y(k) = y_0(k) + n_y(k)$, $n_y$ zero mean i.i.d., var$=\sigma_y^2$
- model : $G = B/A$, $B$ and $A$ polynomials (in $z^{-1}$)
- data : in frequency domain by DFT
- cost : ML for EV (resistor example)

$$V_{ML} = \sum_{k=1}^{N} \frac{(Y(k) - Y_0(k))^2}{\sigma_Y^2(k)} + \frac{(U(k) - U_0(k))^2}{\sigma_U^2(k)}$$

constraint $A(z_k)Y_0(k) = B(z_k)U_0(k)$, $k = 1, \ldots, N$

- minimize w.r.t. $A, B, Y_0, U_0$

# COST = ML

COST = Maximum Likelihood

- $\theta$ : $n_a + n_b + 1$ real model parameters
- $U_p, Y_p$ : $2N$ complex model parameters
- assumes Gaussian noise with known variance

# COST = ML

COST = Maximum Likelihood

$$Y \textbf{ and } U \textbf{ not correlated}$$

$$\Rightarrow \sigma^2_{YU}(k) = 0,$$

$$V_N = \frac{1}{N} \sum_{k=1}^{N} \frac{|Y(k) - Y_p(k)|^2}{\sigma^2_Y(k)} + \frac{|U(k) - U_p(k)|^2}{\sigma^2_U(k)}.$$

- $\theta$ : $n_a + n_b + 1$ real model parameters
- $U_p, Y_p$ : $2N$ complex model parameters
- assumes Gaussian noise with known variance

# COST = ML

COST = Maximum Likelihood

**$Y$ and $U$ not correlated**

$$\Rightarrow \sigma^2_{YU}(k) = 0,$$

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \frac{|Y(k) - Y_p(k)|^2}{\sigma^2_Y(k)} + \frac{|U(k) - U_p(k)|^2}{\sigma^2_U(k)}.$$

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}^H C(k)^{-1} \begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}, \quad \begin{matrix} e_y = Y - Y_p \\ e_u = U - U_p \end{matrix}$$

with $C = \begin{bmatrix} \sigma_y^2 & 0 \\ 0 & \sigma_u^2 \end{bmatrix}$ and $A(z_k)Y_p(k) = B(z_k)U_p(k)$

- $\theta$ : $n_a + n_b + 1$ real model parameters
- $U_p, Y_p$ : $2N$ complex model parameters

# COST = ML



Fisher 1912

COST = Maximum Likelihood

**in general**

$$V_N = \frac{1}{N}\sum_{k=1}^{N}\begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}^H C(k)^{-1}\begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}, \quad \begin{aligned} e_y &= Y - Y_p \\ e_u &= U - U_p \end{aligned}$$

with $C = \begin{bmatrix} \sigma_y^2(k) & \sigma_{yu}^2(k) \\ \sigma_{uy}^2(k) & \sigma_u^2(k) \end{bmatrix}$ (correlation)

- $\theta$ : $n_a + n_b + 1$ real model parameters
- $U_p, Y_p$ : $2N$ complex model parameters
- assumes Gaussian noise with known variance

# COST = ML



Fisher 1912

COST = Maximum Likelihood

**in general**

$$V_N = \frac{1}{N}\sum_{k=1}^{N}\begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}^H C(k)^{-1}\begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}, \quad \begin{aligned} e_y &= Y - Y_p \\ e_u &= U - U_p \end{aligned}$$

with $C = \begin{bmatrix} \sigma_y^2(k) & \sigma_{yu}^2(k) \\ \sigma_{uy}^2(k) & \sigma_u^2(k) \end{bmatrix}$ (correlation)

and $A(z_k)\ Y_p(k) = B(z_k)\ U_p(k)$ (constraint)

- $\theta$ : $n_a + n_b + 1$ real model parameters
- $U_p, Y_p$ : $2N$ complex model parameters
- assumes Gaussian noise with known variance

# COST = ML



Fisher 1912

COST = Maximum Likelihood

**in general**

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}^H C(k)^{-1} \begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}, \quad \begin{aligned} e_y &= Y - Y_p \\ e_u &= U - U_p \end{aligned}$$

with $C = \begin{bmatrix} \sigma_y^2(k) & \sigma_{yu}^2(k) \\ \sigma_{uy}^2(k) & \sigma_u^2(k) \end{bmatrix}$ (correlation)

and $A(z_k)\ Y_p(k) = B(z_k)\ U_p(k)$ (constraint)

- $\theta$ : $n_a + n_b + 1$ real model parameters
- $U_p, Y_p$ : $2N$ complex model parameters
- assumes Gaussian noise with known variance

# COST = ML



Fisher 1912

COST = Maximum Likelihood

**in general**

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}^H C(k)^{-1} \begin{bmatrix} e_y(k) \\ e_u(k) \end{bmatrix}, \quad \begin{aligned} e_y &= Y - Y_p \\ e_u &= U - U_p \end{aligned}$$

with $C = \begin{bmatrix} \sigma_y^2(k) & \sigma_{yu}^2(k) \\ \sigma_{uy}^2(k) & \sigma_u^2(k) \end{bmatrix}$ (correlation)

and $A(z_k)\ Y_p(k) = B(z_k)\ U_p(k)$ (constraint)

- $\theta$ : $n_a + n_b + 1$ real model parameters
- $U_p, Y_p$ : $2N$ complex model parameters
- assumes Gaussian noise with known variance

# Special cases

$$Y(z) = G(z)U(z)$$

**FRF data**

assume $U(z) = 1$ then $G(z) = Y(z)$
$\Rightarrow \sigma_U^2 = 0$ and $\sigma_Y^2(k) = \sigma_G^2(k)$

$$V_N = \frac{1}{N} \sum_{k=1}^{N} \frac{|G(z_k) - G(z_k, \theta)|^2}{\sigma_G^2(k)}$$

# Special cases

$$Y(z) = G(z)U(z)$$

**Exact input**

$$\Rightarrow \sigma_U^2(k) = 0,\ \sigma_{YU}^2(k) = 0$$

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \frac{|Y(k) - G(z_k, \theta)U(k)|^2}{\sigma_Y^2(k)}.$$

# Special cases

$$Y(z) = G(z)U(z)$$

**$Y$ and $U$ not correlated**

$$\Rightarrow \sigma^2_{YU}(k) = \sigma^2_{UY}(k) = 0,$$

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \frac{|Y(k) - Y_p(k)|^2}{\sigma^2_Y(k)} + \frac{|U(k) - U_p(k)|^2}{\sigma^2_U(k)}.$$

# COST = ML

COST = Maximum Likelihood after elimination of $Y_p$ and $U_p$ usualy replaced by

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \frac{|A(z_k)Y(k) - B(z_k)U(k)|^2}{\sigma_Y^2(k)|A(z_k)|^2 + |B(z_k)|^2\sigma_U^2(k) - 2\mathrm{Re}\{\sigma_{YU}^2 A(z_k)\overline{B(z_k)}\}}.$$

The denominator is the variance of the numerator.

Define residue $R = AY - BU = \underbrace{[Y \ -U]}_{D}\underbrace{[A \ B]^T}_{P} = DP$

then by linearity of data, eliminate $Y_p, U_p$:

$$\underbrace{D\,[\mathrm{E}(D^H D)]^{-1} D^H}_{C} \to DP[P^H \mathrm{E}(D^H D)P]^{-1} P^H D^H = \frac{|DP|^2}{\mathrm{E}(|DP|^2)}.$$

$$DC_D^{-1}D^H \to RC_R^{-1}R = \frac{|R|^2}{\sigma_R^2}$$

# COST = ML

COST = Maximum Likelihood after elimination of $Y_p$ and $U_p$ usualy replaced by

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \frac{|A(z_k)Y(k) - B(z_k)U(k)|^2}{\sigma_Y^2(k)|A(z_k)|^2 + |B(z_k)|^2\sigma_U^2(k) - 2\mathrm{Re}\{\sigma_{YU}^2 A(z_k)\overline{B(z_k)}\}}.$$

The denominator is the variance of the numerator.

Define residue $R = AY - BU = \underbrace{[Y \ -U]}_{D}\underbrace{[A \ B]^T}_{P} = DP$

then by linearity of data, eliminate $Y_p, U_p$:

$$D\underbrace{[\mathrm{E}(D^H D)]}_{C}{}^{-1}D^H \to DP[P^H\mathrm{E}(D^H D)P]^{-1}P^H D^H = \frac{|DP|^2}{\mathrm{E}(|DP|^2)}.$$

$$DC_D^{-1}D^H \to RC_R^{-1}R = \frac{|R|^2}{\sigma_R^2}$$

# COST = ML

COST = Maximum Likelihood after elimination of $Y_p$ and $U_p$ usualy replaced by

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \frac{|A(z_k)Y(k) - B(z_k)U(k)|^2}{\sigma_Y^2(k)|A(z_k)|^2 + |B(z_k)|^2\sigma_U^2(k) - 2\mathrm{Re}\{\sigma_{YU}^2 A(z_k)\overline{B(z_k)}\}}.$$

The denominator is the variance of the numerator.
Define residue $R = AY - BU = \underbrace{[Y \ -U]}_{D}\underbrace{[A \ B]^T}_{P} = DP$

then by linearity of data, eliminate $Y_p, U_p$:

$$D\underbrace{[\mathrm{E}(D^H D)]}_{C}{}^{-1}D^H \to DP[P^H\mathrm{E}(D^H D)P]^{-1}P^H D^H = \frac{|DP|^2}{\mathrm{E}(|DP|^2)}.$$

$$DC_D^{-1}D^H \to RC_R^{-1}R = \frac{|R|^2}{\sigma_R^2}$$

# COST = ML

COST = Maximum Likelihood after elimination of $Y_p$ and $U_p$ usualy replaced by

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \frac{|A(z_k)Y(k) - B(z_k)U(k)|^2}{\sigma_Y^2(k)|A(z_k)|^2 + |B(z_k)|^2\sigma_U^2(k) - 2\mathrm{Re}\{\sigma_{YU}^2 A(z_k)\overline{B(z_k)}\}}.$$

The denominator is the variance of the numerator.
Define residue $R = AY - BU = \underbrace{[Y \ -U]}_{D}\underbrace{[A \ B]^T}_{P} = DP$

then by linearity of data, eliminate $Y_p, U_p$:

$$D\underbrace{[\mathrm{E}(D^H D)]}_{C}{}^{-1}D^H \to DP[P^H\mathrm{E}(D^H D)P]^{-1}P^H D^H = \frac{|DP|^2}{\mathrm{E}(|DP|^2)}.$$

$$DC_D^{-1}D^H \to RC_R^{-1}R = \frac{|R|^2}{\sigma_R^2}$$

# COST = ML

COST = Maximum Likelihood after elimination of $Y_p$ and $U_p$ usualy replaced by

$$V_N = \frac{1}{N}\sum_{k=1}^{N} \frac{|A(z_k)Y(k) - B(z_k)U(k)|^2}{\sigma_Y^2(k)|A(z_k)|^2 + |B(z_k)|^2\sigma_U^2(k) - 2\mathrm{Re}\{\sigma_{YU}^2 A(z_k)\overline{B(z_k)}\}}.$$

The denominator is the variance of the numerator.
Define residue $R = AY - BU = \underbrace{[Y \ -U]}_{D}\underbrace{[A \ B]^T}_{P} = DP$

then by linearity of data, eliminate $Y_p, U_p$:

$$D\underbrace{[\mathrm{E}(D^H D)]}_{C}{}^{-1}D^H \to DP[P^H\mathrm{E}(D^H D)P]^{-1}P^H D^H = \frac{|DP|^2}{\mathrm{E}(|DP|^2)}.$$

$$DC_D^{-1}D^H \to RC_R^{-1}R = \frac{|R|^2}{\sigma_R^2}$$

# Matrix Weights

Matrix valued weights

$R = [Y \quad -U][A \quad B]^T = D \cdot P,$

data $D = [Y \quad -U]$, parameters $P = [A \quad B]^T$.

$$\|R\|_{w_r} = \|D \cdot P\|_{w_r} = \|w_r D \cdot P\| = \|W \cdot P\| = \|P\|_W$$

$$\|P\|_W^2 = \sum_{k=1}^{N} P^H(z_k)[W^H(k)W(k)]P(z_k), \quad P(z_k) = \begin{bmatrix} A(z_k) \\ B(z_k) \end{bmatrix}$$

vector valued polynomial $P = \begin{bmatrix} A \\ B \end{bmatrix}$

matrix valued weight $W^H W = w_r^2 \begin{bmatrix} Y^2 & -YU \\ -UY & U^2 \end{bmatrix}$

# Matrix Weights

Matrix valued weights

$R = [Y \quad -U][A \quad B]^T = D \cdot P,$

data $D = [Y \quad -U]$, parameters $P = [A \quad B]^T$.

$$\|R\|_{w_r} = \|D \cdot P\|_{w_r} = \|w_r D \cdot P\| = \|W \cdot P\| = \|P\|_W$$

$$W = w_k[Y \quad -U]$$

$$\|P\|_W^2 = \sum_{k=1}^{N} P^H(z_k)[W^H(k)W(k)]P(z_k), \quad P(z_k) = \begin{bmatrix} A(z_k) \\ B(z_k) \end{bmatrix}$$

vector valued polynomial $P = \begin{bmatrix} A \\ B \end{bmatrix}$

matrix valued weight $W^H W = w_r^2 \begin{bmatrix} Y^2 & -YU \\ -UY & U^2 \end{bmatrix}$

# Matrix Weights

Matrix valued weights

$R = [Y \quad -U][A \quad B]^T = D \cdot P,$

data $D = [Y \quad -U]$, parameters $P = [A \quad B]^T$.

$$\|R\|_{w_r} = \|D \cdot P\|_{w_r} = \|w_r D \cdot P\| = \|W \cdot P\| = \|P\|_W$$

$$\|P\|_W^2 = \sum_{k=1}^{N} P^H(z_k)[W^H(k)W(k)]P(z_k), \quad P(z_k) = \begin{bmatrix} A(z_k) \\ B(z_k) \end{bmatrix}$$

vector valued polynomial $P = \begin{bmatrix} A \\ B \end{bmatrix}$

matrix valued weight $W^H W = w_r^2 \begin{bmatrix} Y^2 & -YU \\ -UY & U^2 \end{bmatrix}$

# Matrix Weights

Matrix valued weights

$R = [Y \quad -U][A \quad B]^T = D \cdot P,$

data $D = [Y \quad -U]$, parameters $P = [A \quad B]^T$.

$$\|R\|_{w_r} = \|D \cdot P\|_{w_r} = \|w_r D \cdot P\| = \|W \cdot P\| = \|P\|_W$$

$$\|P\|_W^2 = \sum_{k=1}^{N} P^H(z_k)[W^H(k)W(k)]P(z_k), \quad P(z_k) = \begin{bmatrix} A(z_k) \\ B(z_k) \end{bmatrix}$$

vector valued polynomial $P = \begin{bmatrix} A \\ B \end{bmatrix}$

matrix valued weight $W^H W = w_r^2 \begin{bmatrix} Y^2 & -YU \\ -UY & U^2 \end{bmatrix}$

# Nonlinear problem

- Weights depend on solution $P$: $W(A, B, \sigma_Y^2, \sigma_U^2, \sigma_{YU}^2)$
- No convergence to global min guaranteed, unless good starting point
- Jacobian of min problem can be very ill conditioned unless orthogonal basis for representation of $P = (A, B)^T$ is chosen
- Hence
  (1) choose some $(A, B) \rightarrow$
  (2) compute weight $\rightarrow$
  (3) find solution of WLLS $= (A, B) \rightarrow$
  (4) recompute weight $\rightarrow$
  (5) $\cdots$
- $=$ Sanathanan-Koerner (1963)

# Nonlinear problem

- Weights depend on solution $P$: $W(A, B, \sigma_Y^2, \sigma_U^2, \sigma_{YU}^2)$
- No convergence to global min guaranteed, unless good starting point
- Jacobian of min problem can be very ill conditioned unless orthogonal basis for representation of $P = (A, B)^T$ is chosen
- Hence
  (1) choose some $(A, B) \rightarrow$
  (2) compute weight $\rightarrow$
  (3) find solution of WLLS $= (A, B) \rightarrow$
  (4) recompute weight $\rightarrow$
  (5) $\cdots$
- $=$ Sanathanan-Koerner (1963)

# Nonlinear problem

- Weights depend on solution $P$: $W(A, B, \sigma_Y^2, \sigma_U^2, \sigma_{YU}^2)$
- No convergence to global min guaranteed, unless good starting point
- Jacobian of min problem can be very ill conditioned unless orthogonal basis for representation of $P = (A, B)^T$ is chosen

(1) choose some $(A, B) \rightarrow$
(2) compute weight $\rightarrow$

- Hence (3) find solution of WLLS $= (A, B) \rightarrow$
(4) recompute weight $\rightarrow$
(5) $\cdots$

- $=$ Sanathanan-Koerner (1963)

# Nonlinear problem

- Weights depend on solution $P$: $W(A, B, \sigma_Y^2, \sigma_U^2, \sigma_{YU}^2)$
- No convergence to global min guaranteed, unless good starting point
- Jacobian of min problem can be very ill conditioned unless orthogonal basis for representation of $P = (A, B)^T$ is chosen

(1) choose some $(A, B) \rightarrow$
(2) compute weight $\rightarrow$

- Hence (3) find solution of WLLS $= (A, B) \rightarrow$
(4) recompute weight $\rightarrow$
(5) $\cdots$

- $=$ Sanathanan-Koerner (1963)

# Nonlinear problem

- Weights depend on solution $P$: $W(A, B, \sigma_Y^2, \sigma_U^2, \sigma_{YU}^2)$
- No convergence to global min guaranteed, unless good starting point
- Jacobian of min problem can be very ill conditioned unless orthogonal basis for representation of $P = (A, B)^T$ is chosen

  (1) choose some $(A, B) \rightarrow$
  (2) compute weight $\rightarrow$
- Hence (3) find solution of WLLS $= (A, B) \rightarrow$
  (4) recompute weight $\rightarrow$
  (5) $\cdots$
- $=$ Sanathanan-Koerner (1963)

# Combined method

- Sananthanan-Koerner method

  Start: with $W_k = W_k(1)$

  Iterate: solve for $A, B$, and set $W_k = W_k(A, B)$
  adapt basis to weights and get new $(A, B)$

  May not converge to the solution of the nonlinear problem!

- Needs a true Gauss-Newton-like iteration of nonlinear ML problem
- This gives a correction on the parameters but if the basis changes, the correction can not be added!
- solution: fix the basis and do
  Close: Solve ML problem with fixed basis.
- condition number grows only moderately

# Combined method

- Sananthanan-Koerner method

  Start: with $W_k = W_k(1)$

  Iterate: solve for $A, B$, and set $W_k = W_k(A, B)$
  adapt basis to weights and get new $(A, B)$

-  May not converge to the solution of the nonlinear problem!
- Needs a true Gauss-Newton-like iteration of nonlinear ML problem
- This gives a correction on the parameters but if the basis changes, the correction can not be added!
- solution: fix the basis and do
  Close: Solve ML problem with fixed basis.
- condition number grows only moderately

# Combined method

- Sananthanan-Koerner method

  Start: with $W_k = W_k(1)$

  Iterate: solve for $A, B$, and set $W_k = W_k(A, B)$
  adapt basis to weights and get new $(A, B)$

-  May not converge to the solution of the nonlinear problem!
- Needs a true Gauss-Newton-like iteration of nonlinear ML problem
- This gives a correction on the parameters but if the basis changes, the correction can not be added!
- solution: fix the basis and do
  Close: Solve ML problem with fixed basis.
- condition number grows only moderately

# Combined method

- Sananthanan-Koerner method

  Start: with $W_k = W_k(1)$

  Iterate: solve for $A, B$, and set $W_k = W_k(A, B)$
  adapt basis to weights and get new $(A, B)$

-  May not converge to the solution of the nonlinear problem!
- Needs a true Gauss-Newton-like iteration of nonlinear ML problem
- This gives a correction on the parameters but if the basis changes, the correction can not be added!
- solution: fix the basis and do
  Close: Solve ML problem with fixed basis.
- condition number grows only moderately

# Combined method

- Sananthanan-Koerner method

  Start: with $W_k = W_k(1)$

  Iterate: solve for $A, B$, and set $W_k = W_k(A, B)$
  adapt basis to weights and get new $(A, B)$

-  May not converge to the solution of the nonlinear problem!
- Needs a true Gauss-Newton-like iteration of nonlinear ML problem
- This gives a correction on the parameters but if the basis changes, the correction can not be added!
- solution: fix the basis and do
  Close: Solve ML problem with fixed basis.
- condition number grows only moderately

# Combined method

- Sananthanan-Koerner method

  Start: with $W_k = W_k(1)$

  Iterate: solve for $A, B$, and set $W_k = W_k(A, B)$
  adapt basis to weights and get new $(A, B)$

-  May not converge to the solution of the nonlinear problem!
- Needs a true Gauss-Newton-like iteration of nonlinear ML problem
- This gives a correction on the parameters but if the basis changes, the correction can not be added!
- solution: fix the basis and do
  Close: Solve ML problem with fixed basis.
- condition number grows only moderately

# Solution is OVP

- $\min \|P\|_W$ needs some normalisation.
- If $n = \max\{\deg(A), \deg(B)\}$, then $P$ is of degree $n$. Choose it "monic" (leading coefficient 1).
- What is te monic polynomial of degree $n$ with least possible norm?
- The monic orthogonal polynomial of degree $n$.

# Solution is OVP

- min $\|P\|_W$ needs some normalisation.
- If $n = \max\{\deg(A), \deg(B)\}$, then $P$ is of degree $n$. Choose it "monic" (leading coefficient 1).
- What is te monic polynomial of degree $n$ with least possible norm?
- The monic orthogonal polynomial of degree $n$.

# Solution is OVP

- min $\|P\|_W$ needs some normalisation.
- If $n = \max\{\deg(A), \deg(B)\}$, then $P$ is of degree $n$. Choose it "monic" (leading coefficient 1).
- What is te monic polynomial of degree $n$ with least possible norm?
- The monic orthogonal polynomial of degree $n$.

# Solution is OVP

- min $\|P\|_W$ needs some normalisation.
- If $n = \max\{\deg(A), \deg(B)\}$, then $P$ is of degree $n$. Choose it "monic" (leading coefficient 1).
- What is te monic polynomial of degree $n$ with least possible norm?
- The monic orthogonal polynomial of degree $n$.

# Note 1

## Matrix-vector notation

$$\langle f, g \rangle_W = \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\mathsf{I}-\mathsf{I})$$

$$= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\square\,\mathsf{I})$$

$$= \mathbf{f}^H \mathbf{W}^H \mathbf{W} \mathbf{g}$$

$$= [- \cdots -] \begin{bmatrix} \square & & \\ & \ddots & \\ & & \square \end{bmatrix} \begin{bmatrix} \mathsf{I} \\ \vdots \\ \mathsf{I} \end{bmatrix}$$

$$\mathbf{f} = \begin{bmatrix} f(z_1) \\ \vdots \\ f(z_N) \end{bmatrix}, \mathbf{W}^H \mathbf{W} = \begin{bmatrix} W_1^H W_1 & & \\ & \ddots & \\ & & W_N^H W_N \end{bmatrix}, \mathbf{g} = \begin{bmatrix} g(z_1) \\ \vdots \\ g(z_N) \end{bmatrix}$$

# Note 1

## Matrix-vector notation

$$
\begin{aligned}
\langle f, g \rangle_W &= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\mathsf{I}-\mathsf{I}) \\
&= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\square\mathsf{I}) \\
&= \mathbf{f}^H \mathbf{W}^H \mathbf{W} \mathbf{g} \\
&= [- \cdots -] \begin{bmatrix} \square & & \\ & \ddots & \\ & & \square \end{bmatrix} \begin{bmatrix} | \\ \vdots \\ | \end{bmatrix}
\end{aligned}
$$

$$
\mathbf{f} = \begin{bmatrix} f(z_1) \\ \vdots \\ f(z_N) \end{bmatrix}, \mathbf{W}^H \mathbf{W} = \begin{bmatrix} W_1^H W_1 & & \\ & \ddots & \\ & & W_N^H W_N \end{bmatrix}, \mathbf{g} = \begin{bmatrix} g(z_1) \\ \vdots \\ g(z_N) \end{bmatrix}
$$

# Note 1

## Matrix-vector notation

$$
\begin{aligned}
\langle f, g \rangle_W &= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\mathsf{I}-\mathsf{I}) \\
&= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\square\mathsf{I}) \\
&= \mathbf{f}^H \mathbf{W}^H \mathbf{W} \mathbf{g}
\end{aligned}
$$

$$
= \begin{bmatrix} - & \cdots & - \end{bmatrix} \begin{bmatrix} \square & & \\ & \ddots & \\ & & \square \end{bmatrix} \begin{bmatrix} | \\ \vdots \\ | \end{bmatrix}
$$

$$
\mathbf{f} = \begin{bmatrix} f(z_1) \\ \vdots \\ f(z_N) \end{bmatrix}, \mathbf{W}^H \mathbf{W} = \begin{bmatrix} W_1^H W_1 & & \\ & \ddots & \\ & & W_N^H W_N \end{bmatrix}, \mathbf{g} = \begin{bmatrix} g(z_1) \\ \vdots \\ g(z_N) \end{bmatrix}
$$

# Note 1

## Matrix-vector notation

$$
\begin{aligned}
\langle f, g \rangle_W &= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\mathsf{I}-\mathsf{I}) \\
&= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\square\mathsf{I}) \\
&= \mathbf{f}^H \mathbf{W}^H \mathbf{W} \mathbf{g} \\
&= [-\cdots-] \begin{bmatrix} \square & & \\ & \ddots & \\ & & \square \end{bmatrix} \begin{bmatrix} \mathsf{I} \\ \vdots \\ \mathsf{I} \end{bmatrix}
\end{aligned}
$$

$$
\mathbf{f} = \begin{bmatrix} f(z_1) \\ \vdots \\ f(z_N) \end{bmatrix}, \mathbf{W}^H \mathbf{W} = \begin{bmatrix} W_1^H W_1 & & \\ & \ddots & \\ & & W_N^H W_N \end{bmatrix}, \mathbf{g} = \begin{bmatrix} g(z_1) \\ \vdots \\ g(z_N) \end{bmatrix}
$$

# Note 1

## Matrix-vector notation

$$
\begin{aligned}
\langle f, g \rangle_W &= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(\text{—}\,\text{|}\,\text{—}\,\text{|}) \\
&= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(\text{—}\,\square\,\text{|}) \\
&= \mathbf{f}^H \mathbf{W}^H \mathbf{W} \mathbf{g} \\
&= [\text{—} \cdots \text{—}] \begin{bmatrix} \square & & \\ & \ddots & \\ & & \square \end{bmatrix} \begin{bmatrix} \text{|} \\ \vdots \\ \text{|} \end{bmatrix}
\end{aligned}
$$

$$
\mathbf{f} = \begin{bmatrix} f(z_1) \\ \vdots \\ f(z_N) \end{bmatrix}, \mathbf{W}^H \mathbf{W} = \begin{bmatrix} W_1^H W_1 & & \\ & \ddots & \\ & & W_N^H W_N \end{bmatrix}, \mathbf{g} = \begin{bmatrix} g(z_1) \\ \vdots \\ g(z_N) \end{bmatrix}
$$

# Note 2

## Vector polynomials

$$P(z) = \begin{bmatrix} A(z) \\ B(z) \end{bmatrix} = \left[\begin{array}{cccc|cccc} 1 & z^{-1} & \cdots & z^{-n} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 1 & z^{-1} & \cdots & z^{-n} \end{array}\right] \mathbf{p}$$

$$\mathbf{p} = \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \quad \mathbf{a} = [a_0, \ldots, a_n]^T, \quad \mathbf{b} = [b_0, \ldots, b_n]^T$$

$$P(z) = \begin{bmatrix} A(z) \\ B(z) \end{bmatrix} = \left[\begin{array}{cc|cc|c|cc} 1 & 0 & z^{-1} & 0 & \cdots & z^{-n} & 0 \\ 0 & 1 & 0 & z^{-1} & \cdots & 0 & z^{-n} \end{array}\right] \tilde{\mathbf{p}}$$

$$\tilde{\mathbf{p}} = \left[ \tilde{p}_0^T, \ldots, \tilde{p}_n^T \right]^T, \quad \tilde{p}_k = [a_k, b_k]^T$$

# Note 2

## Vector polynomials

$$P(z) = \begin{bmatrix} A(z) \\ B(z) \end{bmatrix} = \left[\begin{array}{cccc|cccc} 1 & z^{-1} & \cdots & z^{-n} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 1 & z^{-1} & \cdots & z^{-n} \end{array}\right] \mathbf{p}$$

$$\mathbf{p} = \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \quad \mathbf{a} = [a_0, \ldots, a_n]^T, \quad \mathbf{b} = [b_0, \ldots, b_n]^T$$

$$P(z) = \begin{bmatrix} A(z) \\ B(z) \end{bmatrix} = \left[\begin{array}{cc|cc|c|cc} 1 & 0 & z^{-1} & 0 & \cdots & z^{-n} & 0 \\ 0 & 1 & 0 & z^{-1} & \cdots & 0 & z^{-n} \end{array}\right] \tilde{\mathbf{p}}$$

$$\tilde{\mathbf{p}} = \left[ \tilde{p}_0^T, \ldots, \tilde{p}_n^T \right]^T, \quad \tilde{p}_k = [a_k, b_k]^T$$

# Note 2

## Vector polynomials

$$P(z) = \begin{bmatrix} A(z) \\ B(z) \end{bmatrix} = \underbrace{\left[\, \phi_0(z) \,\middle|\, \phi_1(z) \,\middle|\, \cdots \,\middle|\, \phi_n(z) \,\right]}_{\mathbf{\Phi}(z)} \mathbf{p}$$

$\phi_k(z) \in \mathbb{P}_k^{2\times 2}$, col.space $\mathbf{\Phi}(z) = \mathbb{P}_n^{2\times 1}$

$$\mathbf{p} = \left[\, p_0^T, \ldots, p_n^T \,\right]^T, \quad p_k = [p_{1k}, p_{2k}]^T$$

$$P(z) = \begin{bmatrix} A(z) \\ B(z) \end{bmatrix} = \left[\begin{array}{cc|cc|c|cc} 1 & 0 & z^{-1} & 0 & \cdots & z^{-n} & 0 \\ 0 & 1 & 0 & z^{-1} & \cdots & 0 & z^{-n} \end{array}\right] \tilde{\mathbf{p}}$$

$$\tilde{\mathbf{p}} = \left[\, \tilde{p}_0^T, \ldots, \tilde{p}_n^T \,\right]^T, \quad \tilde{p}_k = [a_k, b_k]^T$$

# Proof

Thm: arg min $\|P\|_W$ with $P$ monic is monic OVP

$$\begin{aligned}\langle f, g\rangle_W &= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(- \square \mathsf{I})\\ &= \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(-\mathsf{I} - \mathsf{I})\end{aligned}$$

$\dim \mathbb{P}_n^{2\times 1} = 2(n+1)$,

arrange OVP in $2\times 2$ blocks $\phi_i$

block-orthogonality: $\langle\cdot,\cdot\rangle_w = \sum(\square\ \square\ \square)$

$P(z) = \sum_{i=0}^{n} \phi_i(z) p_i, = \sum(\square\ \mathsf{I})$ $\quad p_n \neq 0$

$\min \|P\|_W^2 = \min \mathbf{p}^H \Phi^H \mathbf{W}^H \mathbf{W} \Phi \mathbf{p} = \min \mathbf{p}^H \mathbf{p}$

because $\Phi^H \mathbf{W}^H \mathbf{W} \Phi = I$ by orthogonality.

$\min \sum p_k^H p_k$ and $p_n \neq 0 \Rightarrow p_0 = \cdots p_{n-1} = 0$. QED

# Proof

Thm: arg min $\|P\|_W$ with $P$ monic is monic OVP

$$\langle f, g \rangle_W = \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(- \square \mathsf{I})$$

$\dim \mathbb{P}_n^{2\times 1} = 2(n+1)$,

arrange OVP in $2\times 2$ blocks $\phi_i$

block-orthogonality: $\langle \cdot, \cdot \rangle_w = \sum(\square \; \square \; \square)$

$$\left[\begin{array}{cccc|cccc} 1 & z^{-1} & \cdots & z^{-n} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 1 & z^{-1} & \cdots & z^{-n} \end{array}\right]$$

$P(z) = \sum_{i=0}^{n} \phi_i(z) p_i, = \sum(\square \; \mathsf{I}) \qquad p_n \neq 0$

$\min \|P\|_W^2 = \min \mathbf{p}^H \Phi^H \mathbf{W}^H \mathbf{W} \Phi \mathbf{p} = \min \mathbf{p}^H \mathbf{p}$

because $\Phi^H \mathbf{W}^H \mathbf{W} \Phi = I$ by orthogonality.

$\min \sum p_k^H p_k$ and $p_n \neq 0 \Rightarrow p_0 = \cdots p_{n-1} = 0$. QED

# Proof

Thm: arg min $\|P\|_W$ with $P$ monic is monic OVP

$$\langle f, g \rangle_W = \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(- \square \mathsf{I})$$

$\dim \mathbb{P}_n^{2\times 1} = 2(n+1)$,

arrange OVP in 2×2 blocks $\phi_i$

block-orthogonality: $\langle \cdot, \cdot \rangle_w = \sum(\square \; \square \; \square)$

$$\left[\begin{array}{cc|cc|c|cc} 1 & 0 & z^{-1} & 0 & \cdots & z^{-n} & 0 \\ 0 & 1 & 0 & z^{-1} & \cdots & 0 & z^{-n} \end{array}\right]$$

$P(z) = \sum_{i=0}^{n} \phi_i(z) p_i, = \sum(\square \; \mathsf{I})$ $\quad p_n \neq 0$

$\min \|P\|_W^2 = \min \mathbf{p}^H \Phi^H \mathbf{W}^H \mathbf{W} \Phi \mathbf{p} = \min \mathbf{p}^H \mathbf{p}$

because $\Phi^H \mathbf{W}^H \mathbf{W} \Phi = I$ by orthogonality.

$\min \sum p_k^H p_k$ and $p_n \neq 0 \Rightarrow p_0 = \cdots p_{n-1} = 0$. QED

# Proof

Thm: arg min $\|P\|_W$ with $P$ monic is monic OVP

$$\langle f, g \rangle_W = \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(- \square \mathsf{I})$$

$\dim \mathbb{P}_n^{2\times 1} = 2(n+1)$,

arrange OVP in 2×2 blocks $\phi_i$

block-orthogonality: $\langle \cdot, \cdot \rangle_w = \sum(\square \; \square \; \square)$

$P(z) = \sum_{i=0}^{n} \phi_i(z) p_i, = \sum(\square \; \mathsf{I})$ $\quad p_n \neq 0$

$\min \|P\|_W^2 = \min \mathbf{p}^H \Phi^H \mathbf{W}^H \mathbf{W} \Phi \mathbf{p} = \min \mathbf{p}^H \mathbf{p}$
because $\Phi^H \mathbf{W}^H \mathbf{W} \Phi = I$ by orthogonality.
$\min \sum p_k^H p_k$ and $p_n \neq 0 \Rightarrow p_0 = \cdots p_{n-1} = 0$. QED

How to construct the vector OP?

# Proof

Thm: arg min $\|P\|_W$ with $P$ monic is monic OVP

$$\langle f, g\rangle_W \quad = \quad \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(\blacksquare\square\blacksquare)$$

$\dim \mathbb{P}_n^{2\times 1} = 2(n+1)$,

arrange OVP in 2×2 blocks $\phi_i$

block-orthogonality: $\langle\cdot,\cdot\rangle_w = \sum(\square\ \square\ \square)$

$P(z) = \sum_{i=0}^{n} \phi_i(z) p_i, = \sum(\square\blacksquare) \qquad p_n \neq 0$

$\min \|P\|_W^2 = \min \mathbf{p}^H \Phi^H \mathbf{W}^H \mathbf{W} \Phi \mathbf{p} = \min \mathbf{p}^H \mathbf{p}$
because $\Phi^H \mathbf{W}^H \mathbf{W} \Phi = I$ by orthogonality.
$\min \sum p_k^H p_k$ and $p_n \neq 0 \Rightarrow p_0 = \cdots p_{n-1} = 0$. QED

How to construct the vector OP?

# Proof

Thm: arg min $\|P\|_W$ with $P$ monic is monic OVP

$$\langle f, g\rangle_W \quad = \quad \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(\blacksquare\,\square\,\blacksquare)$$

$\dim \mathbb{P}_n^{2\times 1} = 2(n+1)$,

arrange OVP in 2×2 blocks $\phi_i$

block-orthogonality: $\langle\cdot,\cdot\rangle_w = \sum(\square\ \square\ \square)$

$P(z) = \sum_{i=0}^{n} \phi_i(z) p_i, = \sum(\square\,\blacksquare)$ $\quad p_n \neq 0$

$\min \|P\|_W^2 = \min \mathbf{p}^H \Phi^H \mathbf{W}^H \mathbf{W} \Phi \mathbf{p} = \min \mathbf{p}^H \mathbf{p}$
because $\Phi^H \mathbf{W}^H \mathbf{W} \Phi = I$ by orthogonality.

$\min \sum p_k^H p_k$ and $p_n \neq 0 \Rightarrow p_0 = \cdots p_{n-1} = 0$. QED

How to construct the vector OP?

# Proof

Thm: arg min $\|P\|_W$ with $P$ monic is monic OVP

$$\langle f, g\rangle_W \quad = \quad \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(\blacksquare\,\square\,\blacksquare)$$

$\dim \mathbb{P}_n^{2\times 1} = 2(n+1)$,

arrange OVP in 2×2 blocks $\phi_i$

block-orthogonality: $\langle\cdot,\cdot\rangle_w = \sum(\square\ \square\ \square)$

$P(z) = \sum_{i=0}^{n} \phi_i(z) p_i, = \sum(\square\,\blacksquare)$ $\quad p_n \neq 0$

$\min \|P\|_W^2 = \min \mathbf{p}^H \Phi^H \mathbf{W}^H \mathbf{W} \Phi \mathbf{p} = \min \mathbf{p}^H \mathbf{p}$
because $\Phi^H \mathbf{W}^H \mathbf{W} \Phi = I$ by orthogonality.
$\min \sum p_k^H p_k$ and $p_n \neq 0 \Rightarrow p_0 = \cdots p_{n-1} = 0$. QED

How to construct the vector OP?

# Proof

Thm: arg min $\|P\|_W$ with $P$ monic is monic OVP

$$\langle f, g\rangle_W \quad = \quad \sum_k f(z_k)^H W_k^H W_k g(z_k) = \Sigma(\blacksquare\,\square\,\mathsf{I})$$

$\dim \mathbb{P}_n^{2\times 1} = 2(n+1)$,

arrange OVP in 2×2 blocks $\phi_i$

block-orthogonality: $\langle\cdot,\cdot\rangle_w = \sum(\square\,\square\,\square)$

$P(z) = \sum_{i=0}^{n} \phi_i(z) p_i, = \sum(\square\,\mathsf{I})$ $\quad p_n \neq 0$

$\min \|P\|_W^2 = \min \mathbf{p}^H \Phi^H \mathbf{W}^H \mathbf{W} \Phi \mathbf{p} = \min \mathbf{p}^H \mathbf{p}$
because $\Phi^H \mathbf{W}^H \mathbf{W} \Phi = I$ by orthogonality.
$\min \sum p_k^H p_k$ and $p_n \neq 0 \Rightarrow p_0 = \cdots p_{n-1} = 0$. QED

How to construct the vector OP?

# Recurrence

$$z^{-1}\phi_{k-1}(z) = \phi_k(z)\eta_{kk} + \cdots + \phi_0(z)\eta_{0k}, \quad k = 1, \ldots, n+1$$

$\Phi = [\phi_0 \ldots \phi_n]$, $\mathbf{H} =$ "white" block Hessenberg

# Recurrence

$$z^{-1}\phi_{k-1}(z) = \phi_k(z)\eta_{kk} + \cdots + \phi_0(z)\eta_{0k}, \quad k = 1, \ldots, n+1$$

$$z^{-1}[\qquad\qquad \phi_n] = [\phi_0 \cdots \phi_n \phi_{n+1}] \begin{bmatrix} & \eta_{0,n+1} \\ & \eta_{1,n+1} \\ & \vdots \\ & \eta_{n,n+1} \\ & \eta_{n+1,n+1} \end{bmatrix}$$

$\Phi = [\phi_0 \ldots \phi_n]$, $\mathbf{H} =$ "white" block Hessenberg

# Recurrence

$$z^{-1}\phi_{k-1}(z) = \phi_k(z)\eta_{kk} + \cdots + \phi_0(z)\eta_{0k}, \quad k = 1, \ldots, n+1$$

$$z^{-1}[\quad \cdots \phi_{n-1}\ \phi_n] = [\phi_0 \cdots \phi_n \phi_{n+1}] \begin{bmatrix} & \cdots & \eta_{0n} & \eta_{0,n+1} \\ & \cdots & \eta_{1n} & \eta_{1,n+1} \\ & \ddots & \vdots & \vdots \\ & & \eta_{nn} & \eta_{n,n+1} \\ & & & \eta_{n+1,n+1} \end{bmatrix}$$

$\Phi = [\phi_0 \ldots \phi_n]$, $\mathbf{H} =$ "white" block Hessenberg

# Recurrence

$$z^{-1}\phi_{k-1}(z) = \phi_k(z)\eta_{kk} + \cdots + \phi_0(z)\eta_{0k}, \quad k = 1, \ldots, n+1$$

$$z^{-1}[\phi_0 \cdots \phi_{n-1}\ \phi_n] = [\phi_0 \cdots \phi_n \phi_{n+1}] \begin{bmatrix} \eta_{01} & \cdots & \eta_{0n} & \eta_{0,n+1} \\ \eta_{11} & \cdots & \eta_{1n} & \eta_{1,n+1} \\ & \ddots & \vdots & \vdots \\ & & \eta_{nn} & \eta_{n,n+1} \\ & & & \eta_{n+1,n+1} \end{bmatrix}$$

$\Phi = [\phi_0 \ldots \phi_n]$, $\mathbf{H} =$ "white" block Hessenberg

# Recurrence

$$z^{-1}\phi_{k-1}(z) = \phi_k(z)\eta_{kk} + \cdots + \phi_0(z)\eta_{0k}, \quad k = 1, \ldots, n+1$$

$$z^{-1}[\phi_0 \cdots \phi_{n-1}\ \phi_n] = [\phi_0 \cdots \phi_n \phi_{n+1}] \begin{bmatrix} \eta_{01} & \cdots & \eta_{0n} & \eta_{0,n+1} \\ \eta_{11} & \cdots & \eta_{1n} & \eta_{1,n+1} \\ & \ddots & \vdots & \vdots \\ & & \eta_{nn} & \eta_{n,n+1} \\ & & & \eta_{n+1,n+1} \end{bmatrix}$$

$\Phi = [\phi_0 \ldots \phi_n]$, $\mathbf{H} =$ "white" block Hessenberg

# Recurrence

$$z^{-1}\phi_{k-1}(z) = \phi_k(z)\eta_{kk} + \cdots + \phi_0(z)\eta_{0k}, \quad k = 1, \ldots, n+1$$

$$z^{-1}[\phi_0 \cdots \phi_{n-1}\ \phi_n] = [\phi_0 \cdots \phi_n \phi_{n+1}] \begin{bmatrix} \eta_{01} & \cdots & \eta_{0n} & \eta_{0,n+1} \\ \eta_{11} & \cdots & \eta_{1n} & \eta_{1,n+1} \\ & \ddots & \vdots & \vdots \\ & & \eta_{nn} & \eta_{n,n+1} \\ & & & \eta_{n+1,n+1} \end{bmatrix}$$

$\Phi = [\phi_0 \ldots \phi_n]$, $\mathbf{H} =$ "white" block Hessenberg

$$z^{-1}\Phi(z) = \Phi(z)\mathbf{H} + [0 \cdots 0 \phi_{n+1}\eta_{n+1,n+1}]$$

# Recurrence

$$z^{-1}\Phi(z) = \Phi(z)\mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(z)\eta_{n+1,n+1}]$$

$$\Phi = [\phi_0 \cdots \phi_n], \quad \mathbf{H} = \begin{bmatrix} \eta_{01} & \cdots & \eta_{0n} & \eta_{0,n+1} \\ \eta_{11} & \cdots & \eta_{1n} & \eta_{1,n+1} \\ & \ddots & \vdots & \vdots \\ & & \eta_{nn} & \eta_{n,n+1} \end{bmatrix}$$

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} \quad = \mathbf{\Phi} \quad \mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

Setting

$$\phi_k(\mathbf{z}) = \begin{bmatrix} \phi_k(z_1) \\ \vdots \\ \phi_k(z_N) \end{bmatrix}, \quad \mathbf{\Phi} = \Phi(\mathbf{z}) = [\phi_0(\mathbf{z}) \cdots \phi_n(\mathbf{z})]$$
$$\tilde{\mathbf{Z}} = \mathrm{diag}(\mathbf{z}I_2)$$

# Recurrence

$$z^{-1}\Phi(z) = \Phi(z)\mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(z)\eta_{n+1,n+1}]$$

Now write this down for $\mathbf{z} = [z_1, \ldots, z_N]^T$

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} \quad = \mathbf{\Phi} \quad \mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

Setting

$$\phi_k(\mathbf{z}) = \begin{bmatrix} \phi_k(z_1) \\ \vdots \\ \phi_k(z_N) \end{bmatrix}, \quad \mathbf{\Phi} = \Phi(\mathbf{z}) = [\phi_0(\mathbf{z}) \cdots \phi_n(\mathbf{z})], \quad \tilde{\mathbf{Z}} = \mathrm{diag}(\mathbf{z} I_2)$$

# Recurrence

$$z^{-1}\Phi(z) = \Phi(z)\mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(z)\eta_{n+1,n+1}]$$

Now write this down for $\mathbf{z} = [z_1, \ldots, z_N]^T$

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} \quad = \mathbf{\Phi} \quad \mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

Setting

$$\phi_k(\mathbf{z}) = \begin{bmatrix} \phi_k(z_1) \\ \vdots \\ \phi_k(z_N) \end{bmatrix}, \qquad \begin{aligned} \mathbf{\Phi} &= \Phi(\mathbf{z}) = [\phi_0(\mathbf{z}) \cdots \phi_n(\mathbf{z})] \\ \tilde{\mathbf{Z}} &= \mathrm{diag}(\mathbf{z} I_2) \end{aligned}$$

# Recurrence

$$z^{-1}\Phi(z) = \Phi(z)\mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(z)\eta_{n+1,n+1}]$$

Now write this down for $\mathbf{z} = [z_1, \ldots, z_N]^T$

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} \quad = \mathbf{\Phi} \quad \mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

Setting

$$\phi_k(\mathbf{z}) = \begin{bmatrix} \phi_k(z_1) \\ \vdots \\ \phi_k(z_N) \end{bmatrix}, \qquad \begin{aligned} \mathbf{\Phi} &= \Phi(\mathbf{z}) = [\phi_0(\mathbf{z}) \cdots \phi_n(\mathbf{z})] \\ \tilde{\mathbf{Z}} &= \mathrm{diag}(\mathbf{z} I_2) \end{aligned}$$

# Unitary similarity

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} \quad = \mathbf{\Phi} \quad \mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

# Unitary similarity

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} \quad = \mathbf{\Phi} \quad \mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\langle \phi_k, \phi_l \rangle_W = \delta_{kl} I_2 \Rightarrow \mathbf{\Phi}^H \mathbf{W}^H \mathbf{W} \mathbf{\Phi} = \mathbf{I}$$

$$\mathbf{W} = \mathrm{diag}(W_1 \ldots W_N)$$

# Unitary similarity

$$\tilde{\mathbf{Z}}^{-1}\boldsymbol{\Phi} = \boldsymbol{\Phi}\,\mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\langle \phi_k, \phi_l \rangle_W = \delta_{kl} I_2 \Rightarrow \boldsymbol{\Phi}^H \mathbf{W}^H \mathbf{W} \boldsymbol{\Phi} = \mathbf{I}$$

$$\mathbf{W} = \mathrm{diag}(W_1 \ldots W_N)$$

$$\mathbf{Q} = \mathbf{W}\boldsymbol{\Phi} \Rightarrow \mathbf{Q}^H \mathbf{Q} = \mathbf{I}$$

# Unitary similarity

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} = \mathbf{\Phi}\ \mathbf{H} + [0\ \cdots\ 0\ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\langle \phi_k, \phi_l \rangle_W = \delta_{kl} I_2 \Rightarrow \mathbf{\Phi}^H \mathbf{W}^H \mathbf{W} \mathbf{\Phi} = \mathbf{I}$$

$$\mathbf{W} = \mathrm{diag}(W_1 \ldots W_N)$$

$$\mathbf{Q} = \mathbf{W}\mathbf{\Phi} \Rightarrow \mathbf{Q}^H \mathbf{Q} = \mathbf{I}$$

Multiply first eq. from the left with $\mathbf{W}$, $\mathbf{W}\tilde{\mathbf{Z}}^{-1} = \mathbf{Z}^{-1}\mathbf{W}$

$$\mathrm{diag}(W_1, \ldots, W_N)\mathrm{diag}(z_1 I_2, \ldots, z_N I_2)^{-1} =$$

$$\mathrm{diag}(z_1, \ldots, z_N)^{-1}\mathrm{diag}(W_1, \ldots, W_N).$$

$$W_k \cdot z_k^{-1} \mathrm{I}_2 = z_k^{-1} \cdot W_k$$

$$\mathbf{—} \cdot \square = \circ \cdot \mathbf{—}$$

# Unitary similarity

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} = \mathbf{\Phi}\,\mathbf{H} + [0 \ \cdots \ 0 \ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\langle \phi_k, \phi_l \rangle_W = \delta_{kl} I_2 \Rightarrow \mathbf{\Phi}^H \mathbf{W}^H \mathbf{W} \mathbf{\Phi} = \mathbf{I}$$

$$\mathbf{W} = \mathrm{diag}(W_1 \ldots W_N)$$

$$\mathbf{Q} = \mathbf{W}\mathbf{\Phi} \Rightarrow \mathbf{Q}^H \mathbf{Q} = \mathbf{I}$$

$$\mathbf{Z}^{-1}\mathbf{W}\mathbf{\Phi} = \mathbf{W}\mathbf{\Phi}\mathbf{H} + [0 \ \cdots \ 0 \ \mathbf{W}\phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

Replace $\mathbf{W}\mathbf{\Phi}$ by $\mathbf{Q}$

# Unitary similarity

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} = \mathbf{\Phi}\ \mathbf{H} + [0\ \cdots\ 0\ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\langle \phi_k, \phi_l \rangle_W = \delta_{kl} I_2 \Rightarrow \mathbf{\Phi}^H \mathbf{W}^H \mathbf{W} \mathbf{\Phi} = \mathbf{I}$$

$$\mathbf{W} = \mathrm{diag}(W_1 \ldots W_N)$$

$$\mathbf{Q} = \mathbf{W}\mathbf{\Phi} \Rightarrow \mathbf{Q}^H \mathbf{Q} = \mathbf{I}$$

$$\mathbf{Z}^{-1}\ \mathbf{Q} = \mathbf{Q}\mathbf{H} + [0\ \cdots\ 0\ \mathbf{W}\phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

Multiply from the left with $\mathbf{Q}^H$

# Unitary similarity

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} = \mathbf{\Phi}\ \mathbf{H} + [0\ \cdots\ 0\ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\langle \phi_k, \phi_l \rangle_W = \delta_{kl} I_2 \Rightarrow \mathbf{\Phi}^H \mathbf{W}^H \mathbf{W} \mathbf{\Phi} = \mathbf{I}$$

$$\mathbf{W} = \mathrm{diag}(W_1 \ldots W_N)$$

$$\mathbf{Q} = \mathbf{W}\mathbf{\Phi} \Rightarrow \mathbf{Q}^H \mathbf{Q} = \mathbf{I}$$

$$\mathbf{Z}^{-1}\ \mathbf{Q} = \mathbf{Q}\mathbf{H} + [0\ \cdots\ 0\ \mathbf{W}\phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\mathbf{Q}^H \mathbf{Z}^{-1} \mathbf{Q} = \mathbf{Q}^H \mathbf{Q}\mathbf{H} + [0 \cdots 0\ \mathbf{Q}^H \mathbf{W}\phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

use orthogonality!

# Unitary similarity

$$\tilde{\mathbf{Z}}^{-1}\mathbf{\Phi} = \mathbf{\Phi}\ \mathbf{H} + [0\ \cdots\ 0\ \phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\langle \phi_k, \phi_l \rangle_W = \delta_{kl} I_2 \Rightarrow \mathbf{\Phi}^H \mathbf{W}^H \mathbf{W} \mathbf{\Phi} = \mathbf{I}$$

$$\mathbf{W} = \mathrm{diag}(W_1 \ldots W_N)$$

$$\mathbf{Q} = \mathbf{W}\mathbf{\Phi} \Rightarrow \mathbf{Q}^H \mathbf{Q} = \mathbf{I}$$

$$\mathbf{Z}^{-1}\ \mathbf{Q} = \mathbf{Q}\mathbf{H} + [0\ \cdots\ 0\ \mathbf{W}\phi_{n+1}(\mathbf{z})\eta_{n+1,n+1}]$$

$$\mathbf{Q}^H \mathbf{Z}^{-1} \mathbf{Q} = \mathbf{Q}^H \mathbf{Q} \mathbf{H} + [0 \cdots 0\ \mathbf{Q}^H \mathbf{W} \phi_{n+1}(\mathbf{z}) \eta_{n+1,n+1}]$$

$$\mathbf{Q}^H \mathbf{Z}^{-1} \mathbf{Q} = \mathbf{H}$$

# Inverse eigenvalue problem

$$\mathbf{Q}^H \mathbf{Z}^{-1} \mathbf{Q} = \mathbf{H} \Leftrightarrow \mathbf{Z}^{-1} = \mathbf{Q}\mathbf{H}\mathbf{Q}^H$$

Note symmetry of $\mathbf{H}$

Inverse eigenvalue problem is not completely defined

# Inverse eigenvalue problem

$$\mathbf{Q}^H \mathbf{Z}^{-1} \mathbf{Q} = \mathbf{H} \Leftrightarrow \mathbf{Z}^{-1} = \mathbf{Q}\mathbf{H}\mathbf{Q}^H$$

$$\mathbf{Q}^H \begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \mathbf{\Phi}^H \mathbf{W}^H \mathbf{W} \begin{bmatrix} I_2 \\ \vdots \\ I_2 \end{bmatrix}$$

$\phi_0$ is a constant, hence $\phi_0 \phi_0^{-1} = I_2$ and thus

$$\phi_0(\mathbf{z})\phi_0^{-1} = \begin{bmatrix} I_2 \\ \vdots \\ I_2 \end{bmatrix}$$

# Inverse eigenvalue problem

$$\mathbf{Q}^H \mathbf{Z}^{-1} \mathbf{Q} = \mathbf{H} \Leftrightarrow \mathbf{Z}^{-1} = \mathbf{Q}\mathbf{H}\mathbf{Q}^H$$

$$\mathbf{Q}^H \begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \mathbf{\Phi}^H \mathbf{W}^H \mathbf{W} \phi_0(\mathbf{z}) \phi_0^{-1} = \begin{bmatrix} \phi_0^{-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix} =: \begin{bmatrix} \eta_{00} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\|\phi_0\|_W = I_2 \Rightarrow \eta_{00}^H \eta_{00} = \sum W_k^H W_k$$

# Inverse eigenvalue problem

$$\mathbf{Q}^H \mathbf{Z}^{-1} \mathbf{Q} = \mathbf{H} \Leftrightarrow \mathbf{Z}^{-1} = \mathbf{Q}\mathbf{H}\mathbf{Q}^H$$

$$\mathbf{Q}^H \begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \eta_{00} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

# Inverse eigenvalue problem

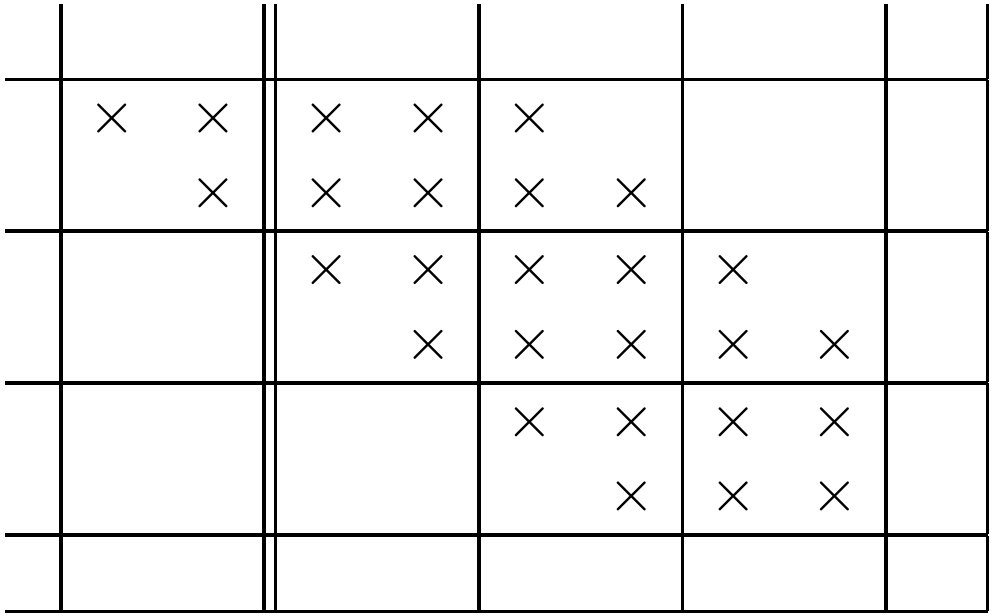
$$\mathbf{Q}^H \mathbf{Z}^{-1} \mathbf{Q} = \mathbf{H} \Leftrightarrow \mathbf{Z}^{-1} = \mathbf{Q}\mathbf{H}\mathbf{Q}^H$$

$$\mathbf{Q}^H \begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \eta_{00} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\mathbf{Q}^H \left[\begin{array}{c|c} W_1 & \\ \vdots & \mathbf{Z}^{-1} \\ W_N & \end{array}\right] \left[\begin{array}{c|c} I_2 & \\ \hline & \mathbf{Q} \end{array}\right] = \left[\begin{array}{c|c} \eta_{00} & \\ & \\ 0 & \mathbf{H} \end{array}\right]$$

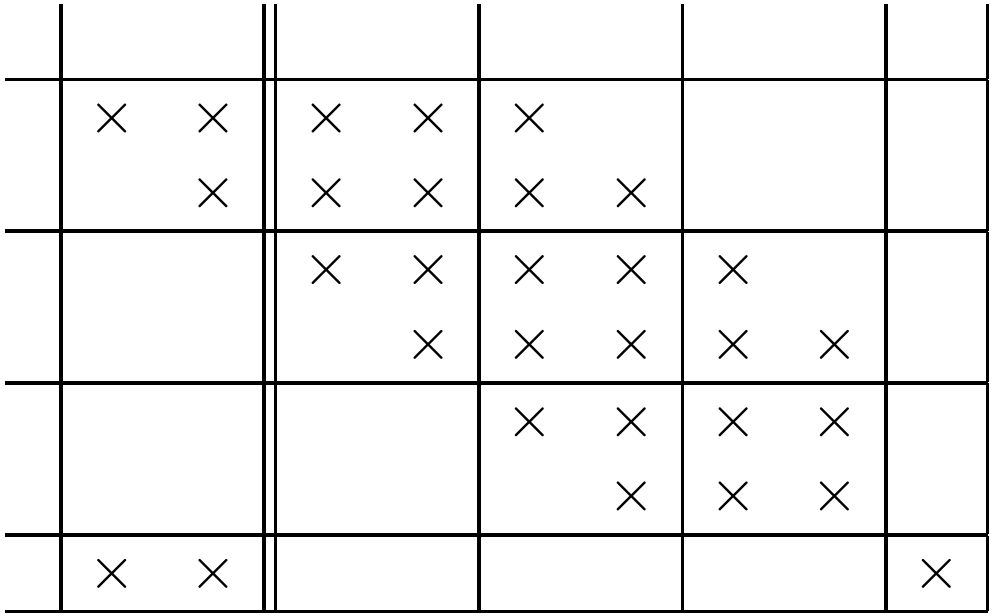
# Algorithm

Continuous time case



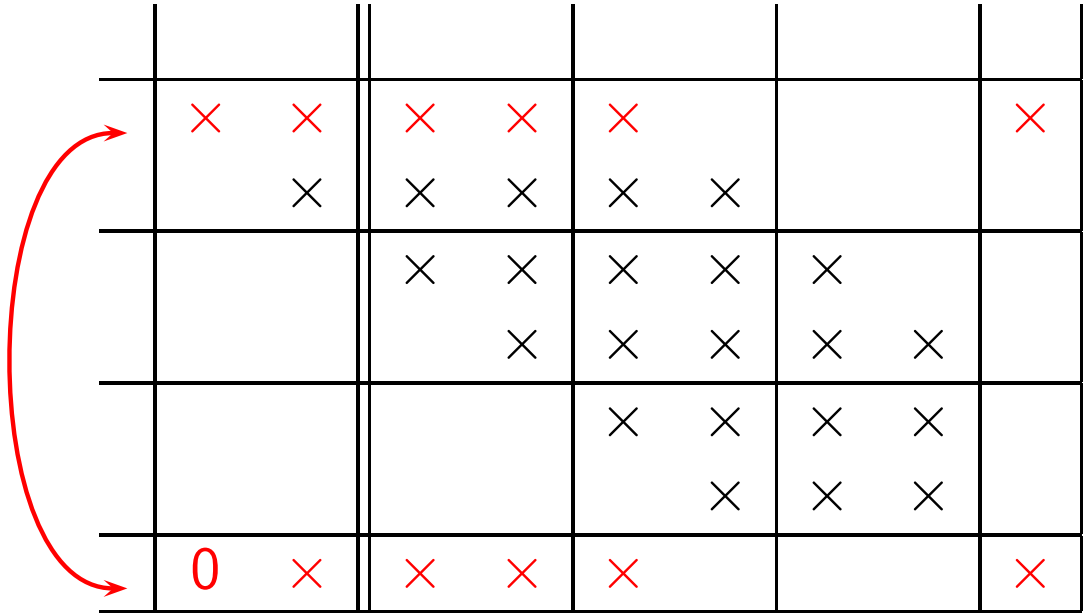
# Algorithm

Continuous time case



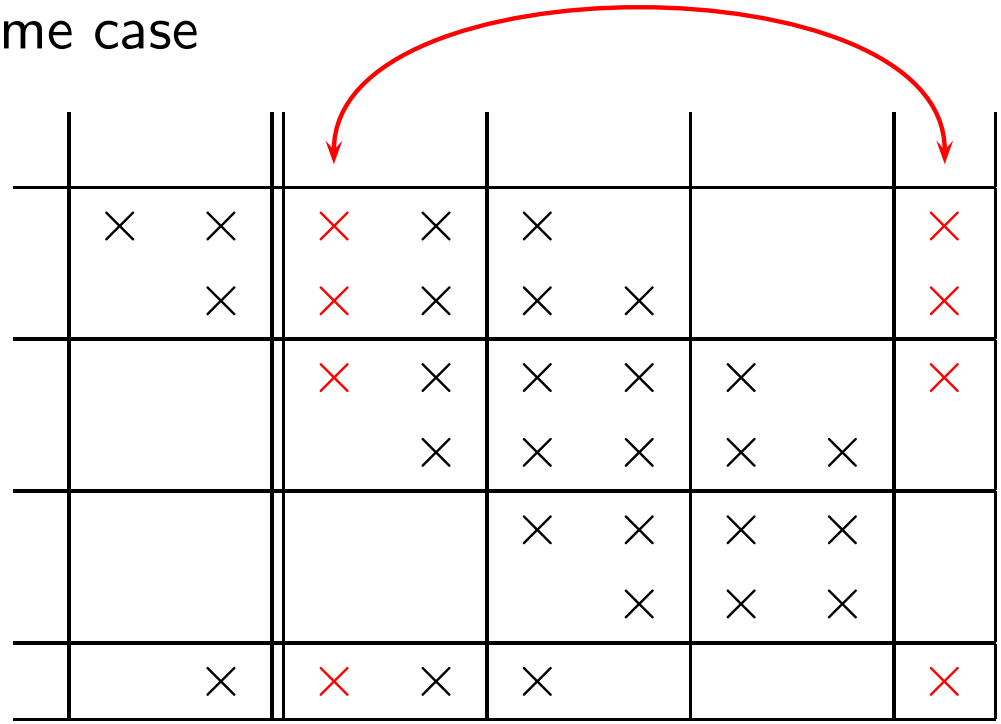
# Algorithm

Continuous time case



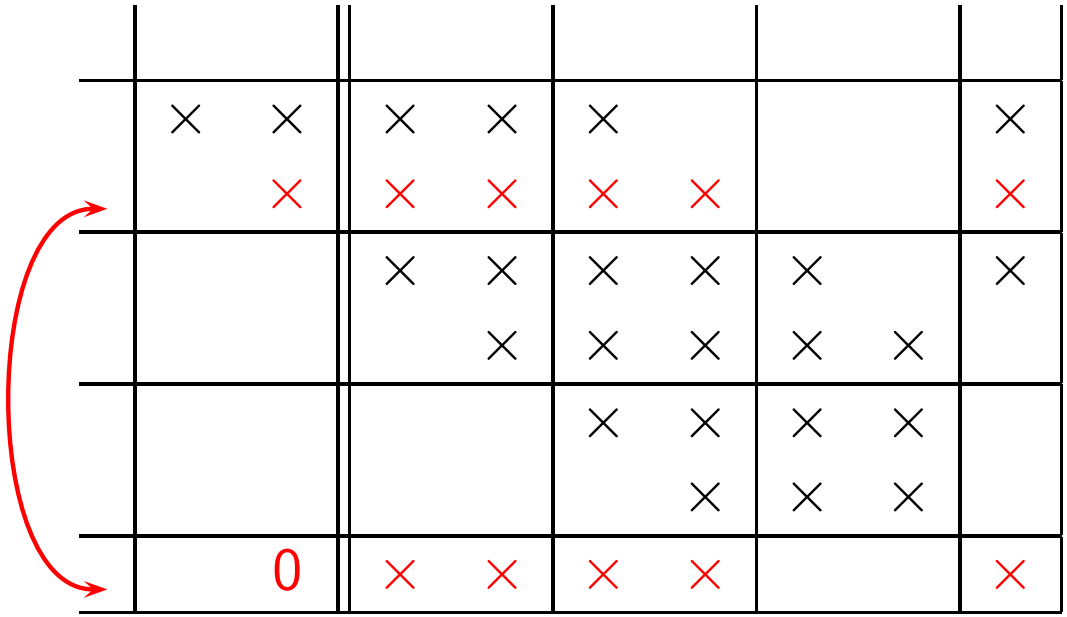
# Algorithm

Continuous time case



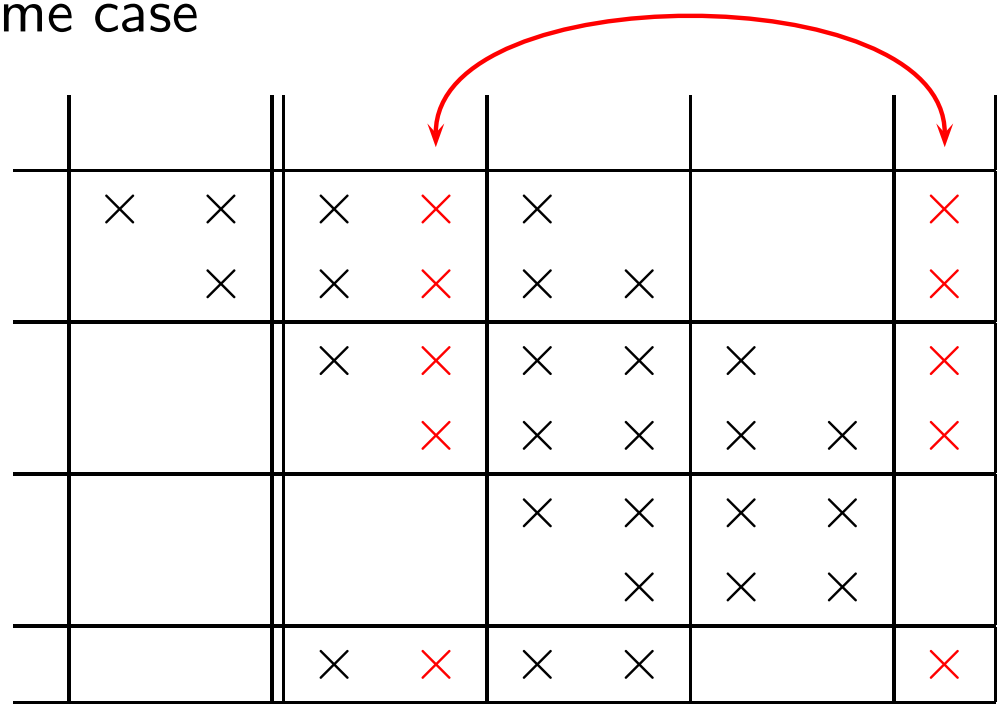
# Algorithm

Continuous time case



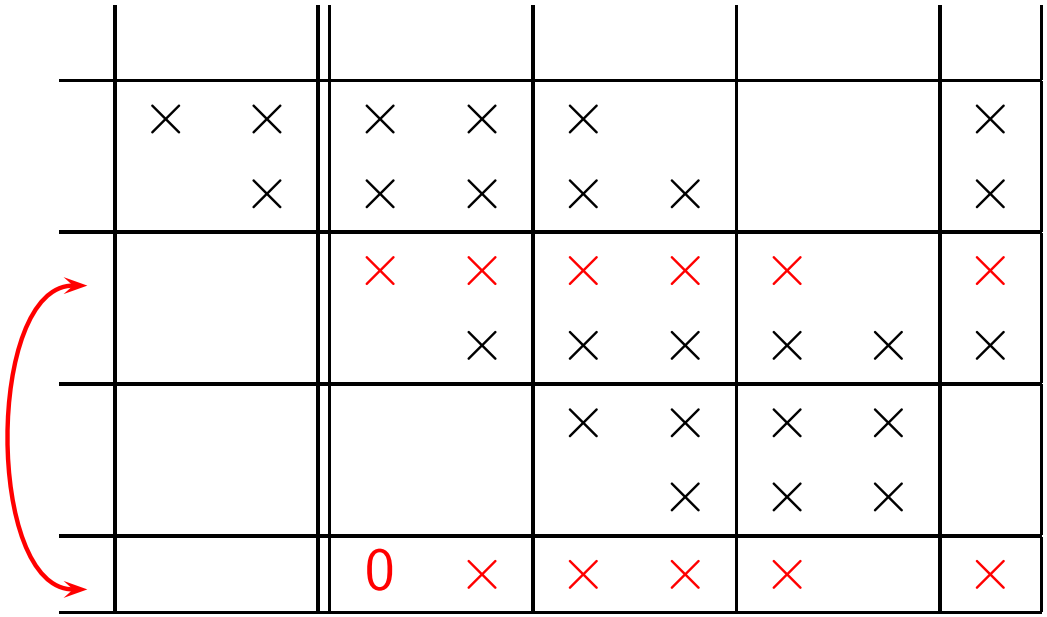
# Algorithm

Continuous time case



# Algorithm

Continuous time case



# Algorithm

Continuous time case




zero by symmetry

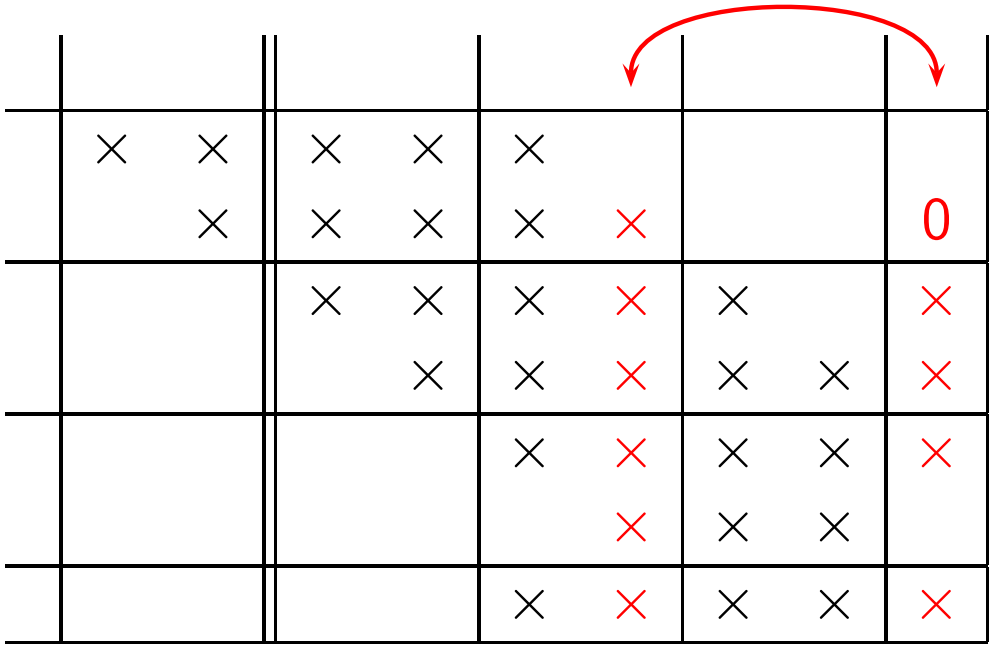
# Algorithm

Continuous time case



# Algorithm

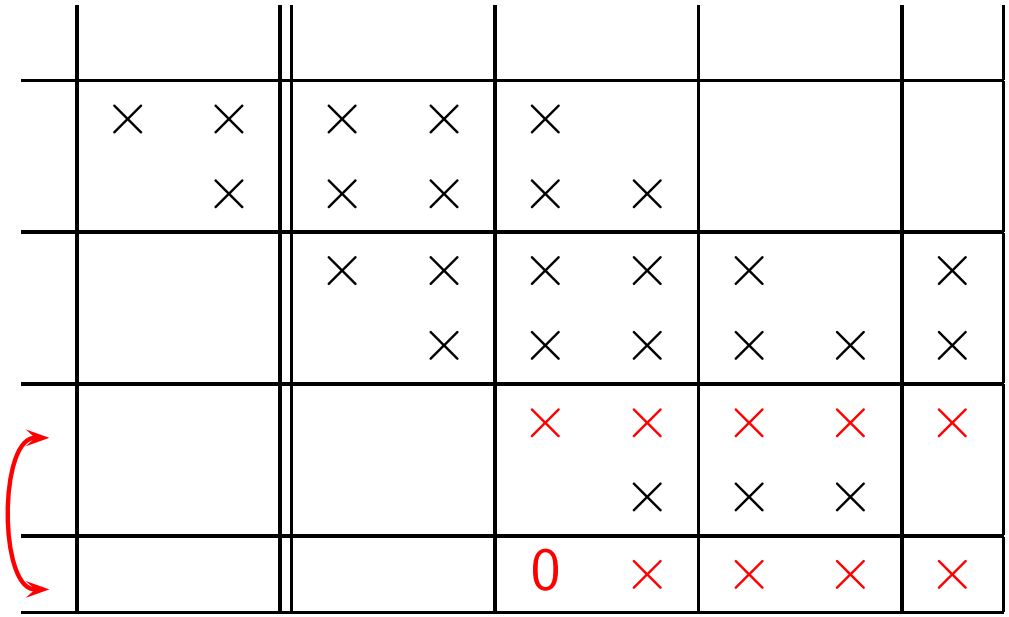
Continuous time case




zero by symmetry

# Algorithm

Continuous time case



# Algorithm

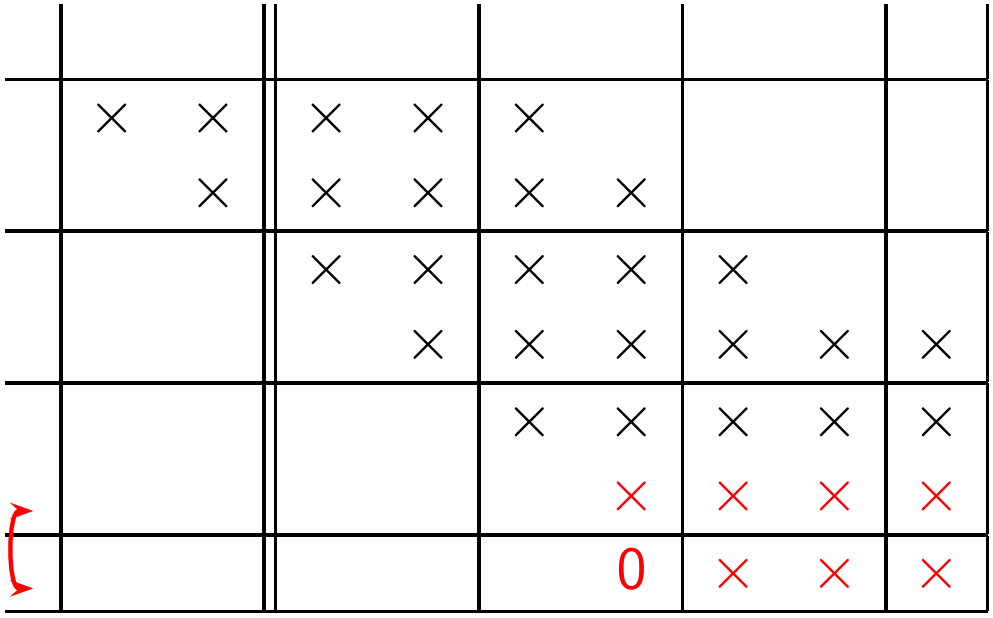
Continuous time case




zero by symmetry

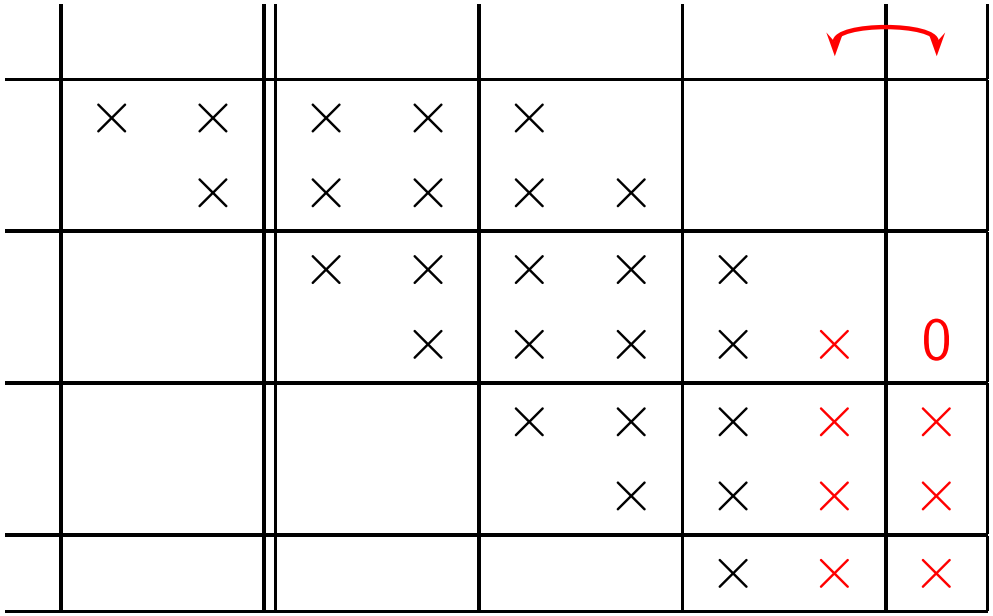
# Algorithm

Continuous time case



# Algorithm

Continuous time case




zero by symmetry

# Algorithm

Discrete time case:

- Szegő-type recurrence:
$$\begin{aligned} \phi_n \sigma_n &= z^{-1}\phi_{n-1} + \phi^*_{n-1}\gamma_n \\ \phi^*_n \sigma^H_n &= z^{-1}\phi_{n-1}\gamma^H_n + \phi^*_{n-1} \end{aligned}$$
$\sigma_n, \gamma_n \in \mathbb{C}^{2\times 2}$ (block Schur parameters)
- Store the block upper Hessenberg matrix in factored form $H = G_1 G_2 \cdots G_m$

$$G_k = \begin{bmatrix} I_{2(k-1)} & & & \\ & -\gamma_k & \hat{\sigma}_k & \\ & \sigma_k & \hat{\gamma}_k & \\ & & & I_{\ldots} \end{bmatrix}$$

- Chasing the elements to block upper Hessenberg form by similarity transformations percolating through the product requires operations on $3 \times 3$ or $5 \times 5$ blocks $\Rightarrow$ fast algorithm.

# Algorithm

Discrete time case:

- Szegő-type recurrence:
$$\begin{aligned} \phi_n \sigma_n &= z^{-1}\phi_{n-1} + \phi_{n-1}^* \gamma_n \\ \phi_n^* \sigma_n^H &= z^{-1}\phi_{n-1}\gamma_n^H + \phi_{n-1}^* \end{aligned}$$
$\sigma_n, \gamma_n \in \mathbb{C}^{2\times 2}$ (block Schur parameters)
- Store the block upper Hessenberg matrix in factored form $H = G_1 G_2 \cdots G_m$

$$G_k = \begin{bmatrix} I_{2(k-1)} & & & \\ & -\gamma_k & \hat{\sigma}_k & \\ & \sigma_k & \hat{\gamma}_k & \\ & & & I_{\ldots} \end{bmatrix}$$

- Chasing the elements to block upper Hessenberg form by similarity transformations percolating through the product requires operations on $3 \times 3$ or $5 \times 5$ blocks $\Rightarrow$ fast algorithm.

# Algorithm

Discrete time case:

- Szegő-type recurrence:
$$\begin{aligned} \phi_n \sigma_n &= z^{-1}\phi_{n-1} + \phi_{n-1}^* \gamma_n \\ \phi_n^* \sigma_n^H &= z^{-1}\phi_{n-1}\gamma_n^H + \phi_{n-1}^* \end{aligned}$$
$\sigma_n, \gamma_n \in \mathbb{C}^{2\times 2}$ (block Schur parameters)
- Store the block upper Hessenberg matrix in factored form $H = G_1 G_2 \cdots G_m$

$$G_k = \begin{bmatrix} I_{2(k-1)} & & & \\ & -\gamma_k & \hat{\sigma}_k & \\ & \sigma_k & \hat{\gamma}_k & \\ & & & I_{\ldots} \end{bmatrix}$$

- Chasing the elements to block upper Hessenberg form by similarity transformations percolating through the product requires operations on $3 \times 3$ or $5 \times 5$ blocks $\Rightarrow$ fast algorithm.

# Real computations

- We can preprocess complex data to do real arithmetic.
- 2 data points $e^{i\omega_k}, e^{-i\omega_k} \in \mathbb{T}$ with corresponding weights $W_k$ and $\overline{W}_k$.
- Define $Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix}$
- Then $Q^H \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \sqrt{2} \begin{bmatrix} \mathrm{Re}(W_k) \\ \mathrm{Im}(W_k) \end{bmatrix}$
- and $Q^H \begin{bmatrix} e^{i\omega_k} & 0 \\ 0 & e^{-i\omega_k} \end{bmatrix} Q = \begin{bmatrix} \cos\omega_k & \sin\omega_k \\ -\sin\omega_k & \cos\omega_k \end{bmatrix}$

# Real computations

- We can preprocess complex data to do real arithmetic.
- 2 data points $e^{i\omega_k}, e^{-i\omega_k} \in \mathbb{T}$ with corresponding weights $W_k$ and $\overline{W}_k$.
- Define $Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix}$
- Then $Q^H \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \sqrt{2} \begin{bmatrix} \mathrm{Re}(W_k) \\ \mathrm{Im}(W_k) \end{bmatrix}$
- and $Q^H \begin{bmatrix} e^{i\omega_k} & 0 \\ 0 & e^{-i\omega_k} \end{bmatrix} Q = \begin{bmatrix} \cos\omega_k & \sin\omega_k \\ -\sin\omega_k & \cos\omega_k \end{bmatrix}$

# Real computations

- We can preprocess complex data to do real arithmetic.
- 2 data points $e^{i\omega_k}, e^{-i\omega_k} \in \mathbb{T}$ with corresponding weights $W_k$ and $\overline{W}_k$.
- Define $Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix}$
- Then $Q^H \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \sqrt{2} \begin{bmatrix} \mathrm{Re}(W_k) \\ \mathrm{Im}(W_k) \end{bmatrix}$
- and $Q^H \begin{bmatrix} e^{i\omega_k} & 0 \\ 0 & e^{-i\omega_k} \end{bmatrix} Q = \begin{bmatrix} \cos\omega_k & \sin\omega_k \\ -\sin\omega_k & \cos\omega_k \end{bmatrix}$

# Real computations

- We can preprocess complex data to do real arithmetic.
- 2 data points $e^{i\omega_k}, e^{-i\omega_k} \in \mathbb{T}$ with corresponding weights $W_k$ and $\overline{W}_k$.
- Define $Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix}$
- Then $Q^H \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \sqrt{2} \begin{bmatrix} \mathrm{Re}(W_k) \\ \mathrm{Im}(W_k) \end{bmatrix}$
- and $Q^H \begin{bmatrix} e^{i\omega_k} & 0 \\ 0 & e^{-i\omega_k} \end{bmatrix} Q = \begin{bmatrix} \cos\omega_k & \sin\omega_k \\ -\sin\omega_k & \cos\omega_k \end{bmatrix}$

# Real computations

- We can preprocess complex data to do real arithmetic.
- 2 data points $e^{i\omega_k}, e^{-i\omega_k} \in \mathbb{T}$ with corresponding weights $W_k$ and $\overline{W}_k$.
- Define $Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix}$
- Then $Q^H \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} W_k \\ \overline{W}_k \end{bmatrix} = \sqrt{2} \begin{bmatrix} \mathrm{Re}(W_k) \\ \mathrm{Im}(W_k) \end{bmatrix}$
- and $Q^H \begin{bmatrix} e^{i\omega_k} & 0 \\ 0 & e^{-i\omega_k} \end{bmatrix} Q = \begin{bmatrix} \cos\omega_k & \sin\omega_k \\ -\sin\omega_k & \cos\omega_k \end{bmatrix}$

# MIMO system

- $\vec{M}=$ stack columns in long vector
- model $A^{-1}B$: $A$ is $n_y \times n_y$, $B$ is $n_y \times n_u$
- $\vec{P} = \begin{bmatrix} \vec{A} \\ \vec{B} \end{bmatrix}$ this is $n_y(n_y + n_u)$ long
- $E = AY - BU$ $(n_y \times 1) \Rightarrow \min \sum_k \vec{E}^H(z_k) C_{\vec{E}}^{-1}(z_k) \vec{E}(z_k)$
- $\vec{E} = ([Y^T \quad -U^T] \otimes I_{n_y})\vec{P}$

  $C_{\vec{E}} = \mathrm{Cov}(\vec{E}) = C_1 + C_2 - 2C_3$

  $$\begin{cases} C_1 = (I_{n_u} \otimes A) C_{\vec{Y}} (I_{n_u} \otimes A^H) \\ C_2 = (I_{n_u} \otimes B) C_{\vec{U}} (I_{n_u} \otimes B^H) \\ C_3 = \mathrm{Herm}[(I_{n_u} \otimes A) C_{\vec{Y}\vec{U}} (I_{n_u} \otimes B^H)] \end{cases}$$

- $\min \|\vec{P}\|_W$, $W = ([Y^T \quad -U^T] \otimes I_{n_y}) C_{\vec{E}}^{-1/2}$ has size $1 \times (n_y + n_u) n_y$

# MIMO system

- $\vec{M}$= stack columns in long vector
- model $A^{-1}B$: $A$ is $n_y \times n_y$, $B$ is $n_y \times n_u$
- $\vec{P} = \begin{bmatrix} \vec{A} \\ \vec{B} \end{bmatrix}$ this is $n_y(n_y + n_u)$ long
- $E = AY - BU$ $(n_y \times 1) \Rightarrow \min \sum_k \vec{E}^H(z_k) C_{\vec{E}}^{-1}(z_k) \vec{E}(z_k)$
- $\vec{E} = ([Y^T \quad -U^T] \otimes I_{n_y})\vec{P}$

  $C_{\vec{E}} = \mathrm{Cov}(\vec{E}) = C_1 + C_2 - 2C_3$

  $$\begin{cases} C_1 = (I_{n_u} \otimes A) C_{\vec{Y}} (I_{n_u} \otimes A^H) \\ C_2 = (I_{n_u} \otimes B) C_{\vec{U}} (I_{n_u} \otimes B^H) \\ C_3 = \mathrm{Herm}[(I_{n_u} \otimes A) C_{\vec{Y}\vec{U}} (I_{n_u} \otimes B^H)] \end{cases}$$

- $\min \|\vec{P}\|_W$, $W = ([Y^T \quad -U^T] \otimes I_{n_y}) C_{\vec{E}}^{-1/2}$ has size $1 \times (n_y + n_u) n_y$

# MIMO system

- $\vec{M}$= stack columns in long vector
- model $A^{-1}B$: $A$ is $n_y \times n_y$, $B$ is $n_y \times n_u$
- $\vec{P} = \begin{bmatrix} \vec{A} \\ \vec{B} \end{bmatrix}$ this is $n_y(n_y + n_u)$ long
- $E = AY - BU$ $(n_y \times 1) \Rightarrow \min \sum_k \vec{E}^H(z_k) C_{\vec{E}}^{-1}(z_k) \vec{E}(z_k)$
- $\vec{E} = ([Y^T \quad -U^T] \otimes I_{n_y})\vec{P}$

  $C_{\vec{E}} = \mathrm{Cov}(\vec{E}) = C_1 + C_2 - 2C_3$

  $$\begin{cases} C_1 = (I_{n_u} \otimes A) C_{\vec{Y}} (I_{n_u} \otimes A^H) \\ C_2 = (I_{n_u} \otimes B) C_{\vec{U}} (I_{n_u} \otimes B^H) \\ C_3 = \mathrm{Herm}[(I_{n_u} \otimes A) C_{\vec{Y}\vec{U}} (I_{n_u} \otimes B^H)] \end{cases}$$

- $\min \|\vec{P}\|_W$, $W = ([Y^T \quad -U^T] \otimes I_{n_y}) C_{\vec{E}}^{-1/2}$ has size $1 \times (n_y + n_u) n_y$

# MIMO system

- $\vec{M}$= stack columns in long vector
- model $A^{-1}B$: $A$ is $n_y \times n_y$, $B$ is $n_y \times n_u$
- $\vec{P} = \begin{bmatrix} \vec{A} \\ \vec{B} \end{bmatrix}$ this is $n_y(n_y + n_u)$ long
- $E = AY - BU$ $(n_y \times 1) \Rightarrow \min \sum_k \vec{E}^H(z_k) C_{\vec{E}}^{-1}(z_k) \vec{E}(z_k)$
- $\vec{E} = ([Y^T \quad -U^T] \otimes I_{n_y})\vec{P}$

  $C_{\vec{E}} = \mathrm{Cov}(\vec{E}) = C_1 + C_2 - 2C_3$

  $$\begin{cases} C_1 = (I_{n_u} \otimes A) C_{\vec{Y}} (I_{n_u} \otimes A^H) \\ C_2 = (I_{n_u} \otimes B) C_{\vec{U}} (I_{n_u} \otimes B^H) \\ C_3 = \mathrm{Herm}[(I_{n_u} \otimes A) C_{\vec{Y}\vec{U}} (I_{n_u} \otimes B^H)] \end{cases}$$

- $\min \|\vec{P}\|_W$, $W = ([Y^T \quad -U^T] \otimes I_{n_y}) C_{\vec{E}}^{-1/2}$ has size $1 \times (n_y + n_u) n_y$

# MIMO system

- $\vec{M}$= stack columns in long vector
- model $A^{-1}B$: $A$ is $n_y \times n_y$, $B$ is $n_y \times n_u$
- $\vec{P} = \begin{bmatrix} \vec{A} \\ \vec{B} \end{bmatrix}$ this is $n_y(n_y + n_u)$ long
- $E = AY - BU$ $(n_y \times 1) \Rightarrow \min \sum_k \vec{E}^H(z_k) C_{\vec{E}}^{-1}(z_k) \vec{E}(z_k)$
- $\vec{E} = ([Y^T \quad -U^T] \otimes I_{n_y})\vec{P}$

  $C_{\vec{E}} = \mathrm{Cov}(\vec{E}) = C_1 + C_2 - 2C_3$

  $$\begin{cases} C_1 = (I_{n_u} \otimes A) C_{\vec{Y}} (I_{n_u} \otimes A^H) \\ C_2 = (I_{n_u} \otimes B) C_{\vec{U}} (I_{n_u} \otimes B^H) \\ C_3 = \mathrm{Herm}[(I_{n_u} \otimes A) C_{\vec{Y}\vec{U}} (I_{n_u} \otimes B^H)] \end{cases}$$

- $\min \|\vec{P}\|_W$, $W = ([Y^T \quad -U^T] \otimes I_{n_y}) C_{\vec{E}}^{-1/2}$ has size $1 \times (n_y + n_u) n_y$

# MIMO system

- $\vec{M}$= stack columns in long vector
- model $A^{-1}B$: $A$ is $n_y \times n_y$, $B$ is $n_y \times n_u$
- $\vec{P} = \begin{bmatrix} \vec{A} \\ \vec{B} \end{bmatrix}$ this is $n_y(n_y + n_u)$ long
- $E = AY - BU$ $(n_y \times 1) \Rightarrow \min \sum_k \vec{E}^H(z_k) C_{\vec{E}}^{-1}(z_k) \vec{E}(z_k)$
- $\vec{E} = ([Y^T \quad -U^T] \otimes I_{n_y})\vec{P}$
  $C_{\vec{E}} = \mathrm{Cov}(\vec{E}) = C_1 + C_2 - 2C_3$
  $$\begin{cases} C_1 = (I_{n_u} \otimes A) C_{\vec{Y}} (I_{n_u} \otimes A^H) \\ C_2 = (I_{n_u} \otimes B) C_{\vec{U}} (I_{n_u} \otimes B^H) \\ C_3 = \mathrm{Herm}[(I_{n_u} \otimes A) C_{\vec{Y}\vec{U}} (I_{n_u} \otimes B^H)] \end{cases}$$
- $\min \|\vec{P}\|_W$, $W = ([Y^T \quad -U^T] \otimes I_{n_y}) C_{\vec{E}}^{-1/2}$ has size $1 \times (n_y + n_u) n_y$

# Generalizations

- Similar for discrete time systems ($z_i \in \mathbb{T}$)
- Fast algorithm ($O(N^2)$ operations)
- Economize if $n < N$ (operate on $N \times n$ matrix)
- Works for MIMO systems (OVP are $s \times 1$)
- Several degree structures are possible (column permutations)

# Generalizations

- Similar for discrete time systems ($z_i \in \mathbb{T}$)
- Fast algorithm ($O(N^2)$ operations)
- Economize if $n < N$ (operate on $N \times n$ matrix)
- Works for MIMO systems (OVP are $s \times 1$)
- Several degree structures are possible (column permutations)

# Generalizations

- Similar for discrete time systems ($z_i \in \mathbb{T}$)
- Fast algorithm ($O(N^2)$ operations)
- Economize if $n < N$ (operate on $N \times n$ matrix)
- Works for MIMO systems (OVP are $s \times 1$)
- Several degree structures are possible (column permutations)

# Generalizations

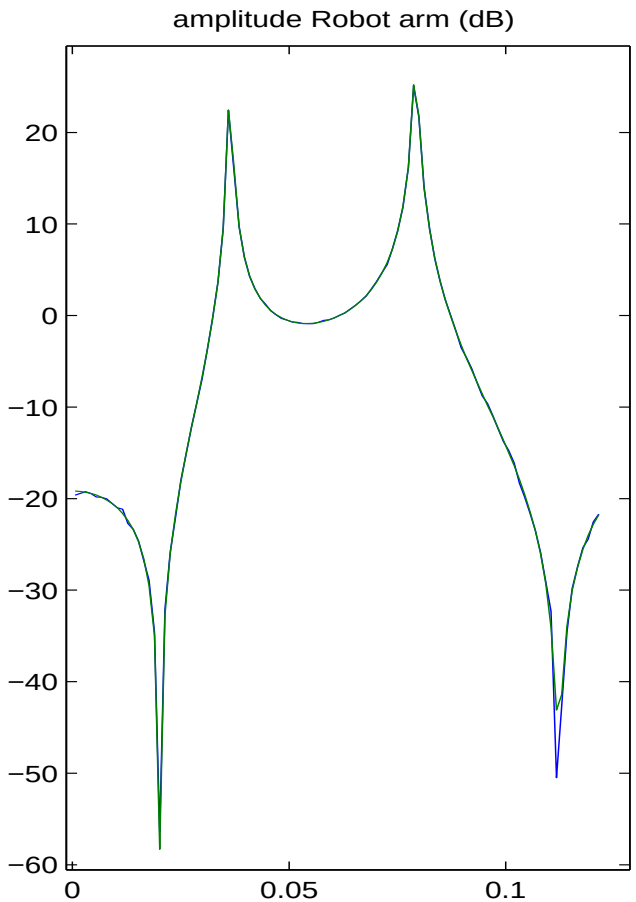
- Similar for discrete time systems ($z_i \in \mathbb{T}$)
- Fast algorithm ($O(N^2)$ operations)
- Economize if $n < N$ (operate on $N \times n$ matrix)
- Works for MIMO systems (OVP are $s \times 1$)
- Several degree structures are possible (column permutations)

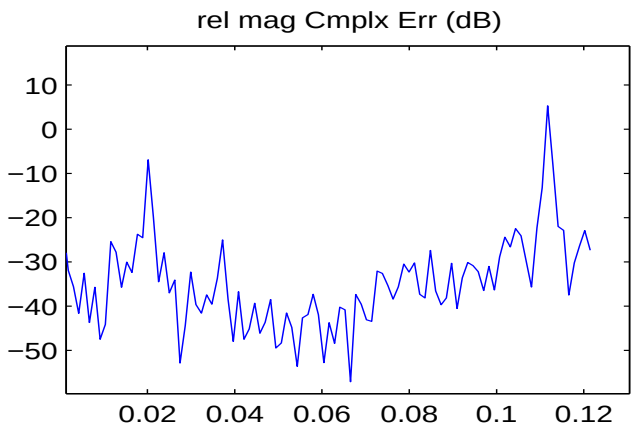
# Generalizations

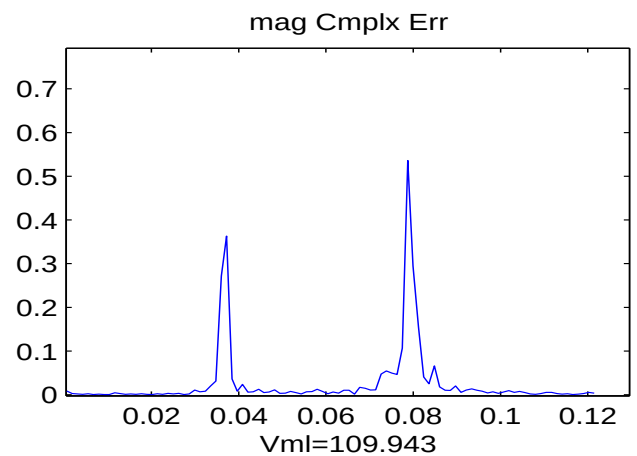
- Similar for discrete time systems ($z_i \in \mathbb{T}$)
- Fast algorithm ($O(N^2)$ operations)
- Economize if $n < N$ (operate on $N \times n$ matrix)
- Works for MIMO systems (OVP are $s \times 1$)
- Several degree structures are possible (column permutations)

# Robotarm

order [6/6], 100 data points, 200Hz, discrete time

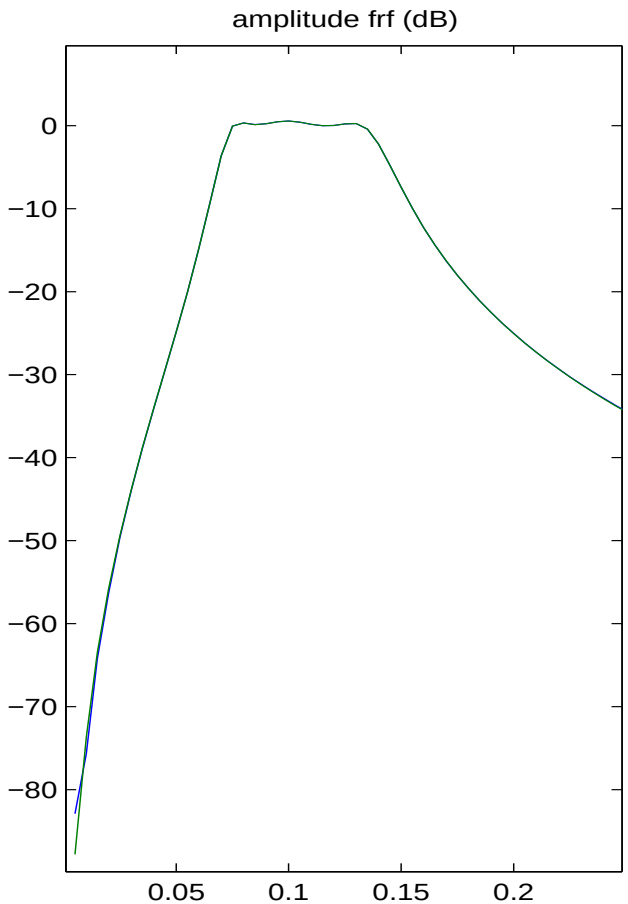


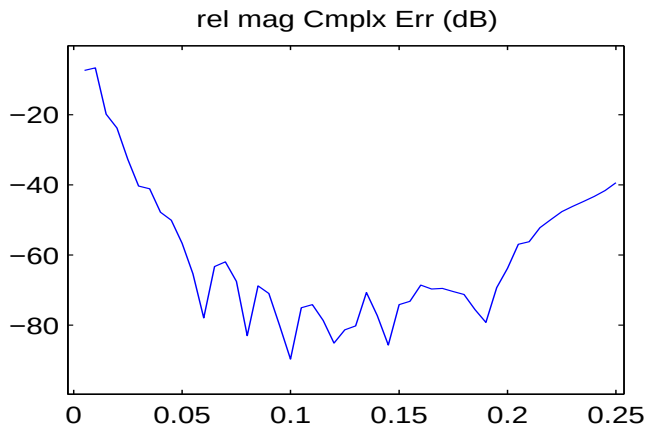



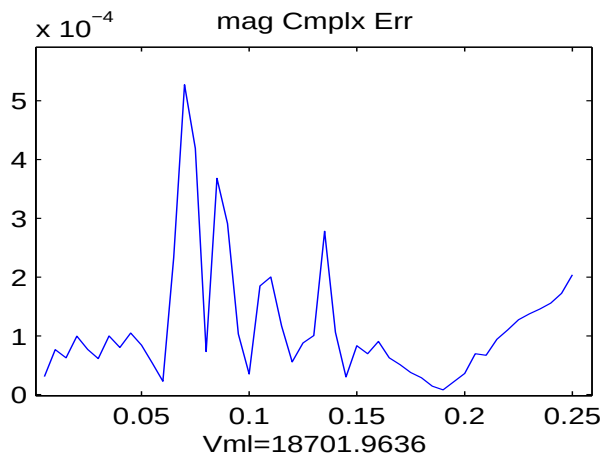



# Bandpass filter

order [6/6], 50 data points, 20KHz, discrete time

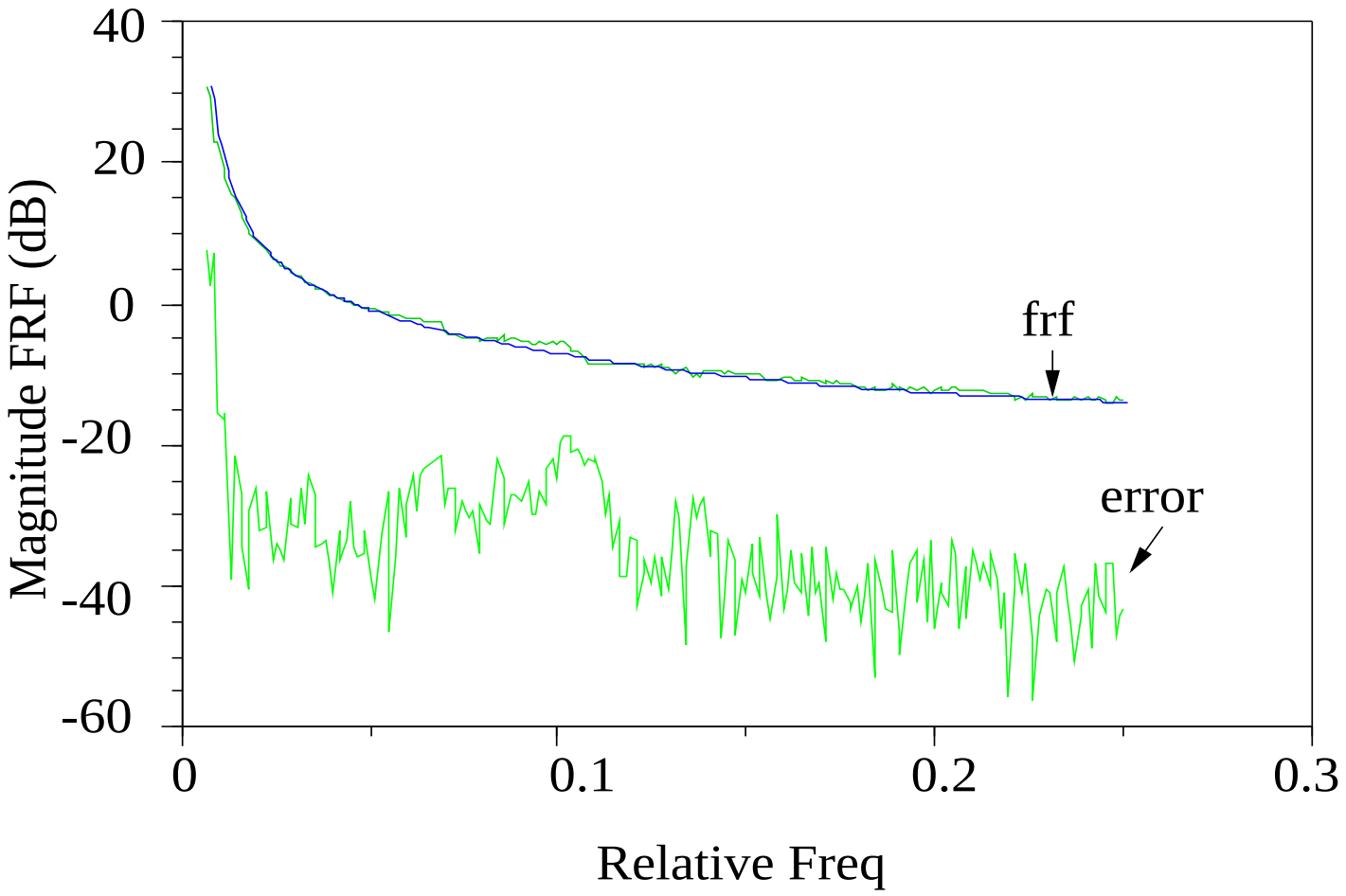


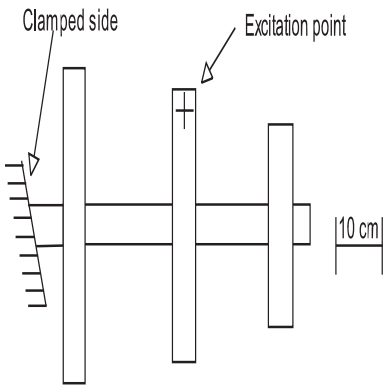
# CD's radial servo system

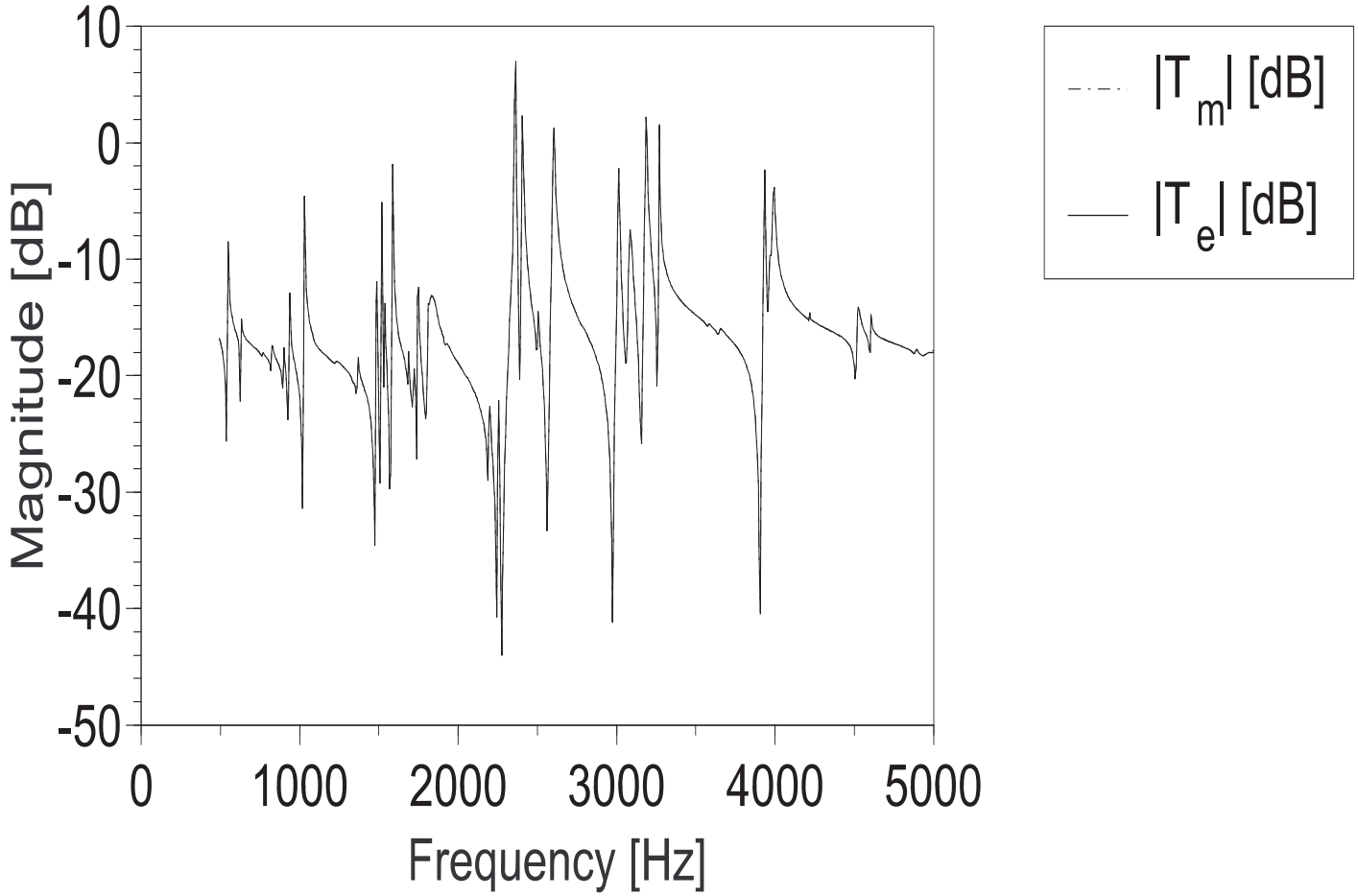
sampling frequency 9.7 kHz, order [5/5]



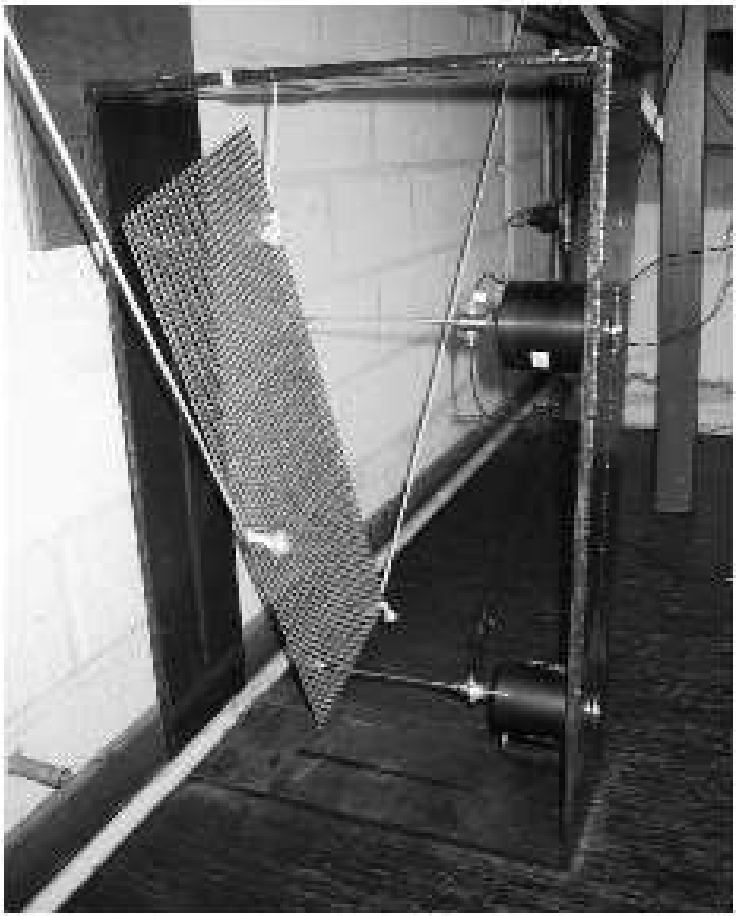
# Mechanical structure

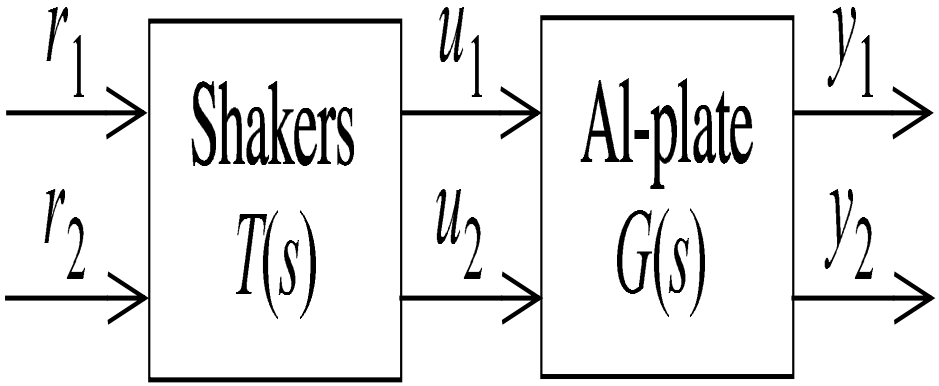
order [120/120]



# MIMO system

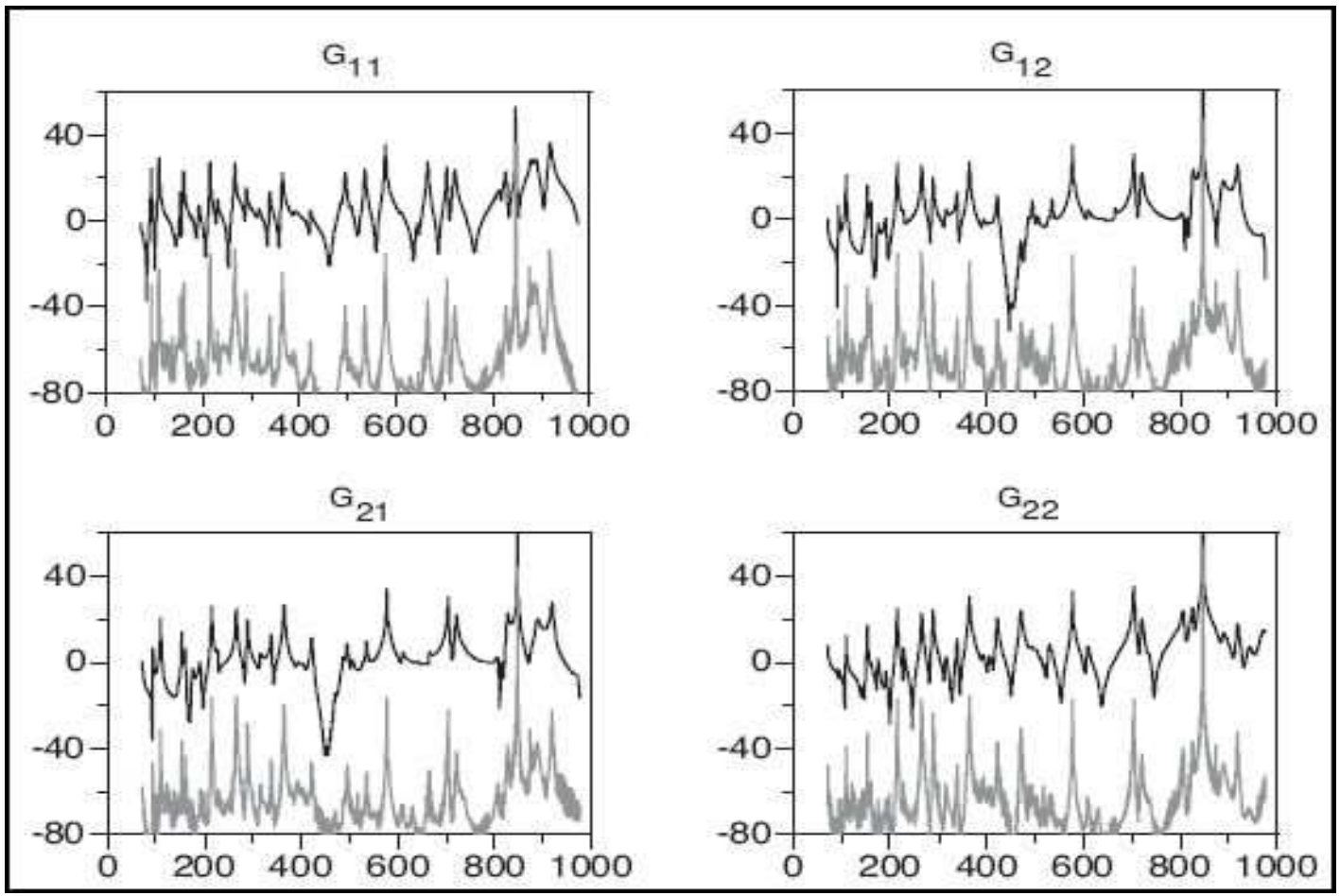





plexi plate; 2 stingers; hung by 2 nylon rods

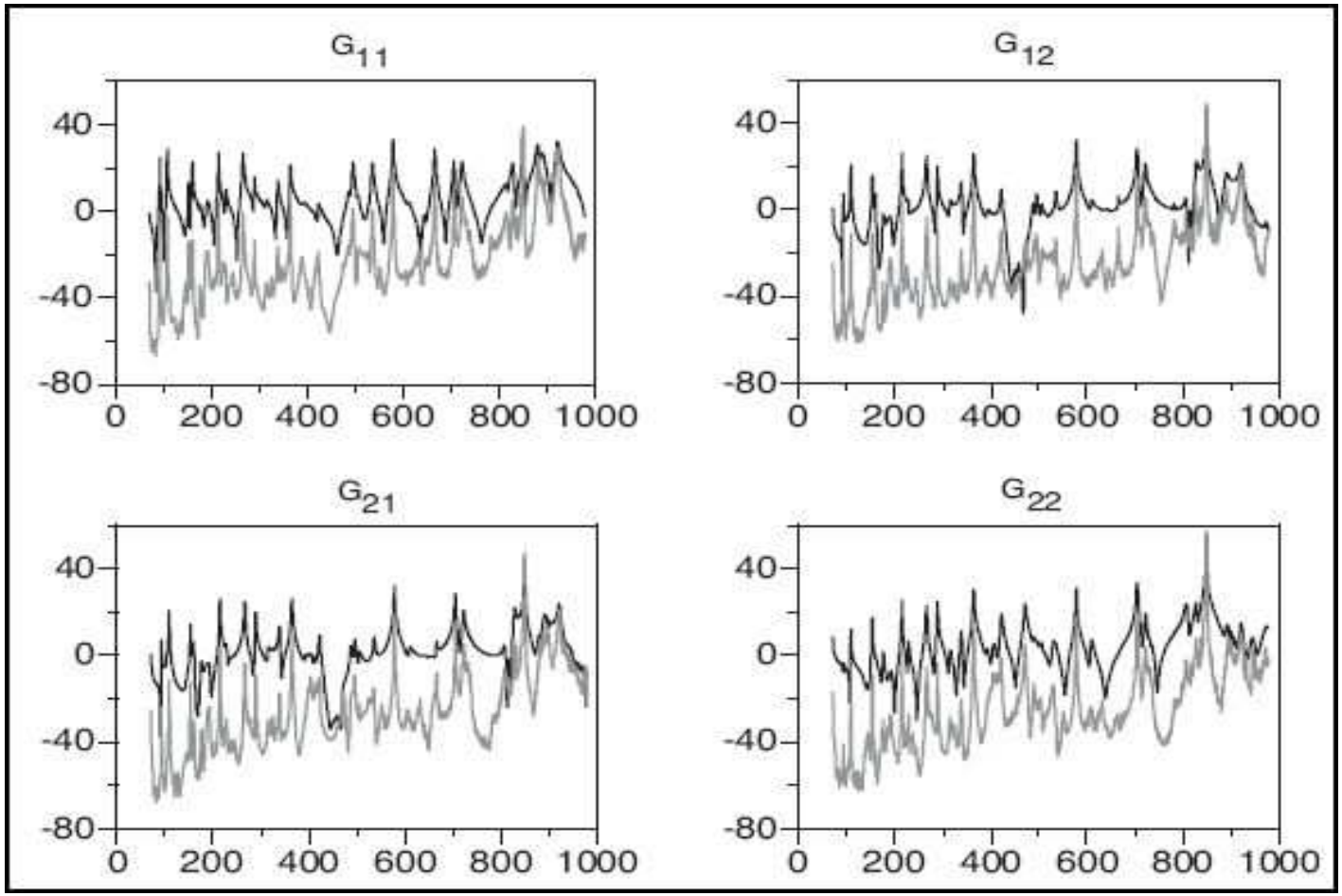
# MIMO system

The observations of the transfer matrix (dB) (black) including the variance (dB) (light grey)



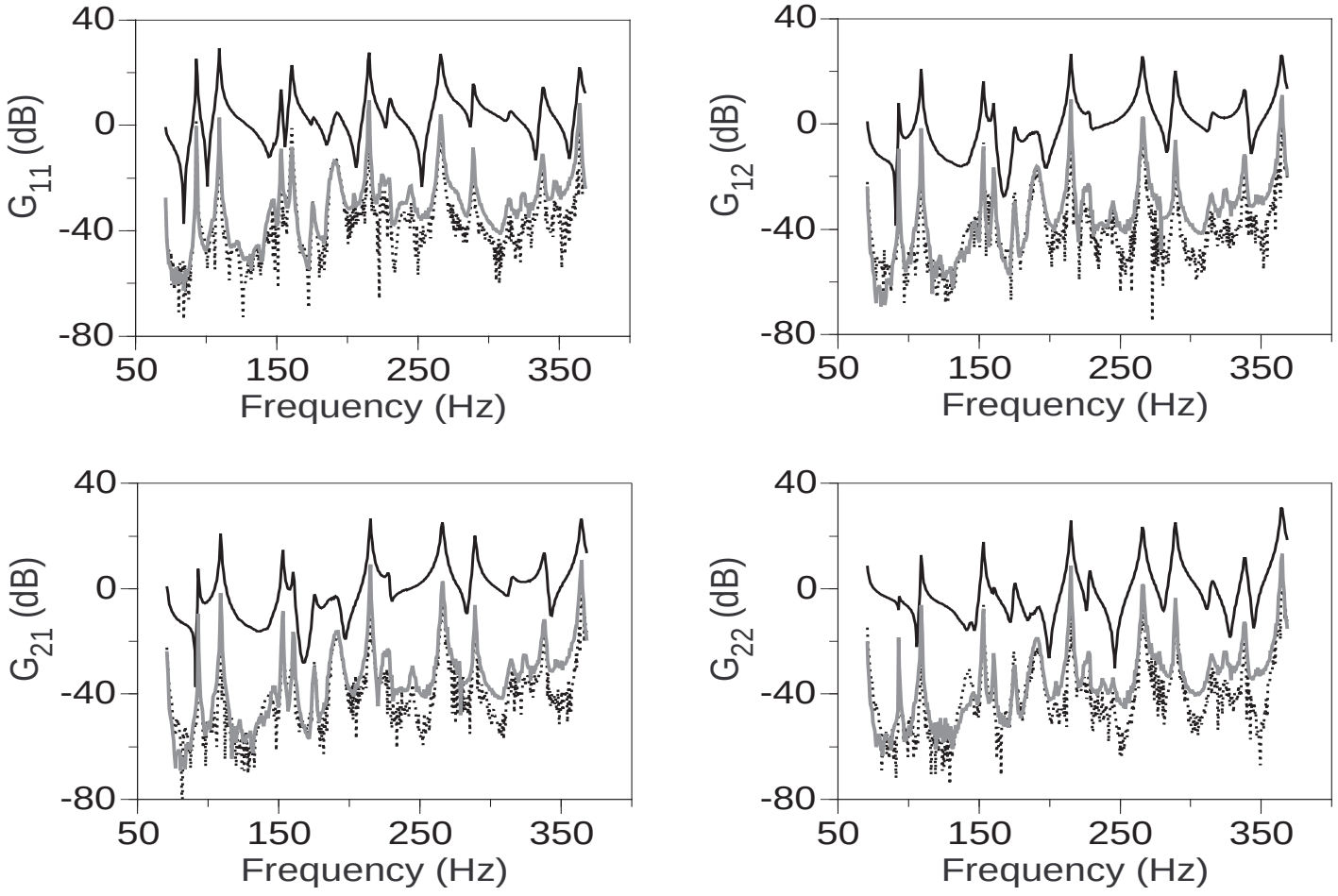
# MIMO system

Transfer matrix averaged over experiments (dB) (black) including the variance (dB) (light grey)



# MIMO system

order = [60/60], discrete time (detail)




estimation error almost always below 95% uncertainty bound.

# ORF and OP

- Recall for ORF we had $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{H}$
- For O(V)P: $\mathbf{Q}^H\mathbf{Z}^{-1}\mathbf{Q} = \mathbf{H}$ or $\mathbf{Q} = \mathbf{ZQH}$
- Thus OP = ORF with all poles $\mathbf{A} = \mathbf{0}$.

# ORF and OP

- Recall for ORF we had $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{H}$
- For O(V)P: $\mathbf{Q}^H\mathbf{Z}^{-1}\mathbf{Q} = \mathbf{H}$ or $\mathbf{Q} = \mathbf{ZQH}$
- Thus OP = ORF with all poles $\mathbf{A} = \mathbf{0}$.

# ORF and OP

- Recall for ORF we had $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{H}$
- For O(V)P: $\mathbf{Q}^H\mathbf{Z}^{-1}\mathbf{Q} = \mathbf{H}$ or $\mathbf{Q} = \mathbf{ZQH}$
- Thus OP = ORF with all poles $\mathbf{A} = \mathbf{0}$.

# Semiseparable matrices

- Link O(V)P and ORF suppose $\mathbf{H} = \mathbf{T}$
- $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{T}$
- $\mathbf{QS} = \mathbf{ZQ} - \mathbf{QA}$
- $\mathbf{Q}(\mathbf{S} + \mathbf{A}) = \mathbf{ZQ}$
- $\mathbf{S} + \mathbf{A} = \mathbf{Q}^H\mathbf{ZQ} \Leftrightarrow \mathbf{Q}(\mathbf{S} + \mathbf{A})\mathbf{Q}^H = \mathbf{Z}$
- $\mathbf{Q}\begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \phi_0^{-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

# Semiseparable matrices

- Link O(V)P and ORF suppose $\mathbf{H} = \mathbf{T}$
- $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{T}$
- $\mathbf{T}$ symmetric tridiagonal, irreducible (generic case)
  Suppose it is invertible.
  Its inverse is a symmetric SemiSeparable (SS) matrix: $\mathbf{S}$

$$\mathbf{S} = \mathbf{S}\begin{pmatrix} u_1, \ldots, u_n \\ v_1, \ldots, v_n \end{pmatrix}, \mathbf{S} = \mathbf{S}^T,\ \mathbf{S}_{ij} = u_i v_j, i \geq j$$

- $\mathbf{QS} = \mathbf{ZQ} - \mathbf{QA}$
- $\mathbf{Q}(\mathbf{S} + \mathbf{A}) = \mathbf{ZQ}$
- $\mathbf{S} + \mathbf{A} = \mathbf{Q}^H\mathbf{ZQ} \Leftrightarrow \mathbf{Q}(\mathbf{S} + \mathbf{A})\mathbf{Q}^H = \mathbf{Z}$
- $\mathbf{Q}\begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \phi_0^{-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

# Semiseparable matrices

- Link O(V)P and ORF suppose $\mathbf{H} = \mathbf{T}$
- $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{T}$
- $\mathbf{S} = \mathbf{T}^{-1}$
- $\mathbf{QS} = \mathbf{ZQ} - \mathbf{QA}$
- $\mathbf{Q}(\mathbf{S} + \mathbf{A}) = \mathbf{ZQ}$
- $\mathbf{S} + \mathbf{A} = \mathbf{Q}^H\mathbf{ZQ} \Leftrightarrow \mathbf{Q}(\mathbf{S} + \mathbf{A})\mathbf{Q}^H = \mathbf{Z}$
- $\mathbf{Q}\begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \phi_0^{-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

# Semiseparable matrices

- Link O(V)P and ORF suppose $\mathbf{H} = \mathbf{T}$
- $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{T}$
- $\mathbf{S} = \mathbf{T}^{-1}$
- $\mathbf{QS} = \mathbf{ZQ} - \mathbf{QA}$
- $\mathbf{Q}(\mathbf{S} + \mathbf{A}) = \mathbf{ZQ}$
- $\mathbf{S} + \mathbf{A} = \mathbf{Q}^H\mathbf{ZQ} \Leftrightarrow \mathbf{Q}(\mathbf{S} + \mathbf{A})\mathbf{Q}^H = \mathbf{Z}$
- $\mathbf{Q} \begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \phi_0^{-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

# Semiseparable matrices

- Link O(V)P and ORF suppose $\mathbf{H} = \mathbf{T}$
- $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{T}$
- $\mathbf{S} = \mathbf{T}^{-1}$
- $\mathbf{QS} = \mathbf{ZQ} - \mathbf{QA}$
- $\mathbf{Q}(\mathbf{S} + \mathbf{A}) = \mathbf{ZQ}$
- $\mathbf{S} + \mathbf{A} = \mathbf{Q}^H\mathbf{ZQ} \Leftrightarrow \mathbf{Q}(\mathbf{S} + \mathbf{A})\mathbf{Q}^H = \mathbf{Z}$
- $\mathbf{Q} \begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \phi_0^{-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

# Semiseparable matrices

- Link O(V)P and ORF suppose $\mathbf{H} = \mathbf{T}$
- $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{T}$
- $\mathbf{S} = \mathbf{T}^{-1}$
- $\mathbf{QS} = \mathbf{ZQ} - \mathbf{QA}$
- $\mathbf{Q}(\mathbf{S} + \mathbf{A}) = \mathbf{ZQ}$
- $\mathbf{S} + \mathbf{A} = \mathbf{Q}^H\mathbf{ZQ} \Leftrightarrow \mathbf{Q}(\mathbf{S} + \mathbf{A})\mathbf{Q}^H = \mathbf{Z}$
- $\mathbf{Q} \begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \phi_0^{-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

# Semiseparable matrices

- Link O(V)P and ORF suppose $\mathbf{H} = \mathbf{T}$
- $\mathbf{Q} = (\mathbf{ZQ} - \mathbf{QA})\mathbf{T}$
- $\mathbf{S} = \mathbf{T}^{-1}$
- $\mathbf{QS} = \mathbf{ZQ} - \mathbf{QA}$
- $\mathbf{Q}(\mathbf{S} + \mathbf{A}) = \mathbf{ZQ}$
- $\mathbf{S} + \mathbf{A} = \mathbf{Q}^H\mathbf{ZQ} \Leftrightarrow \mathbf{Q}(\mathbf{S} + \mathbf{A})\mathbf{Q}^H = \mathbf{Z}$
- $$\mathbf{Q}\begin{bmatrix} W_1 \\ \vdots \\ W_N \end{bmatrix} = \begin{bmatrix} \phi_0^{-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

# Inverse eigenvalue problem

- given $W$, $\mathbf{A}$, $\mathbf{Z}$

$$\mathbf{Q}\left[\begin{array}{c|c} \begin{matrix} W_1 \\ \vdots \\ W_N \end{matrix} & \mathbf{Z} \end{array}\right]\left[\begin{array}{c|c} 1 & \\ \hline & \mathbf{Q}^H \end{array}\right] = \left[\begin{array}{c|c} \begin{matrix} \phi_0^{-1} \\ \\ 0 \end{matrix} & \mathbf{S}+\mathbf{A} \end{array}\right]$$

- stable efficient recursive algorithm $O(N^2)$ or $O(nN)$
- can do real operations combining conjugate pair
- similar for discrete time systems
- also for MIMO systems

# Inverse eigenvalue problem

- given $W$, $\mathbf{A}$, $\mathbf{Z}$

$$\mathbf{Q}\left[\begin{array}{c|c} \begin{matrix} W_1 \\ \vdots \\ W_N \end{matrix} & \mathbf{Z} \end{array}\right]\left[\begin{array}{c|c} 1 & \\ \hline & \mathbf{Q}^H \end{array}\right] = \left[\begin{array}{c|c} \begin{matrix} \phi_0^{-1} \\ \\ 0 \end{matrix} & \mathbf{S}+\mathbf{A} \end{array}\right]$$

- stable efficient recursive algorithm $O(N^2)$ or $O(nN)$
- can do real operations combining conjugate pair
- similar for discrete time systems
- also for MIMO systems

# Inverse eigenvalue problem

- given $W$, $\mathbf{A}$, $\mathbf{Z}$

$$\mathbf{Q}\left[\begin{array}{c|c} \begin{matrix} W_1 \\ \vdots \\ W_N \end{matrix} & \mathbf{Z} \end{array}\right]\left[\begin{array}{c|c} 1 & \\ \hline & \mathbf{Q}^H \end{array}\right] = \left[\begin{array}{c|c} \begin{matrix} \phi_0^{-1} \\ \\ 0 \end{matrix} & \mathbf{S}+\mathbf{A} \end{array}\right]$$

- stable efficient recursive algorithm $O(N^2)$ or $O(nN)$
- can do real operations combining conjugate pair
- similar for discrete time systems
- also for MIMO systems

# Inverse eigenvalue problem

- given $W$, $\mathbf{A}$, $\mathbf{Z}$

$$\mathbf{Q}\left[\begin{array}{c|c} \begin{matrix} W_1 \\ \vdots \\ W_N \end{matrix} & \mathbf{Z} \end{array}\right]\left[\begin{array}{c|c} 1 & \\ \hline & \mathbf{Q}^H \end{array}\right] = \left[\begin{array}{c|c} \begin{matrix} \phi_0^{-1} \\ \\ 0 \end{matrix} & \mathbf{S}+\mathbf{A} \end{array}\right]$$

- stable efficient recursive algorithm $O(N^2)$ or $O(nN)$
- can do real operations combining conjugate pair
- similar for discrete time systems
- also for MIMO systems

# Inverse eigenvalue problem

- given $W$, $\mathbf{A}$, $\mathbf{Z}$

$$\mathbf{Q}\left[\begin{array}{c|c} \begin{matrix} W_1 \\ \vdots \\ W_N \end{matrix} & \mathbf{Z} \end{array}\right]\left[\begin{array}{c|c} 1 & \\ \hline & \mathbf{Q}^H \end{array}\right] = \left[\begin{array}{c|c} \begin{matrix} \phi_0^{-1} \\ \\ 0 \end{matrix} & \mathbf{S}+\mathbf{A} \end{array}\right]$$

- stable efficient recursive algorithm $O(N^2)$ or $O(nN)$
- can do real operations combining conjugate pair
- similar for discrete time systems
- also for MIMO systems

# References OVP



A. Bultheel, M. Van Barel, Y. Rolain, J. Schoukens. *Robust rational approximation for identification*, *Proc. CDC conference*, 2001


A. Bultheel, M. Van Barel, Y. Rolain, R. Pintelon. *Numerically robust transfer function modeling from noisy frequency domain data*,

*IEEE Trans. Autom. Control* 50 (2002) 1835-1839.


A. Bultheel, M. Van Barel. *Vector orthogonal polynomials and least squares approximation*,
*SIAM J. Matrix Anal. Appl.* 16 (1995) 863-885.


M. Van Barel, A. Bultheel. *Orthogonal polynomial vectors and least squares approximation for a discrete inner product*, *Electron. Trans. Numer. Anal.* 3 (1995) 1-23.

# References ORF



A. Bultheel, P. González-Vera, E. Hendriksen, and O. Njåstad. *Orthogonal rational functions*. Cambridge University Press, 1999.


A. Bultheel, M. Van Barel, P. Van gucht. *Orthogonal bases in discrete least squares rational approximation.* *J. Comput. Appl. Math.*, 164-165 (2004) 175-194.


P. Van gucht, A. Bultheel. *Using orthogonal rational functions for system identification: numerical aspects*, *IEEE Trans. Automat. Control*, 48 (2003) 705-709.


A. Bultheel, P. González-Vera, E. Hendriksen, and O. Njåstad. *Orthogonal rational functions and tridiagonal matrices*, *J. Comput. Appl. Math.* 153 (2003) 89-97.

# References SS matrices



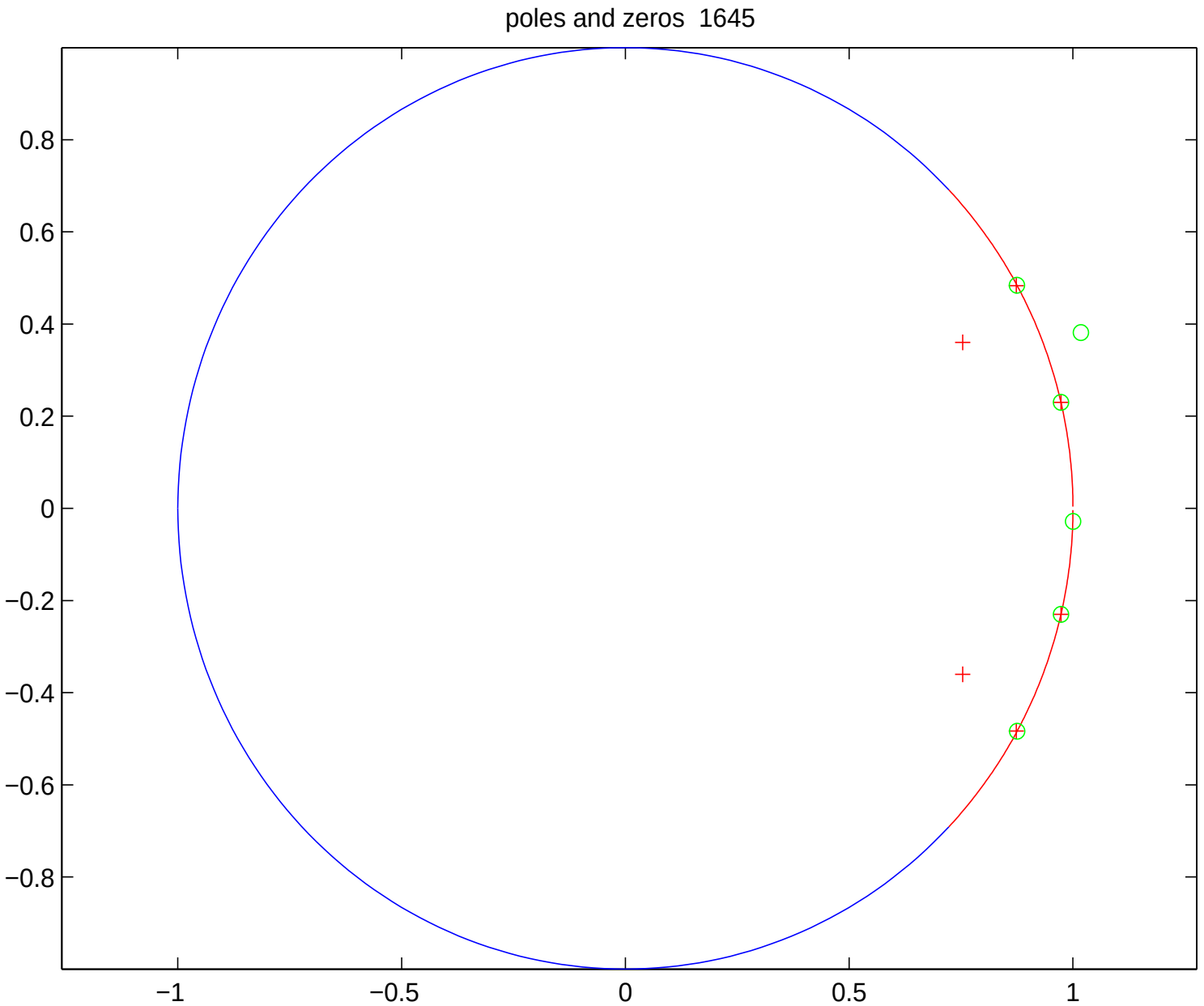
D. Fasino, L. Gemignani. *Direct and inverse eigenvalue problems for diagonal-plus-semiseparable matrices Numerical algorithms* 34 (2001) 313-324.


M. Van Barel, D. Fasino, L. Gemignani, N. Mastronardi. *Orthogonal rational functions and structured matrices*, *SIAM Journal on Matrix Analysis and its Applications* 26 (2005) 810-829.


R. Vandebril, M. Van Barel, G. Golub, and N. Mastronardi, *A bibliography on semiseparable matrices*, *Calcolo* 42 (2005) 249-270.


`www.cs.kuleuven.be/~ade/`

# Robotarm



# Bandpass

