

Welcome to

FRTN15 Predictive Control

Lecture 1

Lund University

Dept Automatic Control

2012

<http://www.control.lth.se/course/FRTN15/>

© Rolf Johansson, 2011

FRTN15 Predictive Control

1



Predictive Control



© Rolf Johansson, 2011

FRTN15 Predictive Control

3



Why Prediction and Adaptation?

- Limitations of linear feedback
- Model-based control
- Optimization
 - Minimization of tracking errors
 - Covariance minimization
 - Minimization of control actions
 - Minimization of tracking errors subject to control constraints
- Time-varying Exosystems & Disturbances
- Adaptation and Learning

© Rolf Johansson, 2011

FRTN15 Predictive Control

2



Predictive Control



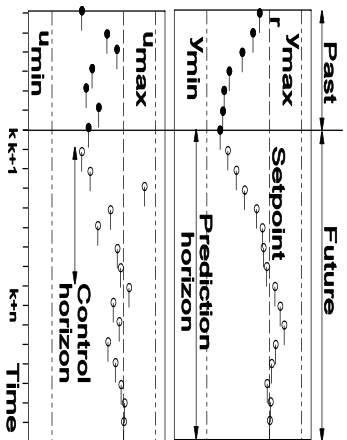
© Rolf Johansson, 2011

FRTN15 Predictive Control

4



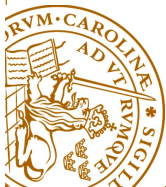
Predictive Control



© Rolf Johansson, 2011

FRTN15 Predictive Control

5



Predictive Control



© Rolf Johansson, 2011

FRTN15 Predictive Control

6



Predictive Control



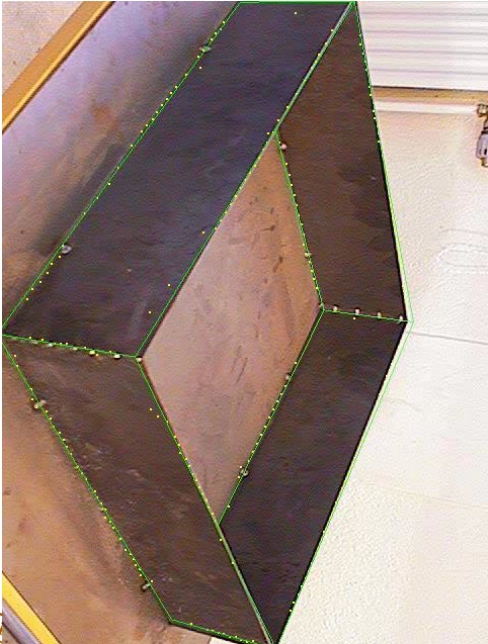
© Rolf Johansson, 2011

FRTN15 Predictive Control

7



Predictive Control



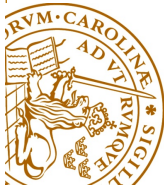
© Rolf Johansson, 2011

FRTN15 Predictive Control

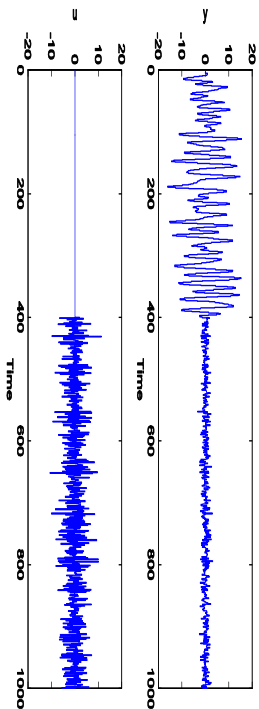
8



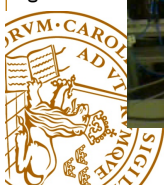
Frida Robot



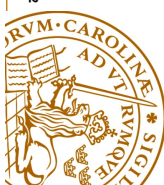
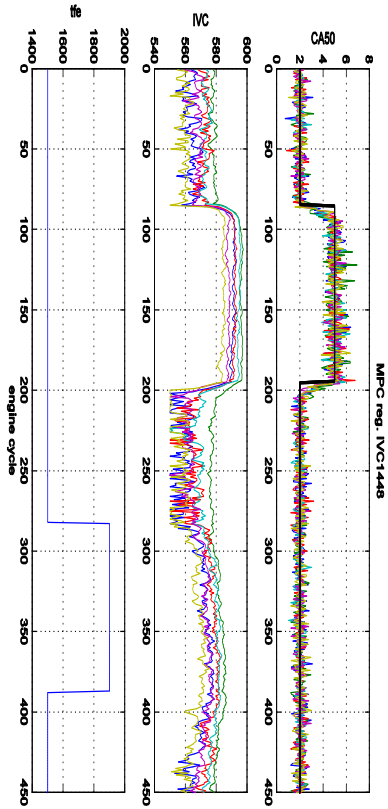
Minimum Variance Control



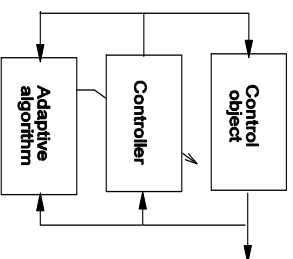
Frida Robot



HCCI MPC Engine Control



Adaptation

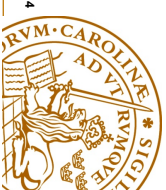
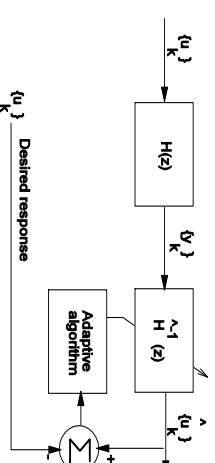


Adaptive Schemes

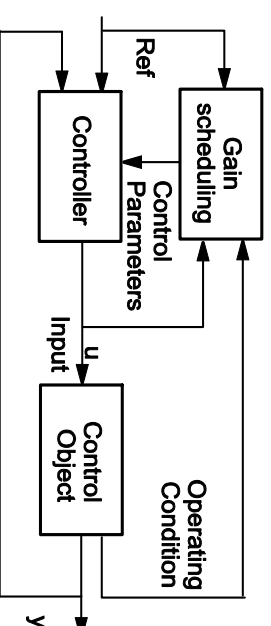
- Gain Scheduling
- Autotuning
- Model Reference Adaptive Control
- Self-Tuning Control
- Extremum Control
- Iterative Learning Control (ILC)



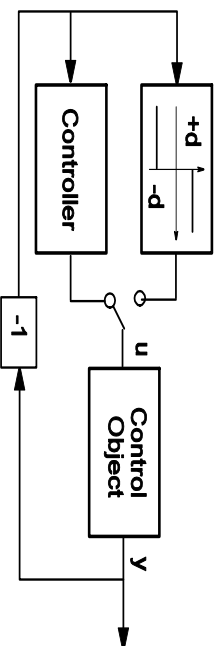
Equalization



Gain Scheduling



Autotuning



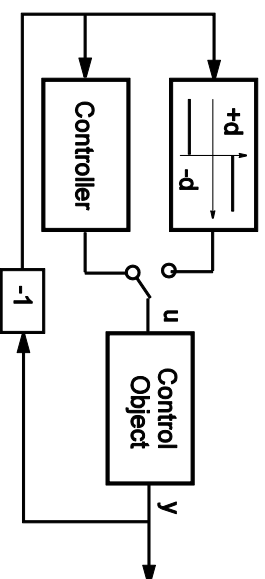
© Rolf Johansson, 2011

FRTN15 Predictive Control

17



Autotuning



© Rolf Johansson, 2011

FRTN15 Predictive Control

19



Autotuning

$$G_0(s) = \frac{K_0}{1 + s\tau} e^{-sL}$$

where k is the static gain, L is the latency (or apparent time delay) and τ a time constant.
Tuning of a PID regulator

$$u(t) = K_c(e(t) + \frac{1}{T_I} \int_0^t e(s) ds + T_D \frac{de}{dt})$$

$$v(t) = f(v(t))$$

$$P(t) = K_c(\beta u_c(t) - y(t))$$

$$\frac{dI}{dt} = \frac{K_c}{T_I} (u_c(t) - y(t)) + \frac{1}{T_I} (v(t) - u(t))$$

$$\frac{T_D dD}{N dt} = -D - K_c T_D \frac{dy}{dt}$$

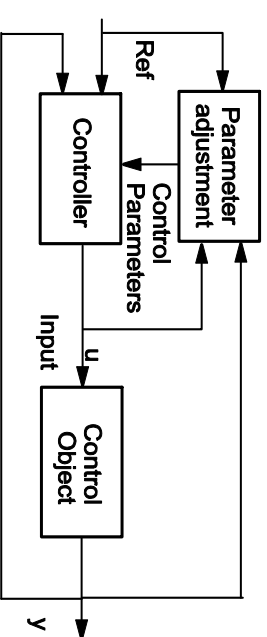
© Rolf Johansson, 2011

FRTN15 Predictive Control

18



Adaptive Control



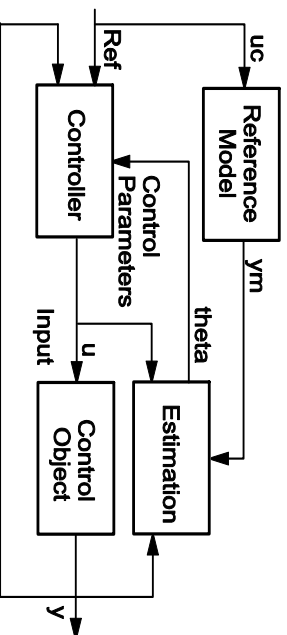
© Rolf Johansson, 2011

FRTN15 Predictive Control

20



Model Reference Adaptive Control



© Rolf Johansson, 2011

FRTN15 Predictive Control

21



Adaptive Control

The polynomial $T(z^{-1})$ can, for instance, simply be chosen such that the controlled system has static gain equal to 1—i.e.,

$$T(z^{-1}) = \frac{P(1)}{B(1)} = \frac{1 + \sum_{i=1}^3 p_i}{\sum_{i=1}^2 b_i}$$

These choices suggest the controller

$$u_k = -r_1 u_{k-1} + s_0 y_k + s_1 y_{k-1} + t_0 r_k$$

© Rolf Johansson, 2011

FRTN15 Predictive Control

23



Adaptive Control

Let the system

$$\frac{Y(z)}{U(z)} = H(z) = \frac{b_1 z + b_2}{z^2 + a_1 z + a_2}$$

be controlled by means of output feedback control designed for pole assignment with the denominator polynomial

$$P(z) = z^3 + p_1 z^2 + p_2 z + p_3$$

© Rolf Johansson, 2011

FRTN15 Predictive Control

22



Adaptive Control

Equation including a Sylvester matrix is then reduced to the linear equation

$$\begin{pmatrix} 1 & b_1 & 0 \\ a_1 & b_2 & b_1 \\ a_2 & 0 & b_2 \end{pmatrix} \begin{pmatrix} r_1 \\ s_0 \\ s_1 \end{pmatrix} = \begin{pmatrix} p_1 - a_1 \\ p_2 - a_2 \\ p_3 \end{pmatrix}$$

or

$$A(\theta)\vartheta = b(\theta), \text{ with } \vartheta = \begin{pmatrix} r_1 \\ s_0 \\ s_1 \end{pmatrix}$$

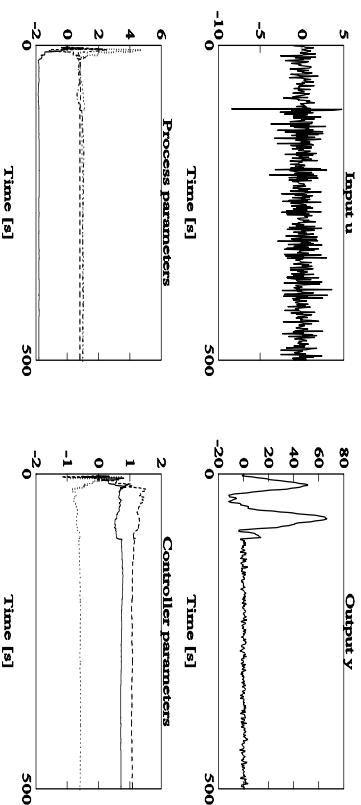
© Rolf Johansson, 2011

FRTN15 Predictive Control

24



Adaptive Control



Indirect adaptive control. The second-order process is controlled by an output feedback control law with pole assignment to the origin $z=0$. The histories of the input u , output y , and the estimated process parameters and the controller parameters are displayed.

© Rolf Johansson, 2011

FRTN15 Predictive Control

25



Discrete-time Signals

Periodic uniform sampling described by

- sampling frequency ω_s
- sampling period h

- Reconstruction of a continuous-time signal from a sequence of samples can be made provided that there is no signal energy beyond the *Nyquist frequency*

$$\omega_N = \frac{1}{2}\omega_s$$

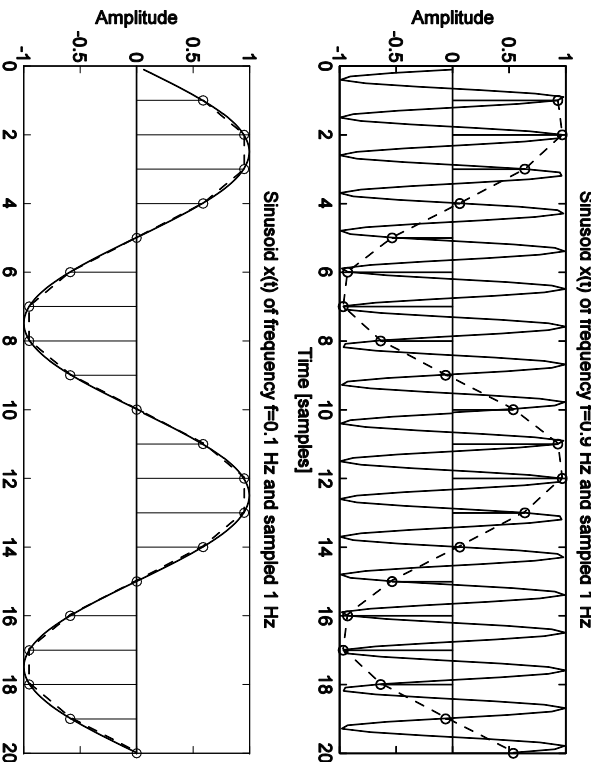
Note: Imperfections in uniform sampling

- Jitter
- Ethernet

© Rolf Johansson, 2011

FRTN15 Predictive Control

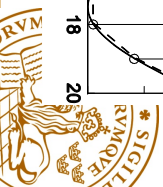
26



© Rolf Johansson, 2011

FRTN15 Predictive Control

27



Discretized Linear Systems

Under an additional assumption of zero-order-hold properties of the input u , sampled linear systems may be discretized as

$$\begin{cases} \frac{dx}{dt} = A_c x + B_c u \\ y = Cx \end{cases} \quad \begin{cases} x_{k+1} = A x_k + B u_k \\ y_k = C x_k \end{cases}$$

with

$$A = e^{A_c h}, \quad B = \int_0^h e^{A_c t} B_c dt$$

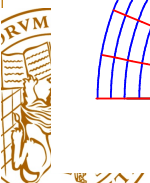
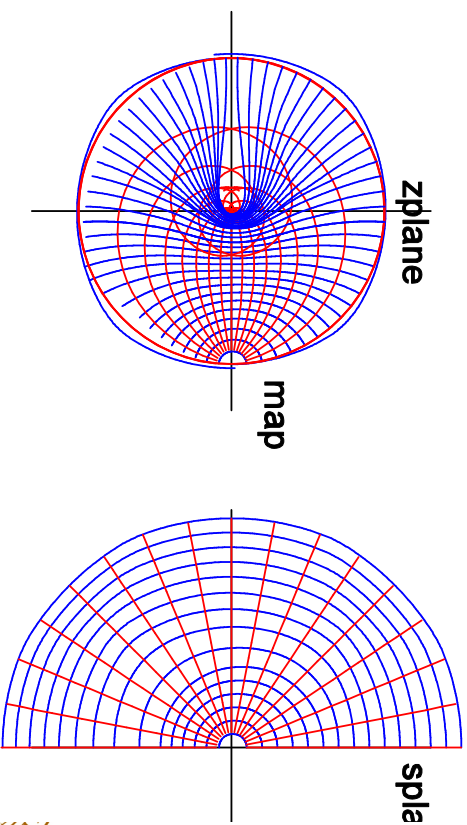
Note that the discretized system is a linear time-invariant system.

© Rolf Johansson, 2011

FRTN15 Predictive Control

28





Example: Double Integrator

The continuous-time system

$$\frac{dx}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

is discretized as

$$x_{k+1} = \begin{bmatrix} 1 & h \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} h^2 \\ h \end{bmatrix} u_k$$

$$y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k$$

under assumptions of uniform sampling with sampling period h and zero-order-hold input u .



The z-Transform

The one-sided z-transform $\mathcal{Z}\{z_k\}$ of a sequence $\{x_k\}_{k=0}^{\infty}$

$$X(z) = \mathcal{Z}\{z_k\} = \sum_{k=0}^{\infty} x_k z^{-k}$$

For further mathematical detail, see Appendix



Example: The z-Transform

A step function

$$u_k = \begin{cases} 1, & k \geq 0 \\ 0, & k < 0 \end{cases}$$

has the z-transform

$$U(z) = 1 + z^{-1} + z^{-2} + z^{-3} + \dots = \frac{z}{z-1}$$



z-transform

	Time function	$\hat{=}$ z transform
Linearity		$\mathcal{Z}\{af + bg\} = a\mathcal{Z}\{f\} + b\mathcal{Z}\{g\}$
Convolution		$\mathcal{Z}\{f * g\} = \mathcal{Z}\{f\} \cdot \mathcal{Z}\{g\}$ $\mathcal{Z}\{f \cdot g\} = \mathcal{Z}\{f\} * \mathcal{Z}\{g\}$
Time translation		$\mathcal{Z}\{f((k-d)h)\} = z^{-d}\mathcal{Z}\{f(kh)\}$
Multiplication		$\mathcal{Z}\{a^k f(k)\} = F_z(a^{-1}z)$
Final-value theorem		$f(\infty) = \lim_{z \rightarrow 1} (1 - z^{-1})F(z)$
Initial-value theorem		$f(0) = \lim_{z \rightarrow \infty} F_z(z)$

Table 2.1. Properties of the z-transform



Impulse Response

The impulse response or the pulse response—*i.e.*, the output response to the unit pulse input—is

$$y_k = CA^{k-1}B, \\ \text{for } u_k = \delta_k = \begin{cases} 1, & k=0 \\ 0, & k \neq 0 \end{cases} \quad k = 1, 2, 3, \dots$$

The components of the pulse response

$$g_k = CA^{k-1}B, \quad k = 1, 2, 3, \dots$$

are known as the *Markov parameters*



State-Space Systems

Consider the discrete-time state-space system

$$x_{k+1} = Ax_k + Bu_k \\ y_k = Cx_k$$

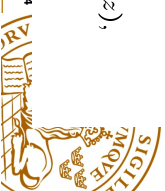
Applications of the z-transform to the state-space system gives

$$zX(z) = AX(z) + BU(z), \quad X(z) = \mathcal{Z}\{x_k\} = \sum_{k=0}^{\infty} x_k z^{-k}$$

$$Y(z) = CX(z)$$

Elimination of $X(z)$ provides the transfer function $G(z)$ obtained as

$$X(z) = (zI - A)^{-1}BU(z) \\ Y(z) = CX(z) = C(zI - A)^{-1}BU(z) = G(z)U(z), \\ G(z) = \frac{Y(z)}{U(z)} = C(zI - A)^{-1}B$$



Impulse Response (cont'd)

In response to an arbitrary input sequence $\{u_k\}_{k=0}^{\infty}$ and with non-zero initial condition x_0 , the output is

$$y_k = CA^k x_0 + \sum_{j=0}^k h_{k-j} u_j \\ = CA^k x_0 + \sum_{j=0}^k CA^{k-j-1} B u_j, \quad k = 1, 2, 3, \dots$$



Transfer Function

The transfer function $G(z)$ of the linear system is defined by

$$\begin{aligned} Y(z) &= G(z)U(z), \\ G(z) &= C(zI - A)^{-1}B \\ &= \sum_{k=0}^{\infty} g_k z^{-k} = \mathcal{Z}\{g_k\} \end{aligned}$$

The *transfer operator* is

$$G(q) = C(qI - A)^{-1}B$$

© Rolf Johansson, 2011

FRTN15 Predictive Control

37



Spectra

The z-spectra are

$$\begin{aligned} S_{yu}(z) &= G(z)S_{uu}(z) \\ &= \text{input-output cross z-spectrum} \\ S_{yy}(z) &= G(z)S_{uu}(z)G^T(z^{-1}) \\ &= \text{output z-autospectrum} \end{aligned}$$

Evaluated the spectra on the unit circle

$$\begin{aligned} S_{yu}(z) &= G(z)S_{uu}(z) \text{ for } z = e^{i\omega h} \\ S_{yy}(z) &= G(z)S_{uu}(z)G^T(z^{-1}) \text{ for } z = e^{i\omega h} \end{aligned}$$

© Rolf Johansson, 2011

FRTN15 Predictive Control

38



Difference Equations

Input-output models may take on the form of *difference equations*

$$\begin{aligned} y_k &+ a_1 y_{k-1} + \dots + a_n y_{k-n} \\ &= b_1 u_{k-1} + b_2 u_{k-2} + \dots + b_n u_{k-n} \end{aligned}$$

or

$$\begin{aligned} y_k &= -a_1 y_{k-1} - \dots - a_n y_{k-n} \\ &+ b_1 u_{k-1} + b_2 u_{k-2} + \dots + b_n u_{k-n} \end{aligned}$$

© Rolf Johansson, 2011

FRTN15 Predictive Control

39



Difference Equations (cont'd)

A difference equation may be converted to a state-space systems by introducing

$$\begin{aligned} \xi_k &= -a_1 \xi_{k-1} - \dots - a_n \xi_{k-n} + u_k \\ x_k &= [\xi_{k-1} \ \xi_{k-2} \ \dots \ \xi_{k-n}]^T \end{aligned}$$

Then, the following state-space system—the *controllable canonical form*—will reproduce the difference equation output

$$x_{k+1} = \begin{bmatrix} -a_1 & -a_2 & \dots & -a_n \\ 1 & 0 & \dots & 0 \\ & \ddots & & \vdots \\ & & 1 & 0 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u_k$$

$$y_k = [b_1 \ b_2 \ \dots \ b_n] x_k$$

© Rolf



Difference Equations (cont'd)

A variety of difference equations are used such as the *autoregressive (AR) models*

$$y_k = -a_1 y_{k-1} - \dots - a_n y_{k-n}$$

and *moving average (MA) models*

$$y_k = b_1 u_{k-1} + b_2 u_{k-2} + \dots + b_n u_{k-n}$$

Linear models of colored noise are often formulated by means of *autoregressive moving average (ARMA) models*

$$y_k + a_1 y_{k-1} + \dots + a_n y_{k-n} = w_k + c_1 w_{k-1} + c_2 w_{k-2} + \dots + c_n w_{k-n}$$

where $\{y_k\}$ is a filtered $\{w_k\}$

© Rolf Johansson, 2011

FRNT15 Predictive Control

41



ARMAX Models

$$y_k = -a_1 y_{k-1} - \dots - a_n y_{k-n} + b_1 u_{k-1} + b_2 u_{k-2} + \dots + b_n u_{k-n} + w_k + c_1 w_{k-1} + c_2 w_{k-2} + \dots + c_n w_{k-n}$$

or

$$\begin{aligned} A(z^{-1})Y(z) &= B(z^{-1})U(z) + C(z^{-1})W(z) \\ A(z^{-1}) &= 1 + a_1 z^{-1} + \dots + a_n z^{-n} \\ B(z^{-1}) &= b_1 z^{-1} + \dots + b_n z^{-n} \\ C(z^{-1}) &= c_0 + c_1 z^{-1} + \dots + c_n z^{-n} \end{aligned}$$

The transfer function relationships are

$$Y(z) = \frac{B(z^{-1})}{A(z^{-1})}U(z) + \frac{C(z^{-1})}{A(z^{-1})}W(z)$$

© Rolf Johansson, 2011

FRNT15 Predictive Control

43



ARMA models

The ARMA models obey

$$Y(z) = -a_1 z^{-1}Y(z) - \dots - a_n z^{-n}Y(z) + W(z) + c_1 z^{-1}W(z) + \dots + c_n z^{-n}W(z)$$

in z-transformed signals $Y(z)$, $W(z)$ and generating polynomials $A(z^{-1})$, $C(z^{-1})$

$$A(z^{-1})Y(z) = C(z^{-1})W(z)$$

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_n z^{-n}$$

$$C(z^{-1}) = 1 + c_1 z^{-1} + \dots + c_n z^{-n}$$

© Rolf Johansson, 2011

FRNT15 Predictive Control

42



Forward vs backward shift form

$$A(z) = z^n + a_1 z^{n-1} + \dots + a_n$$

$$A^*(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_n z^{-n}$$

$$A(z) = z^n A^*(z^{-1}) \text{ where } n = \deg(A)$$

Example:

$$A(z)y = B(z)u \Leftrightarrow A^*(z^{-1})y = z^{-d} B^*(z^{-1})u$$

where $d = \deg(A) - \deg(B)$

© Rolf Johansson, 2011

FRNT15 Predictive Control

44



System Interconnections

Series interconnection

$$\begin{cases} Y_1(z) = H_1(z)U_1(z) \\ Y_2(z) = H_2(z)U_2(z) \end{cases}$$

$$\begin{aligned} Y_2(z) &= H_2(z)U_2(z) = H_2(z)Y_1(z) \\ &= H_2(z)H_1(z)U_1(z) \end{aligned}$$



System Interconnections

Feedback interconnection

$$\begin{cases} Y_1(z) = H_1(z)U_1(z) \\ Y_2(z) = H_2(z)U_2(z) \end{cases}$$

with

$$\begin{cases} U_2(z) = Y_1(z) \\ U_1(z) = -Y_2(z) + U_0(z) \end{cases}$$

with

$$\begin{aligned} Y_1(z) &= H_1(z)U_1(z) = H_1(z)(-Y_2(z) + U_0(z)) \\ &= H_1(z)(-H_2(z)Y_1(z) + U_0(z)) \\ Y_1(z) &= (I + H_2(z)H_1(z))^{-1}U_0(z) \\ Y_2(z) &= H_2(z)(I + H_2(z)H_1(z))^{-1}U_0(z) \end{aligned}$$

System Interconnections

Parallel interconnection

$$Y(z) = Y_1(z) + Y_2(z)$$

with

$$\begin{cases} Y_1(z) = H_1(z)U(z) \\ Y_2(z) = H_2(z)U(z) \end{cases}$$

with

$$\begin{aligned} Y(z) &= H(z)U(z), \\ H(z) &= H_1(z) + H_2(z) \end{aligned}$$



Stability

The set of eigenvalues $\{\lambda_i\}$ of the matrix A of the discrete-time state-space system

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k \end{aligned}$$

will determine the stability properties of the system.



Stability

For $z = \lambda_i \in \mathbb{C}$, the corresponding transfer function

$$G(z) = C(zI - A)^{-1}B$$

will assume infinite value and these points are called the *poles* of the transfer function.

The values of z for which $G(z) = 0$ are called the *zeros* of the transfer function.



Stability

If all poles $z = \lambda_i \in \mathbb{C}$ are such that $|z| = |\lambda_i| < 1$, then the system is asymptotically stable.

Hence, the stability condition for z requires that all poles be within the unit circle $|z| = 1$ in the complex plane.



Example: Double Integrator

The discrete-time double integrator

$$x_{k+1} = \begin{bmatrix} 1 & h \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} h^2 \\ h \end{bmatrix} u_k$$

$$y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k$$

has a transfer function

$$\begin{aligned} G(z) &= C(zI - A)^{-1}B = \\ &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} z-1 & -h \\ 0 & z-1 \end{bmatrix}^{-1} \begin{bmatrix} h^2 \\ h \end{bmatrix} \\ &= \frac{h^2 z}{(z-1)^2} \end{aligned}$$

with two poles at $z = 1 - i.e.$, the system is not asymptotically stable.



Example: Double Integrator

The state feedback control

$$u_k = -Kx_k = -\begin{bmatrix} k_1 & k_2 \end{bmatrix} x_k$$

results in the closed-loop system

$$\begin{aligned} x_{k+1} &= (A - BK)x_k \\ &= \begin{bmatrix} 1 - k_1 h^2/2 & h - k_2 h^2/2 \\ -k_1 h & 1 - k_2 h \end{bmatrix} x_k \end{aligned}$$

with the characteristic equation

$$z^2 + (k_1 \frac{h^2}{2} + k_2 h - 2)z + (k_1 \frac{h^2}{2} - k_2 h + 1) = 0$$



Example: Double Integrator

Stabilizing pole assignment to $z = 0$ is accomplished by choosing k_1, k_2 such that the characteristic equation become $z^2 = 0$ or

$$\begin{bmatrix} \frac{1}{2}h^2 & h \\ \frac{1}{2}h^2 & -h \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \end{bmatrix}$$

that is

$$K = \begin{bmatrix} k_1 & k_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{h^2} & \frac{3}{2h} \end{bmatrix}$$

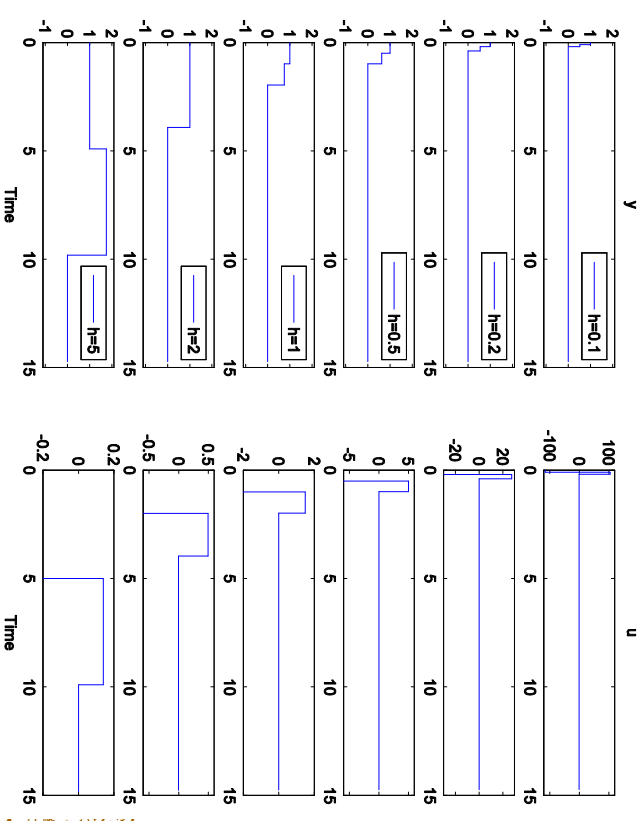
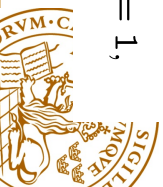
Linear Disturbance Models

Discrete-time impulses are modeled with a duration of one sampling period and with unit amplitude

$$\delta(kh) = \begin{cases} 1, & k = 0 \\ 0, & k \neq 0 \end{cases}, \quad \mathcal{Z}\{\delta(kh)\} = 1 \quad \text{Impulse}$$

For impulses in continuous-time systems, the Dirac impulse $\delta(t)$ is the most commonly used signal model.

$$\delta(t) = \begin{cases} \infty, & t = 0, \\ 0, & t \neq 0 \end{cases} \quad \text{with} \quad \int_{-\infty}^{\infty} \delta(t) dt = 1,$$



Steps and Ramps

$$\mathbf{1}(kh) = f(kh) = \begin{cases} 1, & k \geq 0 \\ 0, & k < 0 \end{cases}$$

$$F(z) = \frac{z}{z-1}$$

$$f(kh) = \begin{cases} kh, & k \geq 0 \\ 0, & k < 0 \end{cases}$$

$$F(z) = \frac{hz}{(z-1)^2}$$



Periodic Disturbances

Sinusoidal disturbances are often used to model periodic disturbances or loads.

$$f(kh) = \begin{cases} \sin \omega kh, & k \geq 0 \\ 0, & k < 0 \end{cases}$$

$$F(z) = \frac{z \sin \omega h}{z^2 - 2z \cos \omega h + 1}$$

© Rolf Johansson, 2011

FRTN15 Predictive Control

57



Useful Matlab Commands

- `s=tf('s');` % define s-operator
- `z=tf('z',h);` % define z (h sample time)
- `Gc=1/s^2;` % define system
- `Gd=c2d(G,0.1);` % sampling system
- `ss,tf,zpk` % system representations
- `acker, place` % pole placement
- `help control`

© Rolf Johansson, 2011

FRTN15 Predictive Control

59



Periodic Disturbances

Disturbances with sinusoidals are often used to model periodic disturbances or loads.

$$f(kh) = \begin{cases} \sum_m \sin \omega m kh, & k \geq 0 \\ 0, & k < 0 \end{cases}$$

$$F(z) = \sum_m \frac{z \sin m\omega h}{z^2 - 2z \cos m\omega h + 1}$$

© Rolf Johansson, 2011

FRTN15 Predictive Control

58



What to do now

- Get book
- Read chap 1-2. (Reading notes available on web)
- Go to Exercise 1 (computer simulation of adaptive system)
- Check out the matlab functions
- Next lecture: Parameter estimation (chap 3-4)

© Rolf Johansson, 2011

FRTN15 Predictive Control

60

