Control Theory

Stefan Simrock
DESY, Hamburg, Germany

Abstract
In engineering and mathematics, control theory deals with the behavior of dynamical systems. The desired output of a system is called the reference. When one or more output variables of a system need to follow a certain reference over time, a controller manipulates the inputs to a system to obtain the desired effect on the output of the system.

Rapid advances in digital system technology have radically altered the control design options. It has become routinely practicable to design very complicated digital controllers and to carry out the extensive calculations required for their design. These advances in implementation and design capability can be obtained at low cost because of the widespread availability of inexpensive and powerful digital processing platforms and high speed analog IO devices.

1 Introduction

The emphasis of the tutorial is on designing digital controls to achieve good dynamic response and small errors while using signals that are sampled in time and quantized in amplitude. Both transform (classical control) and state-space (modern control) methods are described and applied to illustrative examples. The transform methods emphasized are the root-locus method of Evans and frequency response. The state-space methods developed are the technique of pole assignment augmented by an estimator (observer) and optimal quadratic-loss control. The optimal control problems use the steady-state constantgain solution. Other topics covered are system identification and non-linear control.
System Identification is a general term to describe mathematical tools and algorithms that build dynamical models from measured data. A dynamical model in this context is a mathematical description of the dynamic behavior of a system or process.
Non-linear control is a sub-division of control engineering which deals with the control of non-linear systems. The behavior of a non-linear system cannot be described as a linear function of the state of that system or the input variables to that system. There exist several well-developed techniques for analyzing nonlinear feedback systems. Control design techniques for non-linear systems also exist. These can be subdivided into techniques which attempt to treat the system as a linear system in a limited range of operation and use (well-known) linear design techniques for each region. The main components of a control system (Fig. 1) are the plant and a sensor which is used to measure the variable to be controlled. The closed loop system (Fig. 2) requires a connection from the system outputs to the inputs. For feedback control a compensator amplifies and filters the error signal (Fig. 3) to manipulate the input signals the plant to minimize the errors of the variables.

Objective: The course on control theory is concerned with the analysis and design of closed loop control systems.

Analysis: Closed loop system is given → determine characteristic or behavior.

Design: Desired system characteristics or behavior are specified → configure or synthesize closed loop system.

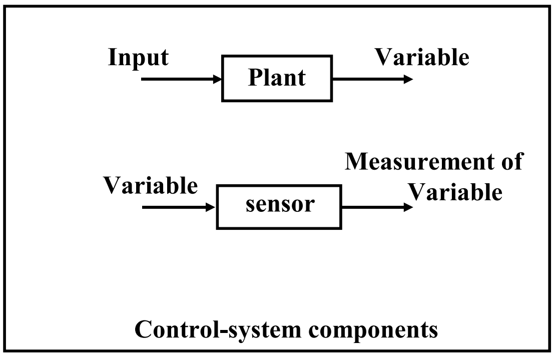


Fig. 1:

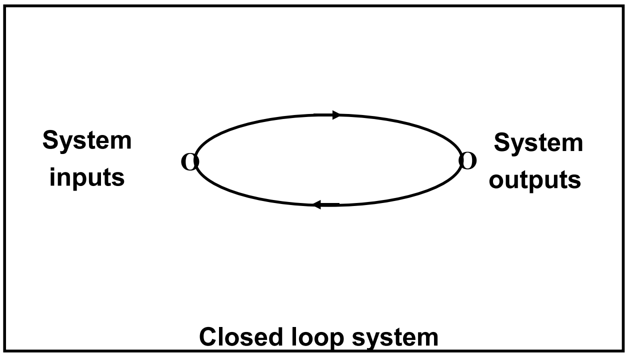


Fig. 2:

<u>Definition:</u> A closed-loop system is a system in which certain forces (we call these inputs) are determined, at least in part, by certain responses of the system (we call these outputs)

Definitions:

- The system for measurement of a variable (or signal) is called a sensor
- A **plant** of a control system is the part of the system to be controlled
- The **compensator** (or controller or simply filter) provides satisfactory characteristics for the total system.

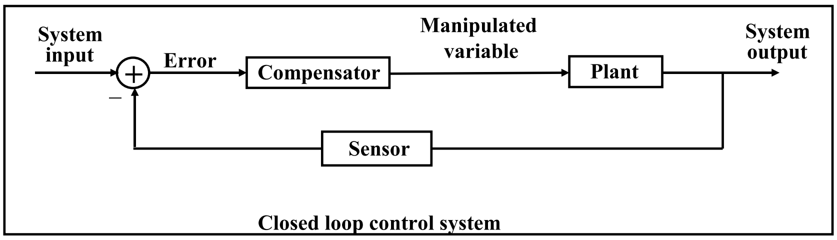


Fig. 3:

Two types of control systems:

- A **regulator** maintains a physical variable at some constant value in the presence of perturbances.
- A **servo mechanism** describes a control system in which a physical variable is required to follow, or track some desired time function (originally applied in order to control a mechanical position or motion).

2 Examples For Control Systems

In the following we will study 2 examples where feedback control is applied. The first example shows how to model an rf control system whereas the second and third example show electronic circuits which can be used to design the compensator (or controller) filter.

2.1 RF Control System

RF control system are used to stabilize amplitude and phase of the accelerating fields. Amplitude and phase are measured, compared to the desired setpoints, and the amplified error signals drive actuators for amplitude and phase thereby forming a negative feedback loop.

Goal:
Maintain stable gradient and phase

Solution:: Feedback for gradient amplitude and phase.

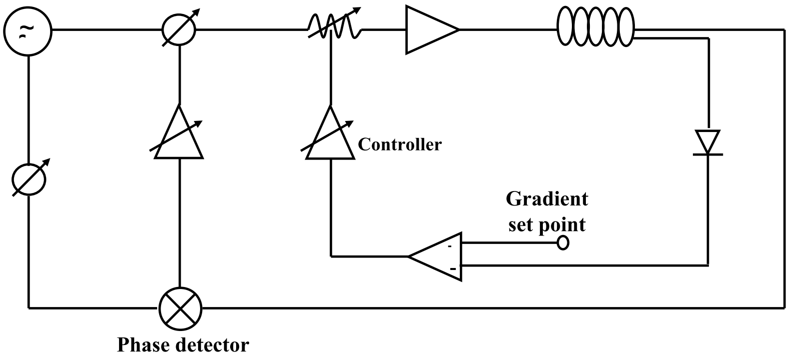


Fig. 4:

Model: Mathematical description if input-output relation of components combined with block diagram.

Amplitude loop (general form):

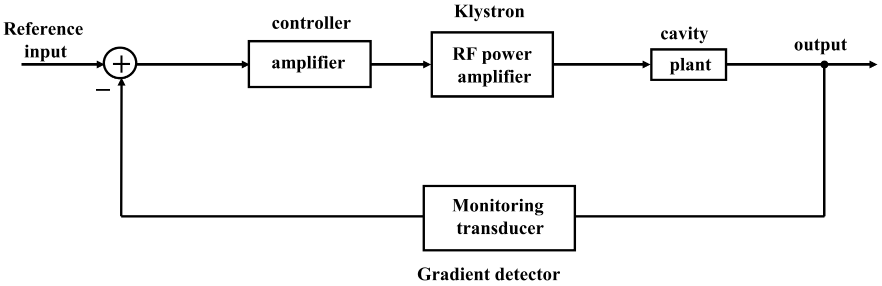


Fig. 5:

RF control model using *"transfer functions"*
A transfer function of a linear system is defined as the ratio of the Laplace Transform of the output and the Laplace transform of the input with I.C.'s=zero.

Input-Output relations

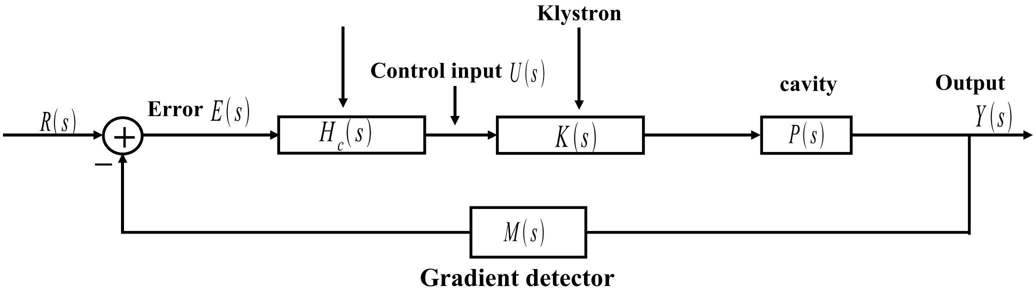


Fig. 6:

$Input$	$Output$	$TransferFunction$
$U(s)$	$Y(s)$	$G(s) = P(s)K(s)$
$E(s)$	$Y(s)$	$L(s) = G(s)H_c(s)$
$R(s)$	$Y(s)$	$T(s) = (1+L(S)M(s))^{-1}L(s)$

2.2 Passive network circuit

For high frequencies the network shown in (Fig. 7.) forms a voltage divider determined by the ratio of the resistors while at low frequencies the input voltage is unchanged at the output.

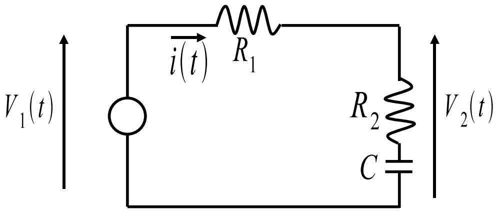


Fig. 7:

Differential Equations:

$R_1 i(t) + R_2 i(t) + \frac{1}{C}\int_0^t i(\tau)d\tau = v_1(t)$

$R_2 i(t) + \frac{1}{C}\int_0^t i(\tau)d\tau = v_2(t)$

Laplace Transform:

$R_1 I(s) + R_2 I(s) + \frac{1}{sC} I(s) = V_1(s)$

$R_2 I(s) + \frac{1}{s.C} I(s) = V_2(s)$

Laplace Transform:

$G(s) = \frac{V_2(s)}{V_1(s)} = \frac{R_2 Cs+1}{(R_1+R_2)Cs+1}$

Input V_1, Output V_2

2.3 Operational amplifier circuit

The transfer function of the circuit shown in (Fig. 8.) represents an amplifier with very high gain at low frequencies and a gain determined by the ratio of the resistors at high frequencies. The more general case is shown in (Fig. 9.) and allows the realization of a wide range of transfer functions.

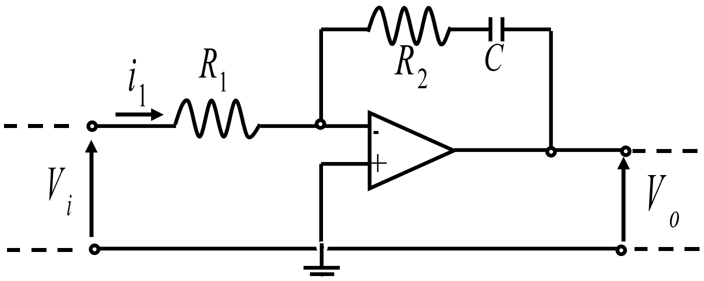


Fig. 8:

$V_i(s) = R_1 I_1(s)$ and $V_o(s) = -(R_2 + \frac{1}{sC})I_1(s)$

$G(s) = \frac{V_0(s)}{V_i(s)} = -\frac{R_2Cs+1}{R_1Cs}$

It is convenient to derive a transfer function for a circuit with a single operational amplifier that contains input and feedback impedance:

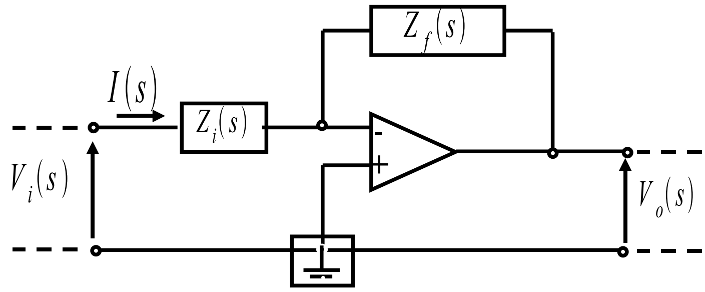


Fig. 9:

$V_i(s) = Z_i(s)I(s)$ and $V_0(s) = -Z_f(s)I(s) \rightarrow G(s) = \frac{V_0(s)}{V_i(s)} = -\frac{Z_f(s)}{Z_i(s)}$

3 Model of Dynamic Systems

Models of dynamic systems can be described by differential equations. A system of order n can be reduced to a set of n first order equations. These can be written in a matrix formalism called state space representations.
We will study the following dynamic system:
Parameters: k : Spring constant, γ : Damping constant, $u(t)$: Force
Quantity of interest: $y(t)$: Displacement from equilibrium

Differential equation: Newton's third law (m=1)

$\ddot{y}(t) = \sum F_{ext} = -ky(t) - \gamma\dot{y}(t) + u(t)$

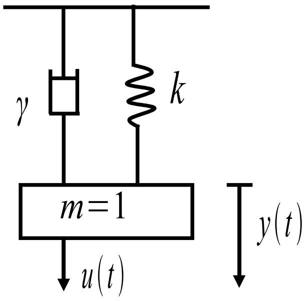


Fig. 10:

$\dot{y}+\gamma\dot{y}(t)+ky(t)=u(t)$

$y(0)=y_0,\dot{y}(0)=\dot{y}_0$

1. Equation is linear (i.e no y^2 like terms)
2. Ordinary (as opposed to partial e.g $\frac{\partial}{\partial x}\frac{\partial}{\partial t}f(x,t)=0$)
3. All coefficients constant: $k(t)=k,\gamma(t)=\gamma$ for all t

Stop Calculating, let's paint!!!

Picture to visualize differential equation.

1. Express highest order term (put it to one side)

   $\ddot{y}(t)=-ky(t)-\gamma\dot{y}(t)+u(t)$

2. Putt adder in front

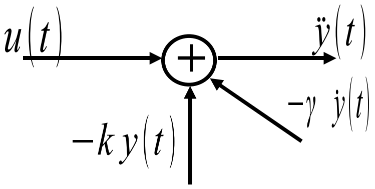


Fig. 11:

3. Synthesize all other terms using integrators!

3.1 Linear Ordinary Differential Equation (LODE)

General form of LODE:

$y^{(n)}(t)+a_{n-1}y^{(n-1)}(t)+....+a_1\dot{y}(t)+a_0y(t)=b_mu^{(m)}(t)+....+b_1\dot{u}(t)+b_0u(t)$
m,n positive integers, $m\leq n$; coefficients $a_0,a_1,...,a_{n-1},b_0,...b_m$ real numbers

Mathematical solution:Hopefully you know it

Solution of LODE: $y(t)=y_h(t)+y_p(t)$,

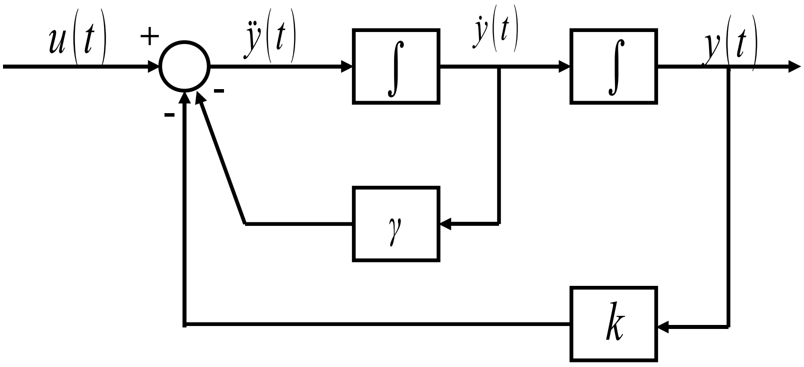


Fig. 12: Block diagram.

sum of homogeneous solution $y_h(t)$ (natural response) solving

$y^{(n)}(t) + a_{n-1}y^{(n-1)}(t) + ... + a_1\dot{y}(t) + a_0 y(t) = 0$

And particular sloution $y_p(t)$

How to get natural response $y_h(t)$? Characteristic polynomial

$x(\lambda) = \lambda_n + a_{n-1}\lambda_{n-1} + a_1\lambda + a_0 = 0$

$(\lambda - \lambda_1)^r.(\lambda - \lambda_{r+1}...(\lambda - \lambda_n) = 0$

$y_h(t) = (c_1 + c_2 t + ... + c_r t^{r-1})e^{\lambda_1 t} + c_{r+1}e^{\lambda_{r+1}t} ... + c_n e^{\lambda_n t}$

Determination of $y_p(t)$ relatively simple, if input $u(t)$ yields only a finite number of independent derivates.E.g:$u(t) \cong e^{xit}, \beta_r t^r$.

Most important for control system/feedback design:
$y_{(n-1)}(t) + a_{n-1}y^{(n-1)}(t) + ... + a_1\dot{y}(t) + a_0 y(t) = b_m u^{(m)}(t) + ... + b_1\dot{u}(t) + b_0 u(t)$

In general: given any linear time invariant system described by LODE can be realized/simulated/easily visualized in a block diagram $(n = 2, m = 2)$

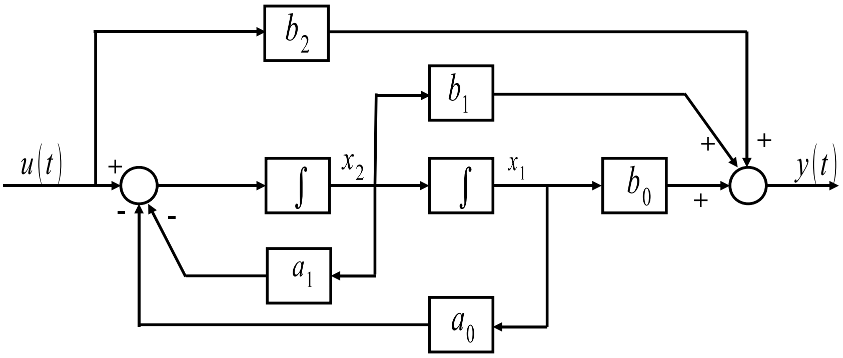


Fig. 13: Control-Canonical form

Very useful to visualize *interaction* between variables!
What are $x_1 and x_2$???

More explanations later, for now: Please simply accept it!

3.2 State Space Equation

Any system which can be presented by LODE can be represented in *state space form* (matrix differential equation).

What to do???

Let's go back to our first example (Newton's law):

$\ddot{y}(t) + \gamma\dot{y}(t) + ky(t) = u(t)$

1.STEP: Deduce set off first order differential equation in variables

$x_j(t)$ (so-called states of system)

$x_1(t) \cong Position : y(t)$

$x_2(t) \cong Velocity : \dot{y}(t) :$

$\dot{x_1}(t) = \dot{y}(t) = x_2(t)$

$\dot{x_2}(t) = \ddot{y}(t) = -ky(t) - \gamma\dot{y}(t) + u(t) = -kx_1(t) - \gamma x_2(t) + u(t)$

One LODE of order n transformed into n LODEs of order 1

2.STEP: Put everything together in a matrix differential equation:

$$\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k & -\gamma \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$\dot{x}(t) = Ax(t) + Bu(t)$ State equation

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

$y(t) = Cx(t) + Du(t)$ Measurement equation

Definition: The *system state x* of a system at any time t_0 is the "amount of information"that, together with all inputs for $t \geq t_0$,uniquely determines the behavior of the system for all $t \geq t_0$

The linear time-invariant (LTI) analog system is described via

Standard form of the State Space Equation

$\dot{x}(t) = Ax(t) + Bu(t)$ State equation

$y(t) = Cx(t) + Du(t)$ State equation

Where $\dot{x}(t)$ is the time derivative of the vector $x(t) = \begin{bmatrix} x_1(t) \\ ... \\ x_2(t) \end{bmatrix}$

System completely described by state space matrixes A,B,C,D (in the most cases $D = 0$)

Declaration of variables

Variable	Dimension	Name
$X(t)$	$nx1$	State vector
A	nXn	System matrix
B	nXr	Input matrix
$u(t)$	$rX1$	Input vector
$y(t)$	$pX1$	Output vector
C	pXn	Output matrix
D	pXr	Matrix representing direct coupling between input and output

Why all this work with state space equation? why bother with?

BECAUSE: Given any system of the LODE form

$$y^{(n)}(t) + a_{n-1}y^{(n-1)}(t) + ... + a_1\dot{y}(T) + a_0 y(t) = b_m u^{(m)}(t) + ... + b_1\dot{u}(t) + b_0 u(t)$$

Can be represented as

$$\dot{x}(t) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t) + Du(t)$$

With e.g. **Control-Canonical Form** (case $n = 3, m = 3$):

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, C = \begin{bmatrix} b_0 & b_1 & b_2 \end{bmatrix}, D = b_3$$

or **Observer-Canonical Form:**

$$A = \begin{bmatrix} 0 & 0 & -a_0 \\ 1 & 0 & -a_1 \\ 0 & 1 & -a_2 \end{bmatrix}, B = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix}, C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}, D = b_3$$

1. Notation is very compact,but not unique!!!
2. Computers love state space equation! (Trust us!)
3. Modern control ($1960-$now) uses state space equation
4. General (vector) block diagram for easy visualization

3.2.1 Block diagrams:

Now: Solution of state space equation in the time domain,out of that

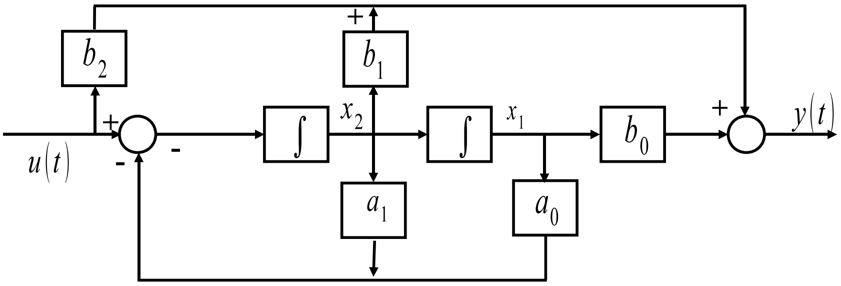


Fig. 14: Control-Canonical Form

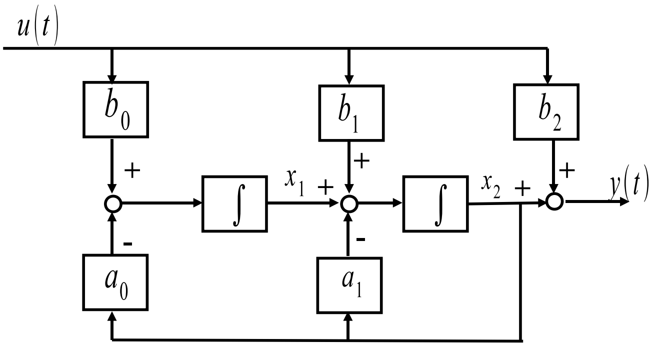


Fig. 15: Observer-Canonical Form

$x(t) = \phi(t)x(0) + \int_0^t \phi(\tau)Bu(t-\tau)d\tau$

Natural Response +Particular solution

$y(t) = Cx(t) + Du(t)$

$= C\phi(t)x(0) + C\int_0^t \phi(\tau)Bu(t-\tau)d\tau + Du(t)$

With the **state transition matrix**

$\phi(t) = I + At + \frac{A^2}{2!}t^2 + \frac{A^3}{3!}t^3 + ... = e^{At}$

Exponential series in the matrix A(time evaluation operator) properties of $\phi(t)$ (state transition matrix).

1. $\frac{d\phi(t)}{dt} = A\phi(t)$
2. $\phi(0) = I$
3. $\phi(t_1 + t_2) = \phi(t_1).\phi(t_2)$
4. $\phi^{-1} = \phi(-t)$

Example:

$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \Longrightarrow A^2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \phi(t) = I + At = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} = e^{At}$

Matrix A is a nilpotent matrix.

3.2.2 *Example:*1

It is given the following differential equation:

$$\frac{d^2}{dt^2}y(t) + 4\frac{d}{dt}y(t) + 3u(t) = 2u(t)$$

1. State equations of differential equation:

   Let $x_1(t) = y(t)$ and $x_2(t) = \dot{y}(t)$ it is:

   $\dot{x_1}(t) = \dot{y}(t) = x_2(t)$

   $\dot{x_2}(t) + 4x_2(t) + 3x_1(t) = 2u(t)$

   $\dot{x_2} = -3x_1(t) - 4x_2(t) + 2u(t)$

2. Write the state equations in matrix form:

   Define system state $x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -3 & -4 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 2 \end{bmatrix} u(t)$
   $y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t)$

3.2.3 *Cavity Model*

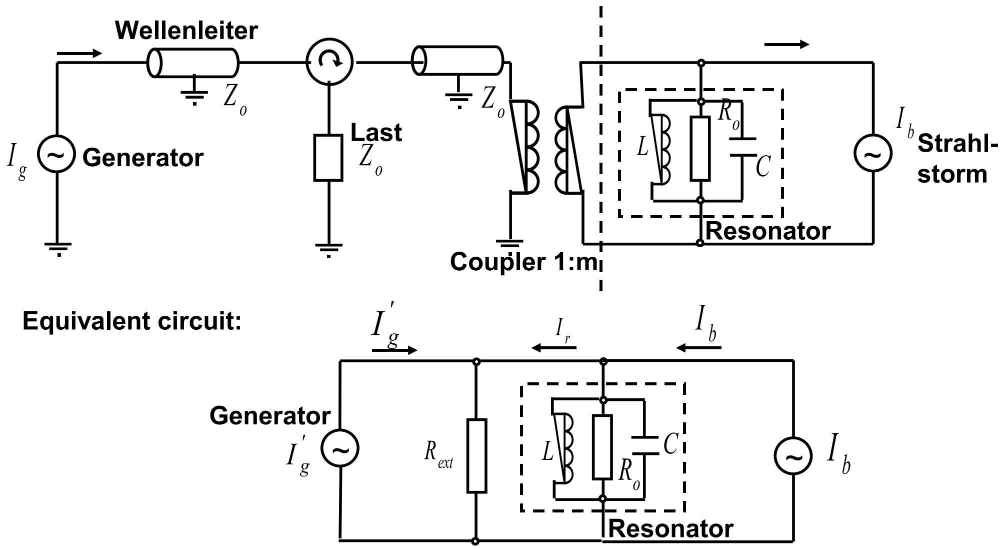


Fig. 16:

$$C.\ddot{U} + \frac{1}{R_L}.\dot{U} + \frac{1}{L}.U = \dot{I}'_g + \dot{I}_b$$

$$\omega_{\frac{1}{2}} := \frac{1}{2R_LC} = \frac{\omega_0}{2Q_L}$$

$\ddot{U} + 2\omega_{\frac{1}{2}}.\dot{U} + \omega_0^2.U = 2R_L\omega_{\frac{1}{2}}.(\frac{2}{m}\dot{I}_g + \dot{I}_b)$ Only envelope of rf (real and imaginary part) is of interest:

$$U(t) = (U_r(t) + iU_i(t)).exp(i\omega_{HF}(t)$$

$I_g(t) = (I_g r(t) + iI_{gi}(t)).exp(i\omega_{HF}(t))$

$I_b(t) = (I_{b\omega r}(t) + iI_{b\omega i}(t)).exp(i\omega_{HF}t) = 2(I_{bor}(t) + iI_{boi}(t)).exp(i\omega_{HF}t)$

Neglect small terms in derivatives for U and I

$\ddot{U}_r + i\ddot{U}_i(t) << \omega_{HF}^2(U_r(t) + iU_i(t))$

$2\omega_{\frac{1}{2}}(\dot{U}_r + i\dot{U}_r(t)) << \omega_{HF}^2(U_r(t) + iU_i(t))$

$\int_{t_1}^{t_2}(\dot{I}_r(t) + i\dot{I}_i(t))dt << \int_{t_1}^{t_2}\omega_{HF}(I_r(t) + iI_i(t))dt$

Envelope equations for real and imaginary component.

$\dot{U}_r(t) + \omega_{\frac{1}{2}}.U_r + \Delta\omega.U_i = \omega_{HF}\frac{r}{Q}.(\frac{1}{m}I_{gr} + I_{bor})$

$\dot{U}_i(t) + \omega_{\frac{1}{2}}.U_i - \Delta\omega.U_r = \omega_{HF}(\frac{r}{Q}).(\frac{1}{m}I_{gi} + I_{boi})$

Matrix equations:

$$\begin{bmatrix} \dot{U}_r(t) \\ \dot{U}_i(t) \end{bmatrix} = \begin{bmatrix} -\omega_{\frac{1}{2}} & -\Delta\omega \\ \Delta\omega & -\omega_{\frac{1}{2}} \end{bmatrix} . \begin{bmatrix} U_r(t) \\ U_i(t) \end{bmatrix} + \omega_{HF}(\frac{r}{Q}). \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} . \begin{bmatrix} \frac{1}{m}I_{gr}(t) + I_{bor}(t) \\ \frac{1}{m}I_{gi}(t) + I_{boi}(t) \end{bmatrix}$$

With system Matrices: $A = \begin{bmatrix} -\omega_{\frac{1}{2}} & -\Delta\omega \\ \Delta\omega & -\omega_{\frac{1}{2}} \end{bmatrix}$ $B = \omega_{HF}(\frac{r}{Q}). \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

$\overrightarrow{x}(t) = \begin{bmatrix} U_r(t) \\ U_i(t) \end{bmatrix}$ $\overrightarrow{u}(t) = \begin{bmatrix} \frac{1}{m}I_{gr}(t) + I_{bor}(t) \\ \frac{1}{m}I_{gi}(t) + I_{boi}(t) \end{bmatrix}$

General form: $\dot{\overrightarrow{x}} = A.\vec{x}(t) + B.\overrightarrow{u}(t)$

Solution:

$\overrightarrow{x}(t) = \phi(t).\overrightarrow{x}(0) + \int_0^t \phi(t - t').B.\overrightarrow{u}(t')dt'$

$\phi(t) = e^{-\omega_{\frac{1}{2}}t} \begin{bmatrix} cos(\Delta\omega t) & -sin(\Delta\omega t) \\ sin(\Delta\omega t) & cos(\Delta\omega t) \end{bmatrix}$

Special case:

$\overrightarrow{u}(t) = \begin{bmatrix} \frac{1}{m}I_{gr}(t) + I_{bor}(t) \\ \frac{1}{m}I_{gi}(t) + I_{boi}(t) \end{bmatrix} =: \begin{bmatrix} I_r \\ I_i \end{bmatrix}$

$$\begin{bmatrix} U_r(t) \\ U_i(t) \end{bmatrix} = \frac{\omega_{HF}(\frac{r}{Q})}{\omega_{\frac{1}{2}}^2 + \Delta\omega^2} . \begin{bmatrix} \omega_{\frac{1}{2}} & -\Delta\omega \\ \Delta\omega & \omega_{\frac{1}{2}} \end{bmatrix} .\{1 - \begin{bmatrix} cos(\Delta\omega t) & -sin(\Delta\omega t) \\ sin(\Delta\omega t) & cos(\Delta\omega t) \end{bmatrix} e^{-\omega_{\frac{1}{2}}t}\}. \begin{bmatrix} I_r \\ I_i \end{bmatrix}$$

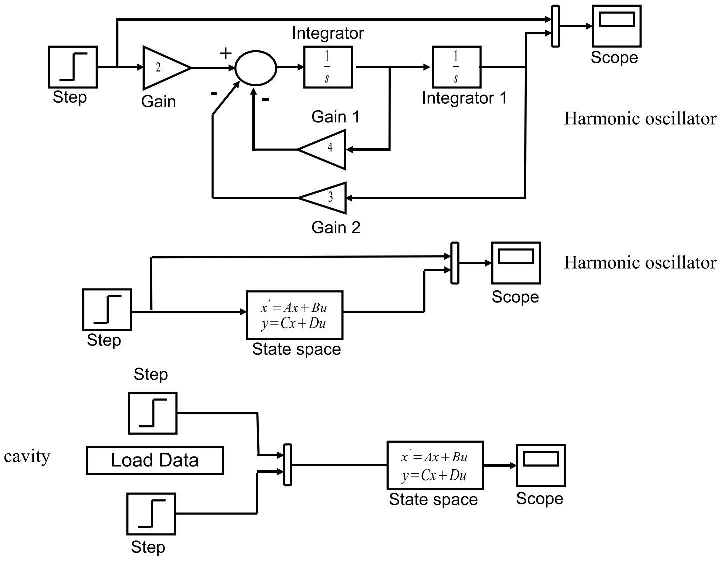


Fig. 17:

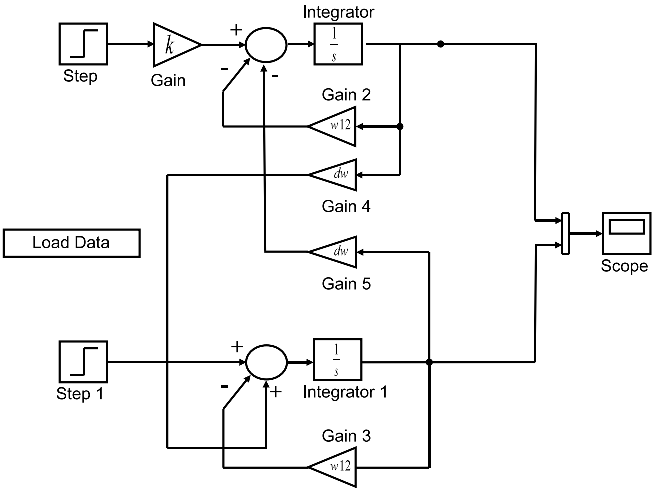


Fig. 18:

3.3 Mason Rule

Mason's rule is a simple formula for reducing block diagrams.Works on continuous and discrete.In its most general form it is messy,but For special case when all path touch

$H(s) = \frac{\Sigma(\text{forward path gains})}{1-\Sigma(\text{loop path gains})}$

Two paths are said to *touch* if they have a component in common,e.g. an adder

Forward path:$F_1 : 1-10-11-5-6$ and $F_2 : 1-2-3-4-5-6$

Loop path: $I_1 : 3-4-5-8-9$ and $I_2 : 5-6-7$

$G(f_1) = H_5H_3, G(f_2) = H_1H_2H_3, G(I_1) = H_2H_4, G(I_2) = H_3$

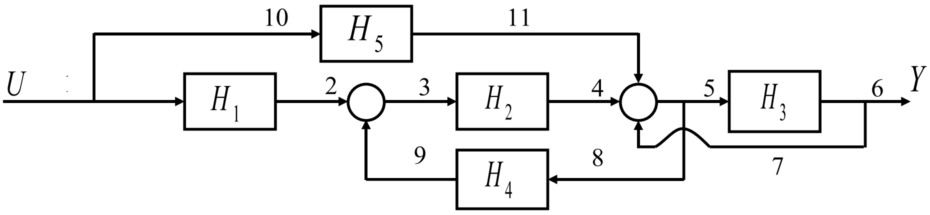


Fig. 19:

Check:All path touch (contain adder between 4 and 5)

$\Longrightarrow$ By mason's rule: $H = \frac{G(f_1)+G(f_2)}{1-G(l_1)-G(l_2)} = \frac{H_5H_3+H_1H_2H_3}{1-H_2H_4-H_3} = \frac{H_3(H_5+H_1H_2)}{1-H_2H_4-H_3}$

4 Transfer Function $G(s)$

The transfer function is commonly used in the analysis of single-input single-output electronic filters, for instance. It is mainly used in signal processing, communication theory, and control theory. The term is often used exclusively to refer to linear, time-invariant systems (LTI), as covered in this article. Most real systems have non-linear input/output characteristics, but many systems, when operated within nominal parameters (not "over-driven") have behavior that is close enough to linear that LTI system theory is an acceptable representation of the input/output behavior.
Continuous-time space model

$\dot{x}(t) = Ax(t) + Bu(t)$ State equation

$y(t) = Cx(t) + Du(t)$ Measurement equation

Transfer function describes input-output relation of system $sX(s) - x(0) = AX(s) + BU(s)$

$U(s)$
System
$Y(s)$

Fig. 20:

$X(s) = (sI - A)^{-1}x(0) + (sI - A)^{-1}BU(s)$

$= \phi(s)x(0) + \phi(s)BU(s)$

$Y(s) = CX(s) + DU(s)$

$= C[(sI - A)^{-1}]x(0) + [c(sI - A)^{-1}B + D]U(s)$

$= C\phi(s)x(0) + C\phi(s)BU(s) + DU(s)$

Transfer function $G(s)$ (pxr) (case:$X(0) = 0$):

$G(s) = C(sI - A)^{-1}B + D = C\phi(s)B + D$

Transfer function of TESLA cavity including $\frac{8}{9} - \pi$ mode

$H_{cont}(s) \approx H_{cav}(s) = H_{\Pi}(s) + H_{\frac{8\pi}{9}}(s)$

$\pi-$ mode $H_{\Pi}(s) = \frac{(\omega_{\frac{1}{2}})\pi}{\Delta\omega_{\pi}^{2}+(s+(\omega_{\frac{1}{2}})_{\pi})^{2}} \begin{bmatrix} s+(\omega_{\frac{1}{2}})_{\pi} & -\Delta\omega_{\pi} \\ -\Delta\omega_{\pi} & s+(\omega_{\frac{1}{2}})_{\pi} \end{bmatrix}$

$\frac{8}{9}\pi-$ mode $H_{\frac{8\pi}{9}}(s) = -\frac{(\omega_{\frac{1}{2}})_{\frac{8\pi}{9}}}{\Delta\omega_{\frac{8\pi}{9}}^{2}+(s+(\omega_{\frac{1}{2}})_{\frac{8\pi}{9}})^{2}} \begin{bmatrix} s+(\omega_{\frac{1}{2}})\frac{8\pi}{9} & -\Delta\omega_{\frac{8\pi}{9}} \\ \Delta\omega_{\frac{8\pi}{9}} & s+(\omega_{\frac{1}{2}})_{\frac{8\pi}{9}} \end{bmatrix}$

4.1 Transfer Function of a Closed Loop System

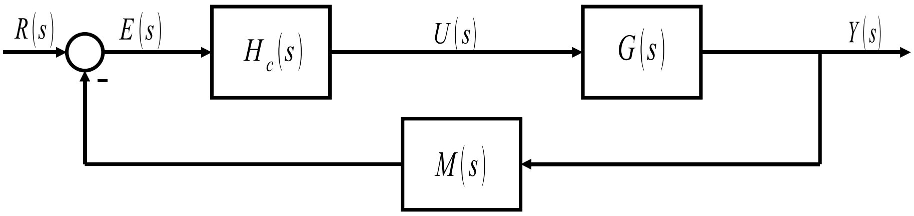


Fig. 21:

We can deduce for the output of the system

$Y(s) = G(s)U(s) = G(s)H_c(s)E(s)$

$= G(s)H_c(s)[R(s) - M(s)Y(s)$

$= L(s)R(s) - L(s)M(s)Y(s)$

With $L(s)$ the transfer function of the open loop system (controllerlus plant)

$(1 + L(s)M(s))Y(s) = L(s)R(s)$

$Y(s) = (I + L(s)M(s))^{-1}L(s)R(s)$

$= T(S)R(s)$ $T(s)$ is called: Reference Transfer Function

4.2 Sensitivity

The ratio of change in Transferfuncstion $T(s)$ by the parameter b can be defined as:

System characteristics change with system parameter variations $S = \frac{\Delta T(s)}{T(s)} \cdot \frac{b}{\Delta b}$

The sensitivity function is defined as:$S_b^T = lim_{\Delta b \longrightarrow 0} \frac{\Delta T(s)}{\Delta b} \cdot \frac{b}{T(s)} = \frac{T(s)}{b} \cdot \frac{b}{T(s)}$

Or in general sensitivity function of a characteristic W with respect to the parameter b:

$S_b^W = \frac{W}{b} \cdot \frac{b}{W}$

Example:Plant with propotional feedback given by $G_c(s) = K_p$ and $G_p(s) = \frac{K}{s+0.1}$

Plant transfer function $T(s) : T(s) \frac{K_p G_p(s)}{1+K_p G_p(s) H_k}$

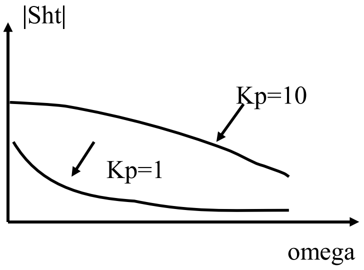


Fig. 22:

$S_H^T(j\omega) = \frac{-K_pG_p(j\omega)H_k}{1+K_pG_p(j\omega)H_k} = \frac{-0.25K_p}{0.1+0.25K_p+j\omega}$

Increase of H results in decrease of T. System can't be insensitive to both H,T

4.3 Disturbance Rejection

Disturbances are system influences we do not control and want to minimize its impact on the system.

$C(s) = \frac{G_c(s).G_p(s)}{1+G_c(s).G_p(s).H(s)}.R(s) + \frac{G_d(s)}{1+G_c(s).G_p(s).H(s)}$

$= T(s).R(s). + T_d(s).D(s)$

To reject disturbances, make $T.d(s).D(s)$ small!

1. Using frequency response approach to investigate disturbance rejection
2. In general $T_d(j\omega)$ can't be small for all $-\omega$ design $T_d(j\omega)$ small for significant portion of system bandwidth.
3. Reduce the Gain $G_d(j\omega)$ between distrubance input and output.
4. Increase the loop gain $G_cG_p(j\omega)$ without increasing the gain $G_d(j\omega)$.Usually accomplished by the compensator choice $G_c(j\omega)$

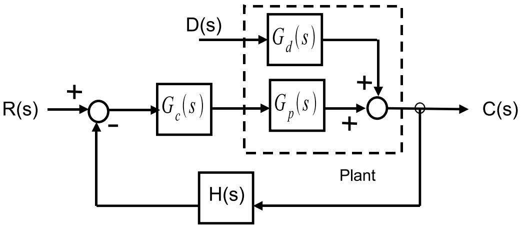


Fig. 23:

5. Reduce the distrubance magnitude $d(t)$ should always be attempted if reasonable
6. Use feed forward compensation,if disturbance can be measured.

5 Stability

In electrical engineering, specifically signal processing and control theory, BIBO Stability is a form of stability for signals and systems. BIBO stands for Bounded-Input Bounded-Output. If a system is BIBO stable then the output will be bounded for every input to the system that is bounded.

Now we have learnt so far:
The impulse response tells us everything about the system response to arbitrary input signal $u(t)$.

What we have not learnt:
If we know the transfer function $G(s)$, how can we deduce the system behavior?

What can we say about the system stability.

<u>**Definition:**</u> A linear time invariant system is called to be **BIBO** stable (Bounded-Input-Bounded-Output) for all bounded inputs $|u(t)| \leq M_1$ (for all t) exists a boundary for the output signal M_2, so that $|y(t)| \leq M_2$ (for all t) with M_1 and $|M_2|$ are positive real numbers.

Input never exceeds M_1 and output never exceeds M_2 then we have BIBO stability!

Note:It has to be valid for ALL bounded input signals!

Example: $Y(s) = G(s)U(s)$, integrator $G(s) = \frac{1}{s}$

1. Case $u(t) = \delta(t), U(s) = 1$

   $|y(t)| = |L^{-1}[Y(s)]| = |L^{-1}[\frac{1}{s}]| = 1$
2. Case $u(t) = 1, U(s) = \frac{1}{s}$

   $|y(t)| = |L^{-1}[Y(s)]| = |L^{-1}[\frac{1}{s^2}]| = t$

BIBO stability has to be shown/proved for any input.It is not sufficient to show its validity for a single input signal!

Condition for BIBO stability:

We start from the input-output relation

$Y(s) = G(s)U(s)$

By means of the convolution theorem we get

$|y(t)| = |\int_0^t g(\tau)u(t-\tau)d\tau| \leq \int_0^t |g(\tau)||u(t-\tau)|d\tau \leq M_1 \int_0^\infty |g(\tau)|d\tau \leq M_2$

Therefore it follows immediately:

If the impulse response is absolutely integrable

$\int_0^\infty |g(t)|dt < \infty$

Then the system is BIBO-stable

5.1 Poles and Zeroes

Can stability be determined if we know the TF of a system?

$G(s) = C\phi(s)B + D = C\frac{[sI-A]_{adj}}{\chi(s)}B + D$

Coefficients of Transfer function $G(s)$ are rational functions in the complex variable s

$g_{ij}(s) = a.\frac{\Pi_{k=1}^{m}(s-z_k)}{\pi_{l=1}^{n}(s-p_l)} = \frac{N_{ij}(s)}{D_{ij}(s)}$

Z_k Zeroes,P_l Poles,α real constant, and it is $m \leq n$ (we assume common factors have already been canceled!)

what do we know about the zeros and the poles?

Since numerator $N(s)$ and denominator $D(s)$ are polynomials with real coefficients, Poles and zeros must be real numbers or must arise as complex conjugated pairs!

5.2 Stability directly from state-space

Recall:$H(s) = C(sI-A)^{-1}B + D$

Assuming $D = 0$ (D could change zeros but not poles)

$H(s) = \frac{C_{adj}(sI-A)B}{det(sI-A)} = \frac{b(s)}{a(s)}$

Assuming there are no common factors between the poly $C_adj(sI-A)B$ and $det(sI-A)$ i.e no pole-Zero cancelations (usually true, system called "minimal") then we can identify
and

$b(s) = C_{adj}(sI-A)B$

$a(s) = det(sI-A)$

i.e poles are root of det$(sI-A)$

Let λ_i be the i_{th} eigen value of A if $\mathbf{Re}\lambda_i \leq 0$ for all $i \Longrightarrow$ System stable

So with computer,with eigenvalue solver, can determine system stability directly from coupling Matrix A.
A system is BIBO stable if, for every bounded input, the outut remains bounded with increasing time.

For a LTI system, this defination requires that all poles of the closed-loop transfer-function (all roots of the system characteristic equation) lie in the left half of the comlex plane.

5.3 Several methods are available for stability analyasis:

1. Routh Hurwitz criterion
2. Calculation of exact locations of roots
   - Root locus technique
   - Nyquist criterion
   - Bode plot
3. Simulation (only general procedures for nonlinear systems)

While the first criterion proofs whether a feedback system is stable or unstable, the second method also provides information about the setting time (damping term) Pole locations tell us about impulse response i.e.also stability:

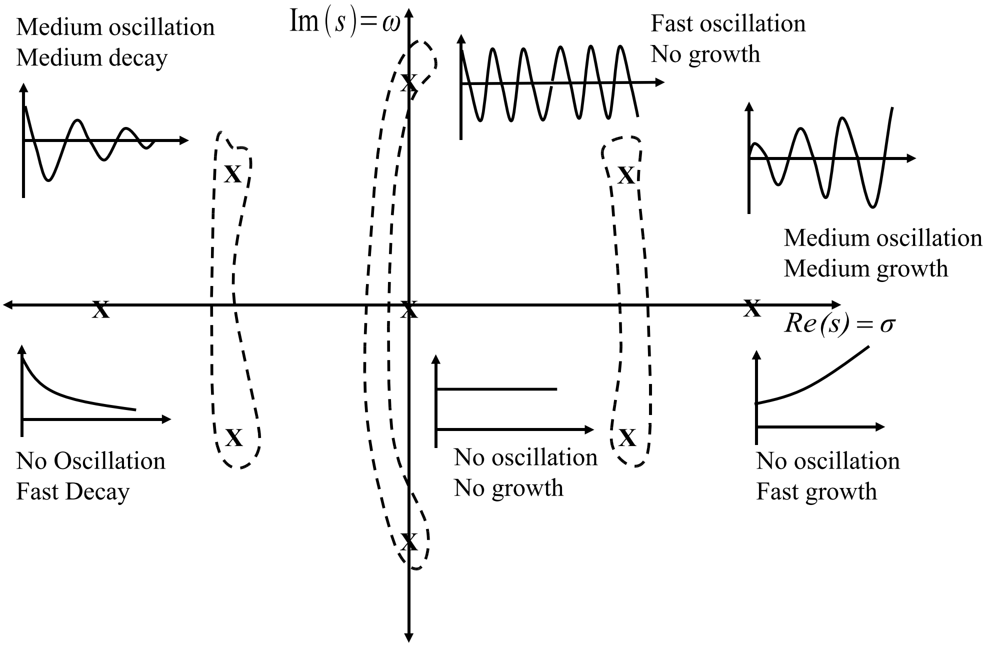


Fig. 24:

Furthermore:Keep in mind the following picture and facts!

1. Complex pole pair:Oscillation with growth to decay
2. Real pole:exponential growth or decay
3. Poles are the eigenvalues of the matrix A.
4. Position of zeros goes into the size of c_j....

In general a complex root must have a corresponding conjugate root $(N(s), D(s))$ polynomials with real coefficients.

Bode Diagram

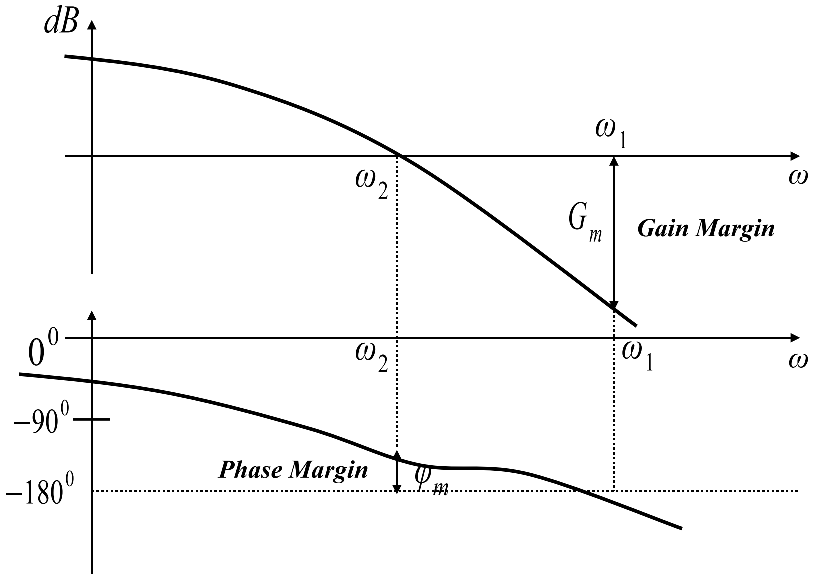
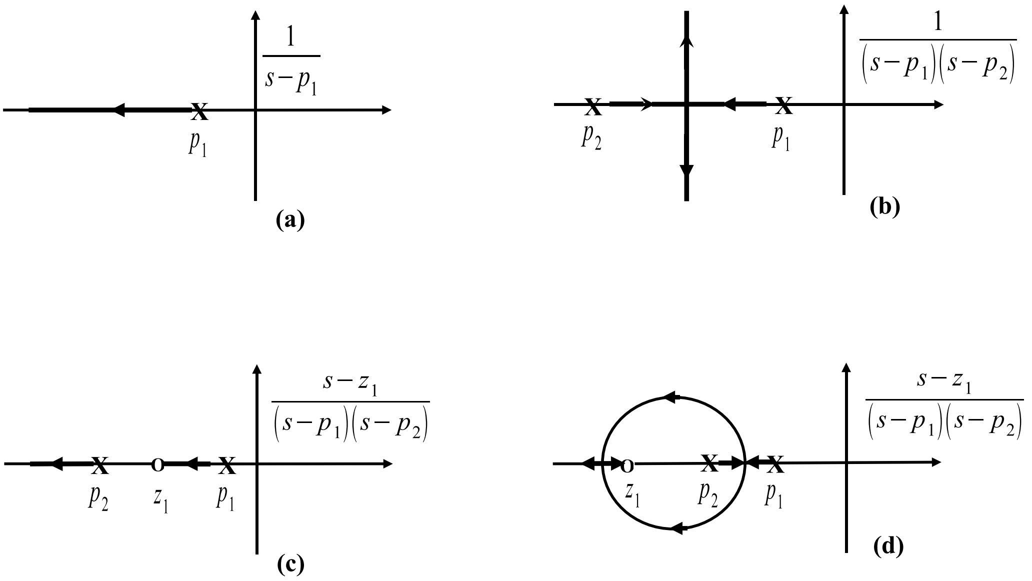


Fig. 25: Bode Diagram

The closed loop is stable if the phase of the unity crossover frequency of the *OPEN LOOP* is larger than -180 degrees.

Nyquist Criterion



$\frac{1}{s-p_1}$
p_1
(a)
$\frac{1}{(s-p_1)(s-p_2)}$
p_2
p_1
(b)
$\frac{s-z_1}{(s-p_1)(s-p_2)}$
p_2
z_1
p_1
(c)
$\frac{s-z_1}{(s-p_1)(s-p_2)}$
z_1
p_2
p_1
(d)

Fig. 26: Nyquist Criterion

5.4 Root Locus Analysis

Definition:A root locus of a system is a plot of the roots of the system characteristic equation (the poles of the closed-loop transfer function) while some parameter of the system (usually the feedback gain) is

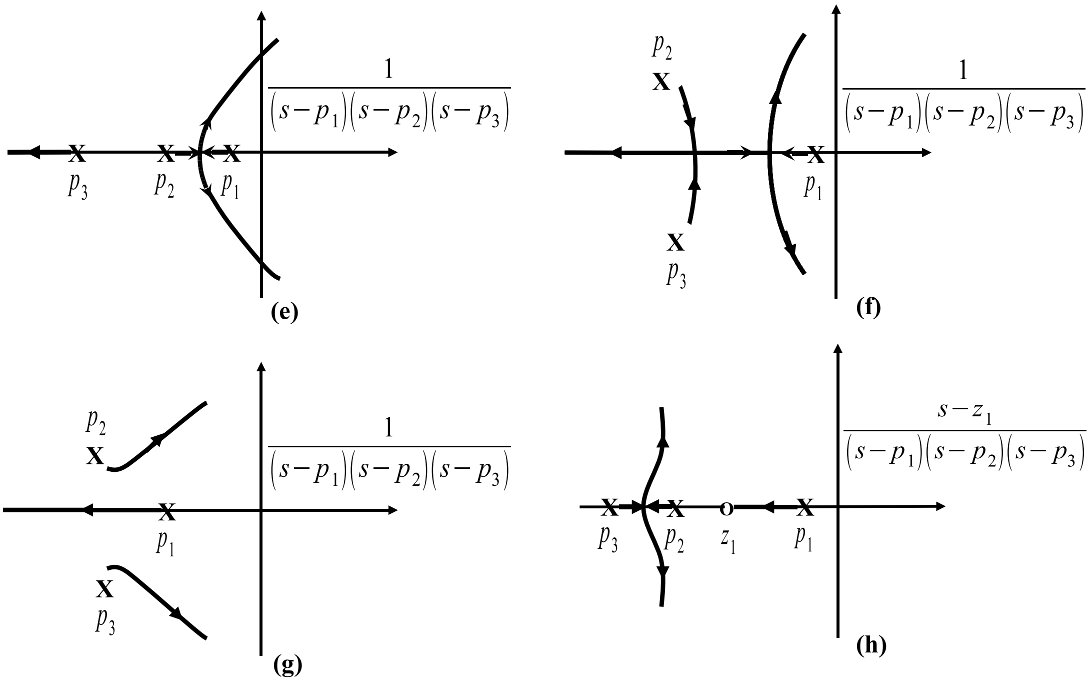


Fig. 27:

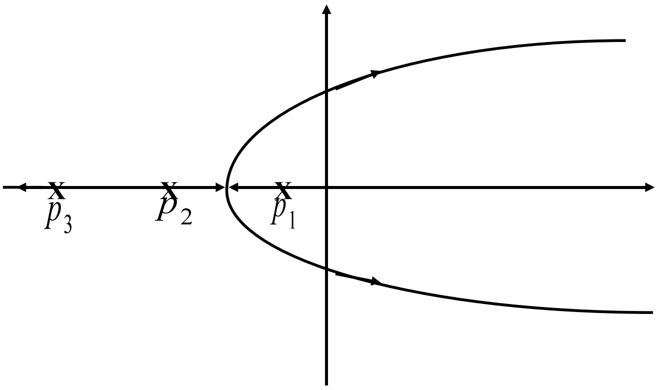


Fig. 28:

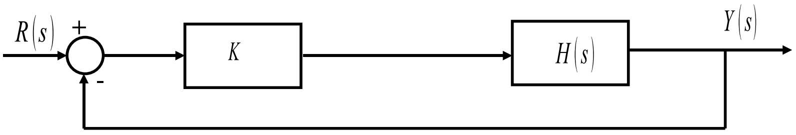


Fig. 29:

varied. $KH(s) = \frac{K}{(s-p_1)(s-p_2)(s-p_3)}$

$G_{cl}(s) = \frac{KH(s)}{1+KH(s)}$ roots at $1+KH(s) = 0$.

How do we move the poles by varying the constant gain constant gain K?

6 Feedback

Feedback is both a mechanism, process and signal that is looped back to control a system within itself. This loop is called the feedback loop. A control system usually has input and output to the system; when the output of the system is fed back into the system as part of its input, it is called the "feedback."
Feedback and regulation are self related. The negative feedback helps to maintain stability in a system in spite of external changes. It is related to homeostasis. Positive feedback amplifies possibilities of divergences (evolution, change of goals); it is the condition to change, evolution, growth; it gives the system the ability to access new points of equilibrium.
The idea:
Suppose we have a system or “plant"

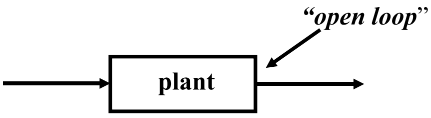


Fig. 30:

We want to improve some aspect of plant’s performance by observing the output and applying a appropriate "correction" signal.**<u>This is feedback</u>**

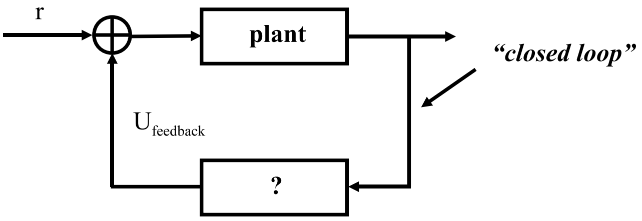


Fig. 31:

Question:What should this be ?

Open loop gain: $G^{O.L}(s) = G(s) = (\frac{u}{y})^{-1}$

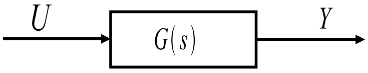


Fig. 32:

Closed loop gain: $G^{C.L}(s) = \frac{G(s)}{1+G(s)H(s)}$

Proof:$y = G(u - u_{fb}) = Gu - Gu_{fb} \Longrightarrow y + GH_y = Gu$

$= Gu - GH_y \Longrightarrow \frac{y}{u} = \frac{G}{(1+GH)}$

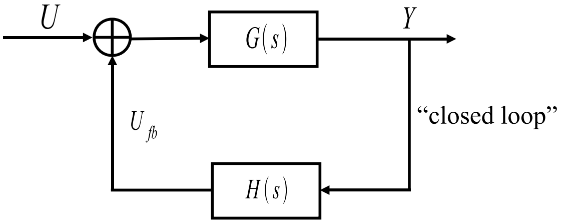


Fig. 33:

6.1 S.H.O

Consider S.H.O with feedback proportional to x i.e:

Where

$\ddot{x}+\gamma\dot{x}+\omega_n^2 x = u+u_{fb}$

$u_{fb}(t) = -\alpha x(t)$

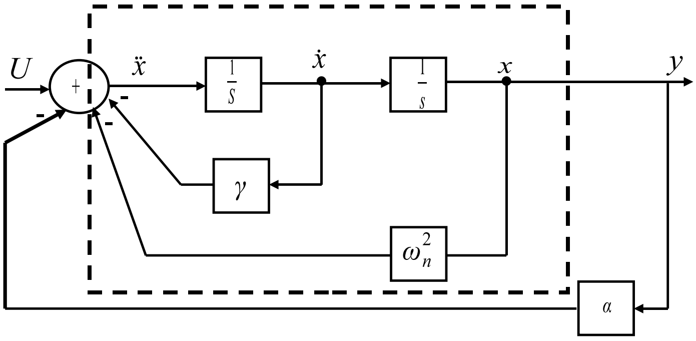


Fig. 34:

Then

$\ddot{x}+\gamma\dot{x}+\omega_n^2 x = u-\alpha x$

$\Longrightarrow \ddot{x}+\gamma\dot{x}+(\omega_n^2+\alpha)x = u$

Same as before,except that "natural" frequency $\omega_n^2+\alpha$

now the closed loop T.F.is: $G^{C.L.}(s) = \frac{1}{s^2+\gamma s+(\omega_n^2+\alpha)}$

So the effect of the proportional feedback in this case is *to increase the bandwidth of the system* (and reduce gain slightly, but this can easily be compensated by adding a constant gain in front..)

6.1.1 S.H.O with Integral Feed Back

In S.H.O suppose we use *integral feedback*

$u_{fb}(t) = -\alpha \int_0^t x(\tau)d\tau$

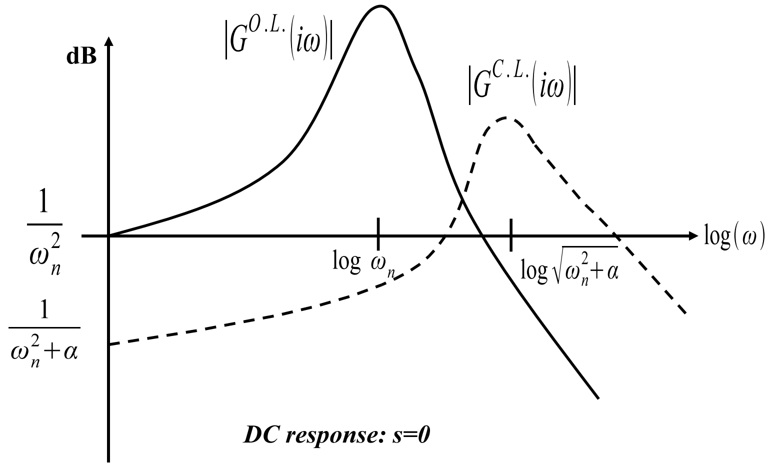


Fig. 35:

i.e $\ddot{x}+\gamma\dot{x}+\omega_n^2 x = u-\alpha\int_0^t x(\tau)d\tau$

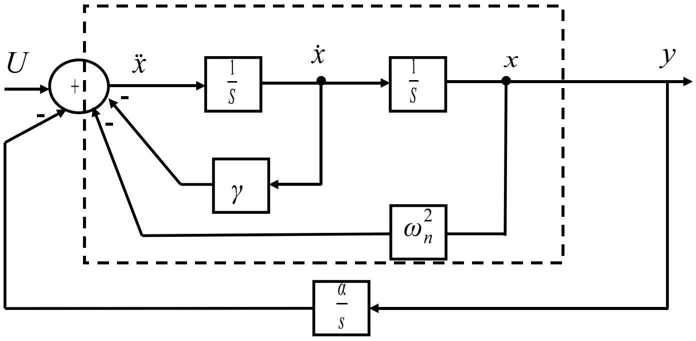


Fig. 36:

Differentiating once more yields: $\dddot{x}+\gamma\ddot{x}+\omega_n^2\dot{x}+\alpha x=\dot{u}$

No longer just simple S.H.O add another state $G^{C.L.}(s) = \frac{\frac{1}{s^2+\gamma s+\omega_n^2}}{1+(\frac{\alpha}{s})(\frac{1}{s^2+\gamma s+(\omega_n^2+\alpha)})}$

$= \frac{s}{s(s^2+\gamma s+\omega_n^2)+\alpha}$

Observe that

1. $G^{C.L}(0=0)$
2. For large s (and hence for large ω)

$G^{C.L}(s) \approx \frac{1}{(s^2+\gamma s+\omega_n^2} \approx G^{O.L.}(s)$

So integral feedback has killed DC gain i.e system *rejects constant* disturbances.

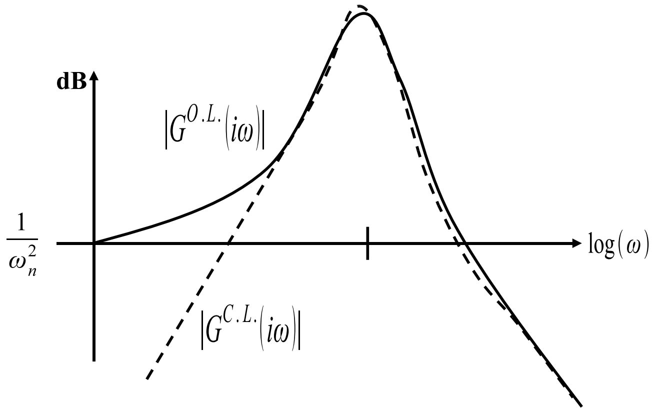


Fig. 37:

6.1.2 S.H.O with Differential Feedback

Suppose S.H.O now apply *differential feedback* i.e.

$u_{fb}(t) = -\alpha\dot{x}(t)$

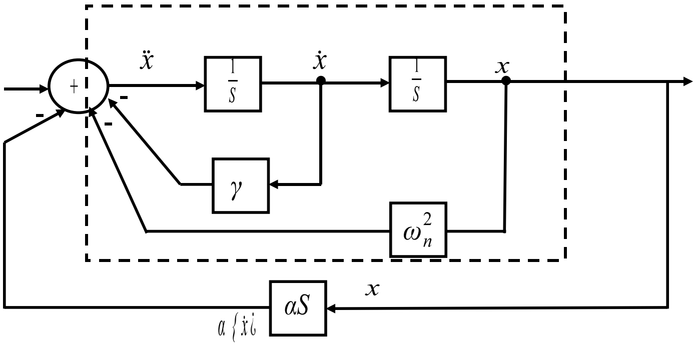


Fig. 38:

Now have $\ddot{x} + (\gamma+\alpha)\dot{x} + \omega_n^2 x = u$

So effect off differential feedback is to increase damping

Now $G^{C.L}(s) = \frac{1}{s^2+(\gamma+\alpha)s+\omega_n^2}$

So the effect of differential feedback here is to "flatten the resonance" i.e **damping is increased**

Note:Differentiators can never be built exactly,only approximately.

6.2 PID controller

1. The latter 3 examples of feedback can all be combined to form a P.I.D. Controller (prop.-integral-diff). $u_{fb} = u_p + u_d + u_I$
2. In example above S.H.O was a vey simple system and it was clear what physical interpretation of P.or I. or D.did.But for *large complex systems* not obvious

   $\Longrightarrow x$ **Require arbitrary "tweaking"**

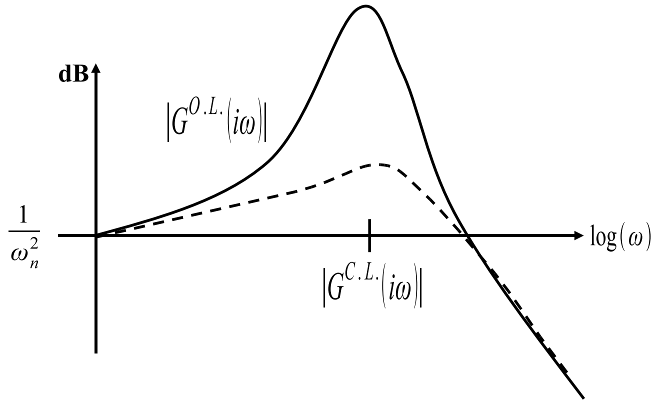


Fig. 39:

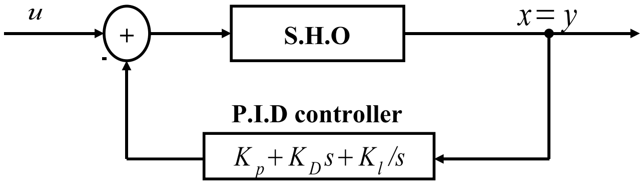


Fig. 40:

That's what we're trying to avoid.

For example, if you are so smart let's see you do this with your P.I.D. Controller:

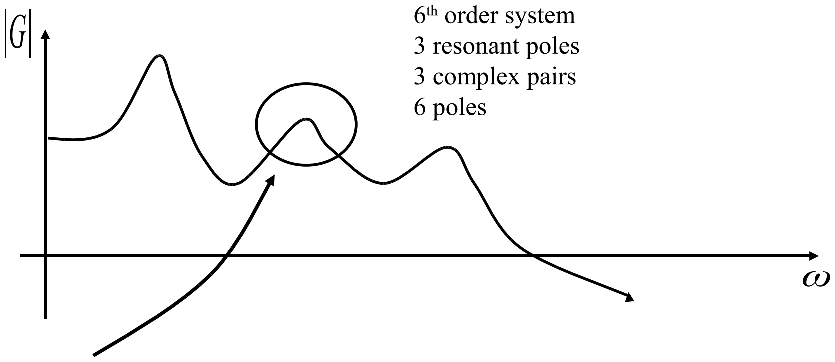


Fig. 41:

Damp this mode,but leave the other two modes undamped,just as they are.

This could turn out to be a tweaking nightmare get you no where fast!

we'll see how this problem can be solved easily.

6.3 Full State Feedback

Suppose we have a system

$\dot{x}(t) = Ax(t) + Bu(t)$

$y(t) = Cx(t)$

Since the state vector $x(t)$ contains all current information about the system the most general feedback makes use of **all** the state info.

$u - -K_1x_1 - ... - K_nx_n = -Kx$

Where $k = [k_1.....k_n]$ (Row matrix)

Where example:In S.H.O. examples

Proportional fbk:$u_p = -k_px = -\left[\begin{array}{cc} K_p & 0 \end{array}\right]$

Differential fbk:$u_D = -k_D\dot{x} = -\left[\begin{array}{cc} 0 & k_D \end{array}\right]$

Theorem:

If there are no poles cancellations in $G_{O.L}(s) = \frac{b(s)}{a(s)} = C(sI - A)^{-1}B$

Then can move the eigen values of $A - BK$ anywhere we want using full state feedback.

Proof: Given **any** system as L.O.D.E or state space it can be written as:

$$\begin{bmatrix} x_1 \\ ... \\ ... \\ x_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & ... & 0 \\ 0 & ... & ... & ... \\ 0 & ... & ... & 1 \\ -a_0 & ... & ... & -a_{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ ... \\ ... \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ .. \\ 1 \end{bmatrix} u$$

$$y = \left[\begin{array}{cccc} b_0 & ... & ... & b_{n-1} \end{array}\right] \begin{bmatrix} x_1 \\ ... \\ ... \\ x_n \end{bmatrix}$$

where $G^{O.L.} = C(sI - A)^{-1}B = \frac{b_{n-1}s^{n-1} + ... + b_0}{s^n + a_{n-1}s^{n-1} + ... + a_0}$ i.e first row of $A^{O.L}$ Gives the coefficients of the denominator

$a^{O.L}(s) = det(sI - A^{O.L.}) = s^n + a_{n-1}s^{n-1} + ... + a_0$

Now $A^{C.L.} = A^{O.L.} - BK = \begin{bmatrix} 0 & 1 & ... & 0 \\ 0 & ... & ... & 1 \\ 0 & ... & ... & 1 \\ -a_0 & ... & ... & -a_{n-1} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ ... \\ 1 \end{bmatrix} \left[\begin{array}{cccc} k_0 & ... & ... & K_{n-1} \end{array}\right] \left[\begin{array}{cccc} 0 & 1 & ... & \\ 0 & ... & ... & . \\ 0 & ... & ... & \\ -(a_0 + k_0) & ... & ... & -(a_{n-1} \end{array}\right.$

So closed loop denominator

$a^{C.L.}(s) = det(sI - A^{C.L})$

$= s^n + (a_0 + k_0)s^{n-1} + ... + (a_{n-1} + k_{n-1})$

Using $u = -Kx$ have direct control over every closed-loop denominator coefficient $\Longrightarrow$ Can place root anywhere we want in $s-$ plane.

Example:Detailed block diagram of S.H.O with full-scale feedback

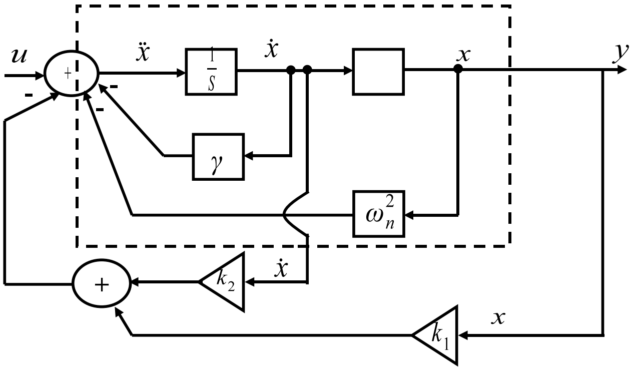


Fig. 42:

of course this **assumes** we have access to the $\dot{x}$ state,which we actually **Don’t** in practice.

However,let’s ignore that "minor" practical detail for now. (Kalman filter will show us how to get $\dot{x}$ form x)
With full state feedback have (assume D=0)

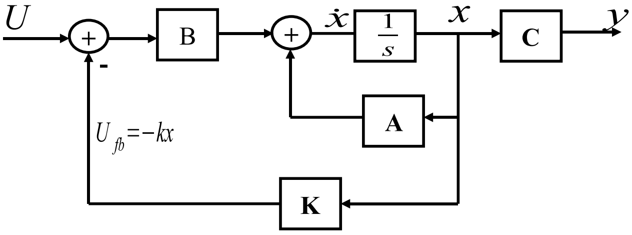


Fig. 43:

So

$\dot{x} = Ax + B[u + u_{fb}]$

$= Ax + Bu + BKu_{fb}$

$\dot{x} = (A - BK)x + Bu$

$u_{fb} = -Kx$

$y = Cx$

With full state feedback, get new closed loop matrix.

$A^{C.L} = (A^{O.L} - BK)$

Now all stability info is now given by the eigen values of new A matrix

The linear time- invariant system

$\dot{x} = Ax + Bu \; y = Cx$

Is said to be controllable if it is possible to find some input $u(t)$ that will transfer the initial state $x(0)$ to the origin of state-space,$x(t_0) = 0$, with t_0 finite.

The solution of the state equation is:

$x(t) = \phi(t)x(0) + \int_0^t \phi(\tau)Bu(t-\tau)d\tau$

For the system to be controllable, a function $u(t)$ must exist that satisfies the equation:

$0 = \phi(t_0)x(0) + \int_0^{t_0} \phi\tau Bu(t_0 - \tau)d\tau$

With t_0 finite. It can be shown that this condition is satisfied if the controllability matrix

$C_M = [B \;\; AB \;\; A^2B \;\; ....A^{n-1}B]$

Has inverse. This is equivalent to the matrix C_M having full rank (rank n for an n^{th} order differential equation.)
Observable:

- The linear time-invarient system is said to be observable if the initial conditions $x(0)$ can be determined from the output function $y(t), 0 \leq t \leq t_1$ where t_1 is finite with $y(t) = Cx = C\phi(t)x_0 + C\int_0^t \phi(\tau)Bu(t-\tau)d\tau$
- The system is observable if this equation can be solved for $x(0)$. It can be shown that the system is observable if the matrix: $O_M = \begin{bmatrix} C \\ CA \\ \dots \\ CA^{n-1} \end{bmatrix}$
- Has inverse. This is equivalent to the matrix C_M having full rank (rank n for an n-th order differential equation).

7 Discrete Systems

Digital control is a branch of control theory that uses digital computers to act as a system. Depending on the requirements, a digital control system can take the form of a microcontroller to an ASIC to a standard desktop computer. Since a digital computer is a discrete system the Laplace transform is replaced with the Z-transform. Also since a digital computer has finite precision (See quantization) extra care is needed to ensure the error in coefficients, A/D conversion, D/A conversion, etc. are not producing undesired or unplanned effects.
A discrete system or discrete-time system, as opposed to a continuous-time system, is one in which the signals are sampled periodically. It is usually used to connote an analog sampled system, rather than a digital sampled system, which uses quantized values.

Where do discrete systems arise?

Typical control engineering example: Assume the DAC+ADC are clocked at sampling period T.
Then $u(t)$ is given by: $u(k) \equiv u_c(t); kT \leq t < (k+1)T$

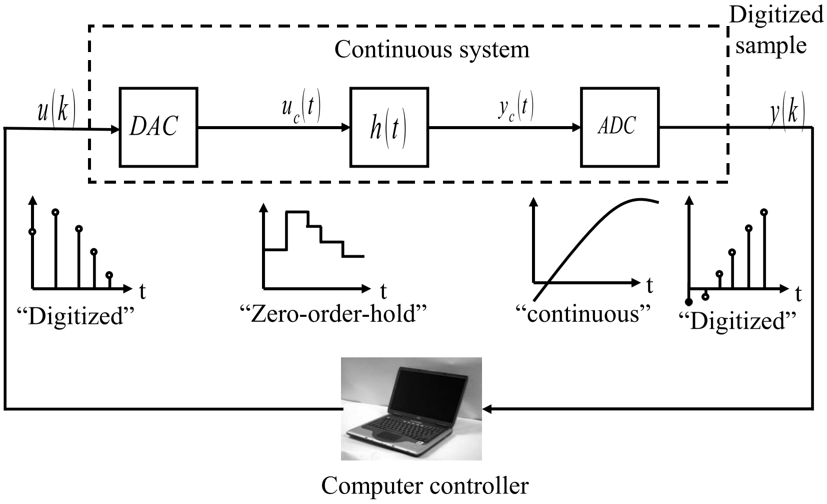


Fig. 44:

$y(k) \equiv y_c(kT); k = 0, 1, 2, ...$

Suppose: time continuous system is given by state-space $\dot{x_c}(t) = Ax_c(t) + Bu_c(t); x_c(0) = x_0$

$y_c(t) = Cx_c(t) + Du_c(t)$

7.1 State Space Description

Can we obtain direct relationship between $u(k)$ and $y(k)$? i.e want Equivalent discrete system:

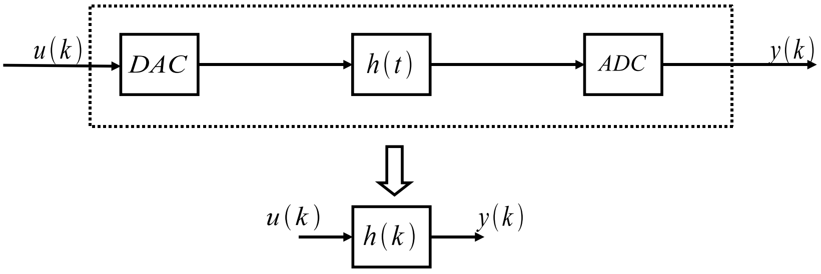


Fig. 45:

Yes! we can obtain equivalent discrete system.
Recall $\quad x_x(t) = e^{At}x_c(0) + \int_0^t e^{A\tau}.Bu_c(t-\tau)d\tau$

From this $\quad x_c(kT+T) = e^{AT}x_c(kT) + \int_0^T e^{A\tau}.Bu_c(kT-\tau)d\tau$

Observe that $\quad u(kT+T-\tau) = u(kT)$ for $\tau\varepsilon[0,T]$ i.e $u(kT+T-\tau)$ is constant $u(kT)$ over $\tau\varepsilon[0,T]$

can pull out of integral

$\Longrightarrow x_c(kT+T) = e^{At}x_c(kT) + (\int_0^t e^{A\tau}.Bd\tau)u_c(kT)$

$x(k+1) = A_d x(k) + B_d u(k)$

$y(k) = C_d x(k) + D_d u(k)$

$x(0) = x_c(0)$

So $A_d = e^{AT}$,$B_d = \int_0^T e^{A\tau}.Bd\tau BD_d = C, D_d = D$

So we have an exact (note: $x(k+1) = x(k) + \dot{x}(k)T + O(.)$) discrete time equivalent to the time continuous system at sample times $t = kT-$ no numerical approximation! A linear ordinary difference equation looks similar to a LODE

$y(k+n) + a_{n-1}y(k+n-1) + ... + a_1 y(k+1) + a_0 y(k) = b_m u(k+m) + ... + b_1 u(k+1) + b_0 u(k) n \geq m$;
Assumes initial values $yy(n-1), ...y(1), y(0) = 0$

7.2 Z-Transform

Z-Transform of the LODE yields (linearity of Z-transform):

$z_n Y(z) + z^{n-1} a_{n-1} Y(z) + ... + z a_1 y(z) + a_0 Y(z) = z_m b_m U(z) + .. + z b_1 U(z) + b_0 U(z)$

it follows the input-output relation:

$(z^n + z^{n-1} a_{n-1} + ... + z a_1 + a_0) Y(z) = (z^m b_m + ... + z b_1 + b_0) U(z)$

$Y(z) = \frac{z^m b_m + ... + z b_1 + b_0}{z^n + ... + z a_1 + a_0} U(z)$

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

Once again: if $U(z) = 1, (u(k) = \delta(k))$, then $Y(z) = G(z)$.

Transfer function of system is the Z-transform of its pulse response!
$x(k+1) = A_d x(k) + B_d u(k)$

$y(k) = Cx(k) + Du(k)$

Applying Z-transform on first equation:

$z.X(z) - zx(0) = A_d X(z) + B_d U(z)$

$(zI - A_d) X(z) = zx(0) + Bu(z)$

$X(z) = (zI - A)^{-1} zx(0) + (zI - A_d)^{-1} BU(z)$ Homogeneous solution

NOW: Particular solution

$Y(z) = CX(z) + DU(z) = C(zI - A_d)^{-1} zx(0) + (C(zI - A_d)^{-1} B + D) U(z)$

if $x(0) = 0$ then we get the input-output relation:

$Y(z) = G(z) U(z)$ with $G(z) = C(zI - A_d)^{-1} B + D$

Exactly like for the continuous systems!!!!!

For analysing discrete-time systems: Z transform (analog to Laplace Transform for time-continous system)
It converts linear ordinary difference equation into algebraic equations: easier to find a solution of the system!
It gives the frequency response for free!
Z-transform == generalized discrete-time Fourier transform

Given any sequence $f(k)$ the discrete-time Fourier tranform is $F(\tilde{\omega}) = \Sigma_{k=-\infty}^{\infty} f(k)e^{-i\omega k}$

$\omega = 2\Pi f, f = \frac{1}{T}$ The sampling frequency in Hz, T : difference/Time between two samples.

In the same spirit: $F(z) = Z[f(k)] = \Sigma_{k=0}^{\infty} f(k)z^{-k}$

with z a complex variable Note: if $f(k) = 0$ for $k = -1, -2, ...$ then $\tilde{F}(\omega) = F(z = e^{j\omega})$.

7.3 Stability

A discrete LTI system is BIBO stable if $|u(k)| < M, \forall k = |y(k)|; \forall k$

$|y(k)| = |\Sigma_0^k u(k-i)h(i)| \leq \Sigma_0^k |u(k-i)||h(i)| \leq M\Sigma_0^k |h(i)| \leq M\Sigma_0^{\infty} |h(i)|$
For L.O.D.E state space system: $H(z) = \alpha \frac{\pi_{i=1}(z-z_i)}{\pi_{i=1}^{n}(z-p_i)} = \Sigma_{i=1}^{k} \beta_i T_i(z)$

With partial fraction of the rational function: Once again pole locations tell a lot about shape of pulse response.
Zeros determine the size of β_i In general Complex pair → oscillatory growth/damping
real pole → exponential growth/decay but maybe oscillatory too (e.g: $r^n 1(n)$ where $r < 0$)

The farther inside unit circle are → The faster the damping → the higher stability i.e $|p_i| \leq 1$ → system stable Stability directly from state space: Exactly as for cts systems, assuming no pole-zero cancelations and $D = 0$

$H(z) = \frac{b(z)}{a(z)} = C(zI - A_d)^{-1}B_d = \frac{C_{adj}(zI - A_d)B_d}{det(zI - A_d)}$

$b(z) = C_{adj}(zI - A_d)B_d \quad a(z) = det(zI - A_d)$

→ Poles are eigenvalues of A_d So check stability, use eigenvalue solver to get e-values of the matrix A_d, then if $|\lambda_i| < 1$ for all $i \longrightarrow$ system stable. Where $|\lambda_i|$ is the i^{th} e-value of A_d.

7.4 Cavity Model

Converting the transfer function from the continuous cavity model to the discrete model:

$$H(s) = \frac{\omega_{12}}{\Delta\omega^2 + (s + \omega_{12})^2} \begin{bmatrix} s + \omega_{12} & -\Delta\omega \\ \Delta\omega & s + \omega_{12} \end{bmatrix}$$

The discretization of the model is represented by the z-transform:

$H(z) = (1 - \frac{1}{z})z(\frac{H(s)}{s}) = \frac{z-1}{z}.zL^{-1}\frac{H(s)}{s}|_{t=kT_s}$

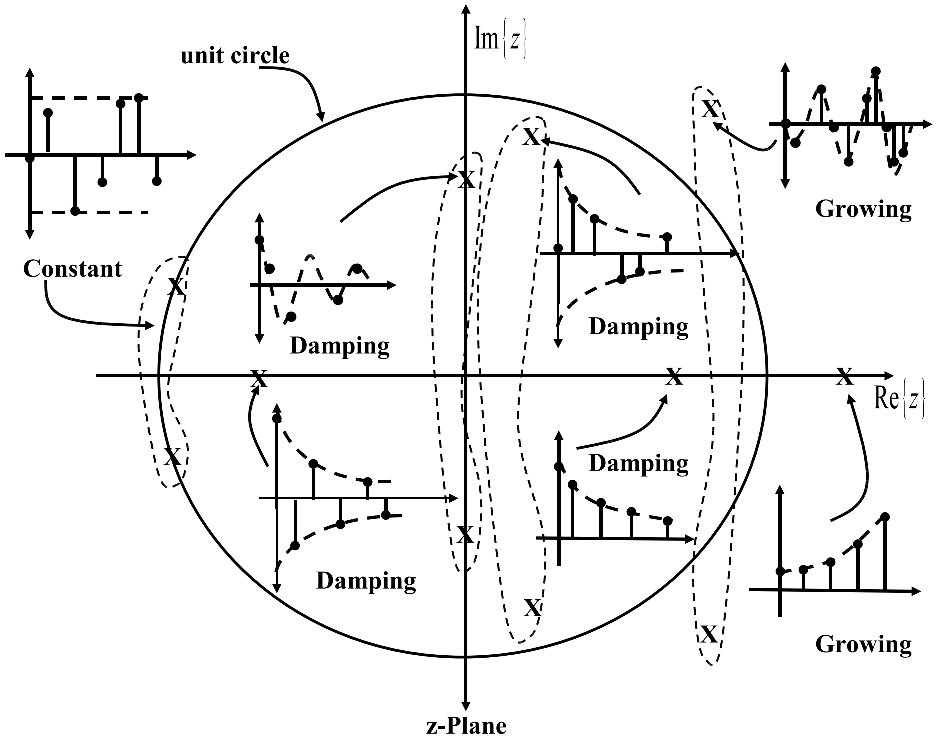


Fig. 46:

$$H(z) = \frac{\omega_{12}}{\Delta\omega^2+\omega_{12}^2}.\begin{bmatrix}\omega_{12} & -\Delta\omega \\ \Delta\omega & \omega_{12}\end{bmatrix} - (\frac{\omega_{12}}{\Delta\omega^2+\omega_{12}^2}.\frac{z-1}{z^2-2ze^{\omega_{12}T_s}.cos(\Delta\omega T)_s+e^{2\omega_{12}T_s}})$$

$$.((z-e^{\omega_{12}T_s}.cos(\Delta\omega T_s)).\begin{bmatrix}\omega_{12} & -\Delta\omega \\ \Delta\omega & \omega_{12}\end{bmatrix}) - e^{\omega_{12}T_s}.sin(\Delta\omega T_s).\begin{bmatrix}\Delta\omega & \omega_{12} \\ -\omega_{12} & \Delta\omega\end{bmatrix}$$

Given: $x(k+1) = Ax(k) + Bu(k)$

$z(k) = Cx(k)$ Assume $D = 0$ for simplicity

8 Optimal Control:

Optimal control deals with the problem of finding a control law for a given system such that a certain optimality criterion is achieved. A control problem includes a cost functional that is a function of state and control variables. An optimal control is a set of differential equations describing the paths of the control variables that minimize the cost functional.

Suppose the system is unstable or almost unstable. we want to find $u_{fb}(k)$ which will bring $x(k)$ to zero, quickly, from any initial condition.

A quadratic form is a quadratic function of the components of a vector:

$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ $\quad f(x) = f(x_1, x_2) = ax_1^2 + bx_1x_2 + cx_1 + dx_2^2$

$[x_1, x_2]\begin{bmatrix} a & \frac{1}{2}b \\ \frac{1}{2}b & d \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} c & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$f(x) = x^TQx$ Quadratic part $+P^Tx$ Linear part $+e$ Constant

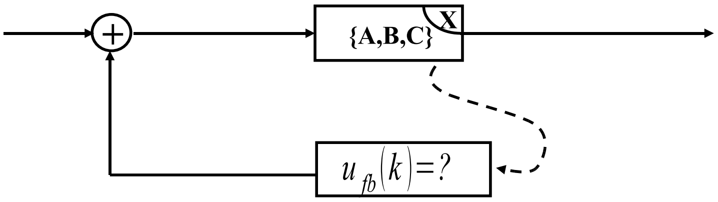


Fig. 47:

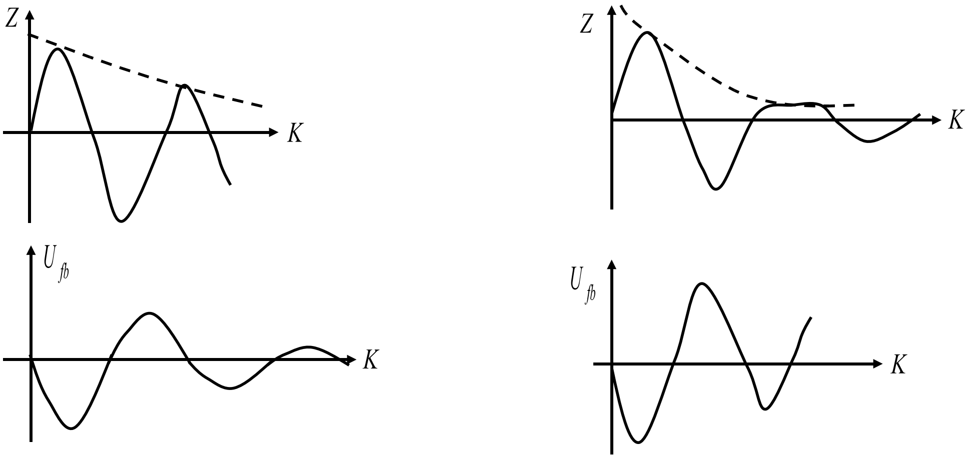


Fig. 48:

What do we mean by “bad” damping and “cheap” control? we now define precisely what we mean. Consider:

$J \equiv \Sigma_{i=0}^{\infty} x_i^T Q x_i + u_i^T R u_i$

The first term penalizes large state executions, the second penalizes large control. $Q \geq 0, R > 0$

Can tradeoff between state excursions and control by varying Q and R

Large $Q \longrightarrow$ “ good” damping importnat

Large $R \longrightarrow$ actuator effort “ expensice”

(Linear quadratic regulator)

$x_{i+1} = Ax_i + Bu_i; \quad x_0$

Find control sequence $u_0, u_1, u_2, ...$ such that $J = \Sigma_{i=0}^{\infty} x_i^T Q x_i + u_i^T R u_i$=minimum

Answer: The optimal control sequence is a state feedback sequence u_{i0}^{∞}

$u_i = -K_{opt} x_i$

$K_{opt} = (R + B^T SB)^{-1} B^T SA$

$S = A^T SA + Q - A^T AB(R + B^T SB)^{-1} B^T SA$

Algebraic Riccati equation (A.R.E) for discrete-time systems.

Note: Since u_i = state feedback, it works for any initial state x_0
Remarks:

1. So optimal control, $u_i = K_{opt} x_i$ is stste feedback! this is why we are interested in state feedback.
2. Equation A.R>E is matrix quadratic equation.Looks pretty intimidating but computer can solve in a second.
3. No tweaking! just specify A, B, C, D and Q and R, press return button, LQR routine spits out $K_{opt}-$ done (of course guaranteed optimal in the sense that it minimizes.
4. Design is guaranteed optimal in the sense that it minimizes.

   $J_{lqr}(x_0, u_{i0}^{\infty} x_i^T Q x_i + u_i^T R u_i$

   (Of course that doesn't mean its "best" in the absolute sense.-
   As vary $\frac{Q}{R}$ ratio we get whole family of k_{lqr}'s i.e can trade-off between state excursion(damping) Vs Actuator effort (control)

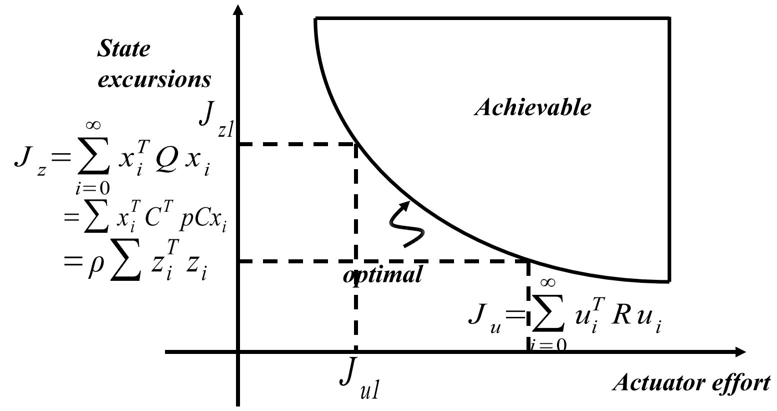

State excursions
J_{z1}
$J_z = \sum_{i=0}^{\infty} x_i^T Q x_i = \sum x_i^T C^T pCx_i = \rho \sum z_i^T z_i$
Achievable
optimal
$J_u = \sum_{i=0}^{\infty} u_i^T R u_i$
J_{ul}
Actuator effort

Fig. 49:

Our optimal control has the form $u_{opt}(k) = -K(k) x_{opt}(k)$

This assumes that we have complete state information $x_{opt}(k)-$ not actually e.g:in SHO, we might have only a position sensor but not a velocity sensor.

8.1 The Kalman Filter

how can we obtain "good" estimates of the velocity state just observing the position state?

Furthermore the sensors may be noisy and the plant itself maybe subject to outside disturbances (process noise) i.e. we are looking for this:
$x(k+1) = Ax(k) + Bw(k)$

$z(k) = Cx(k)$

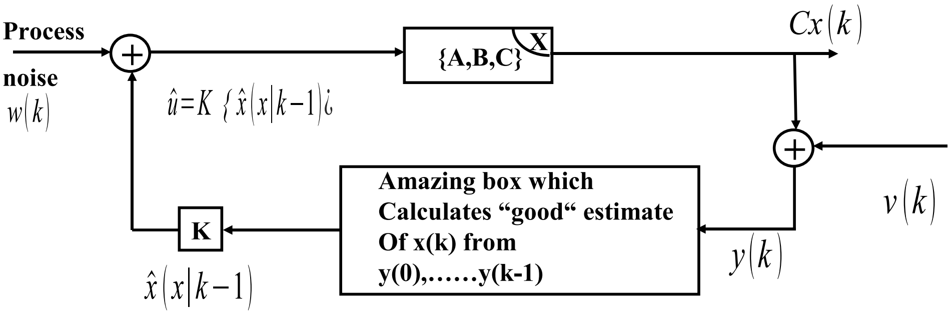


Fig. 50:

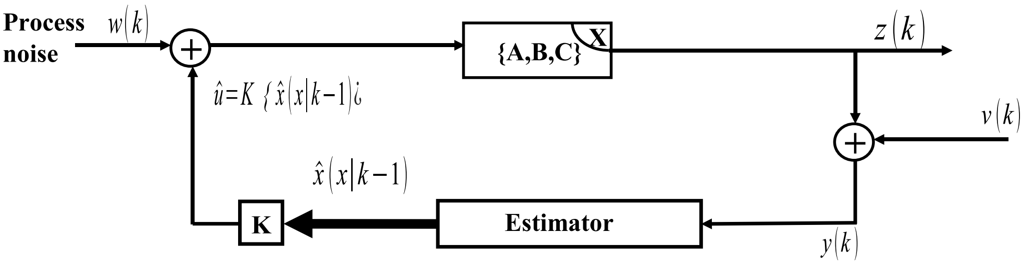


Fig. 51:

$y(k) = Cx(k) + v(k)$ Assumes also $x(0)$ is random and Gaussian and that $x(k), w(k) + v(k)$ are all mutually independent for all k.

Find: $\hat{x}(k|k-1)$ optimal estimate of $x(k)$ given $y_0, ..y_{k-1}$ such that “mean squared error”.

$E[||x(k) - \hat{x}(k|k-1)||_2^2]$=minimal

Fact from statistics: $\hat{x}(k|k-1) = E[x(k)|(y_0, ..., y_{k-1})]$ The Kalman filter is an efficient algorithm that computes the new $\hat{x}_{i+1|i}$ (the linear-least-mean square estimate) of the system state vector x_{i+1}, given $y_0, ..y_i$, by updating the old estimate $\hat{x}_{i|i-1}$ and old $x_{i|\tilde{i}-1}$ (error). The Kalman filter produces $\hat{x}_{i+1|i}$ from

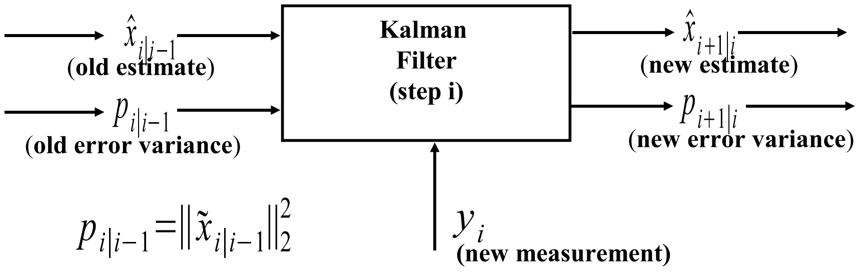


Fig. 52:

$\hat{x}_{i|i-1}$ (rather than $\hat{x}_{i|i}$, because it “tracks" the system “dynamics". By the time we compute $\hat{x}_{i|i}$ from $\hat{x}_{i|i-1}$, the system state has changed from x_i

The kalman filter algorithm can be divided in a measurement update and a time update:

Measurement update(M.U): $\hat{x}_{i|i} = \hat{x}_{i|i-1} + P_{i|i-1}C^T(Cp_{i|i-1}C^T + V)^{-1}(y_i - C\hat{x}_{i|i-1})$

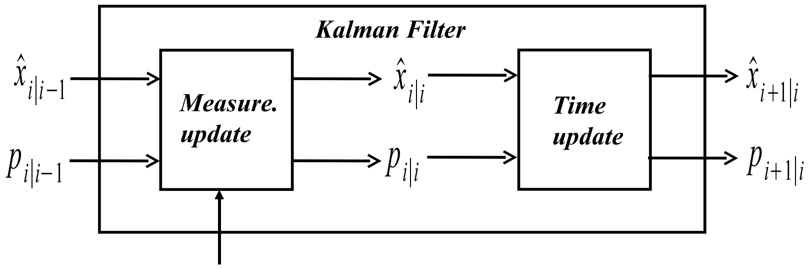


Fig. 53:

$p_{i|i} = P_{i|i-1} - P_{i|i-1}C^T(CP_{i|i-1}C^T + V)^{-1}Cp_{i|i-1}$

Time update (T.U.): $\hat{x}_{i+1|i} = A\hat{x}_{i|i}$ and $P_{i+1|i} = Ap_{i|iA^T} + BWB^T$

With initial conditions: $\hat{x}_{0|-1} = 0$ and $p_{0\hat{|}-1} = X_0$
By plugging M.U equations in to T.U equations one can do both steps at once:

$\hat{x}_{i+1|i} = A\hat{x}_{i|i} = A\hat{x}_{i|i-1} + Ap_{i|i-1}C^T(Cp_{i|i-1}C^T + V)^{-1}(y_i - C\hat{x}_{i|i-1})$

$\hat{x}_{i+1|i} = A\hat{x}_{i|i-1} + L_i(y_i - C\hat{x}_{i|i-1})$ where $L_i \equiv Ap_{i|i}A^T + BWB^T = A[p_{i|i-1} - p_{i|i-1}C^T(Cp_{i|i-1}C^T + V)^{-1}CP_{i|i-1}]A^T + BWB^T$

$p_{i+1|i} = Ap_{i|i}A^T + BWB^T - Ap_{i|i-1}C^T(Cp_{i|i-1}C^T(Cp_{i|i-1}C^T + V)^{-1})(Cp_{i|i-1} - 1)A^T$

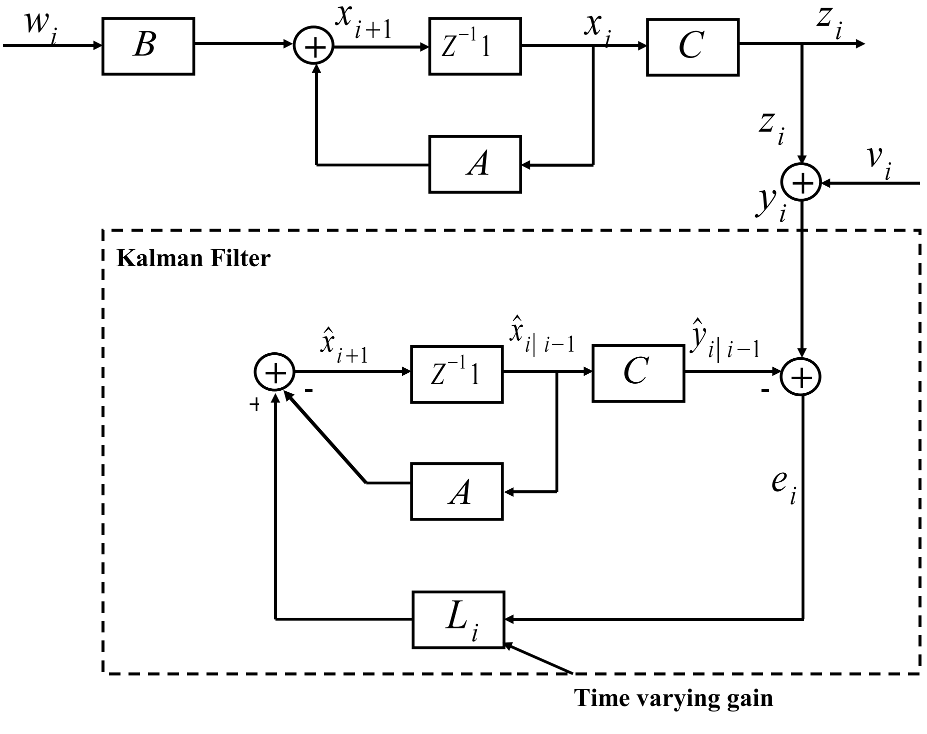


Fig. 54:

Plant equations: $x_{i+1} = Ax_i + Bu_i$ and $y_i = Cx_i + v_i$

Kalman filter: $\hat{x}_{i+1|i} = A\hat{x}_{i|i-1} + L_i(y_i - \hat{y}_{i|i-1})$ and $y_{i|i-1} = C\hat{x}_{i|i-1}$ If $v = w = 0$ kalman filter can esti-

mate the state precisely in a finite number of steps
Remarks:

1. Since $y_i = Cx_i + v_i$ and $\hat{y}_{i|i-1} = C\hat{x}_i$ can write estimator equation as $\hat{x}_{i+1|i} = A\hat{x}_{i|i-1} + L_i(Cx_i + v_i - C\hat{x}_{i|i-1}) = (A - L_iC)x|\hat{i} - 1 + L_iCx_i + v_i$ Can combine this with equation for x_{i+1}

   $$\begin{bmatrix} x_{i+1} \\ \hat{x}_{i+1|i} \end{bmatrix} = \begin{bmatrix} A & 0 \\ L_iC & A - L_iC \end{bmatrix} \begin{bmatrix} x_i \\ \hat{x}_{i|i-1} \end{bmatrix} + \begin{bmatrix} B & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} w_i \\ v_i \end{bmatrix}$$
   $$\begin{bmatrix} z_i \\ \hat{y}_{i|i-1} \end{bmatrix} = \begin{bmatrix} C & 0 \\ 0 & C \end{bmatrix} \begin{bmatrix} x_i \\ \hat{x}_{i|i-1} \end{bmatrix}$$

2. In practice, Riccati equation reaches steady state in few steps.People often run with steady-state K.F.i.e

   $L_ss = Ap_{ss}C^T(CP_{ss}C^T + V)^{-1}$

   Where $p_{ss} = Ap_{ss}A^T + BWB^T - Ap_{ss}C^T(CP_{ss}C^T + V)^{-1}CP_{ss}A$

8.2 LQG Controller

Now we are finally ready to solve the full control problem.

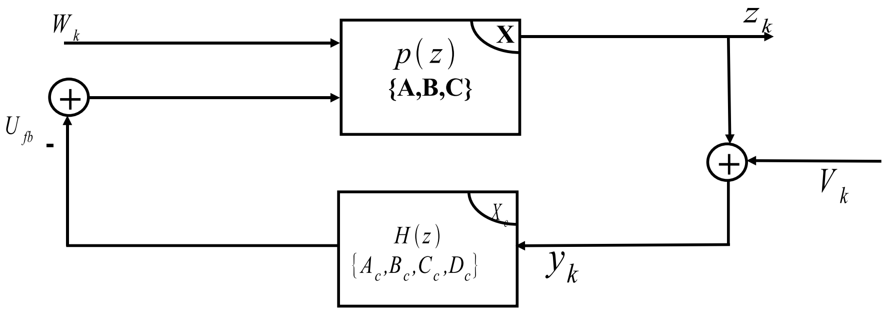


Fig. 55:

given: $x_{k+1} = Ax_k + Bu_k + B_w w_k, z_k = Cx_k$, $y_k = Cx_k + v_k$

$\langle w_i, w_j \rangle = w\delta_{ij}, \langle v_i, v_j \rangle = V\delta_{ij}$, $\langle w_i, v_j \rangle = 0$ where w_k, v_k both Gaussian. For Gaussian, K.L. gives the absolute best estimate.
Separation principle: (We won't prove)The separation principle states that the LGQ optimal controller is obtained by:

1. using Kalman filter to obtain least squares optimal estimate of the plant state,

   i.e:Let $x_c(k) = \hat{x}_{k|k-1}$
2. Feedback estimated LQR-optimal state feedback

   $u(k) = -K_{LQR}x_c(k) = -K_{LQR}\hat{x}_{k|k-1}$ i.e Can treat problems of – Optimal feedback and – state estimate seperately.

Plant: $\|x_{k+1} = Ax_k + (-Bu_k) + B_w w_k, z_k = Cx_k, y_k = Cx_k + v_k\|$

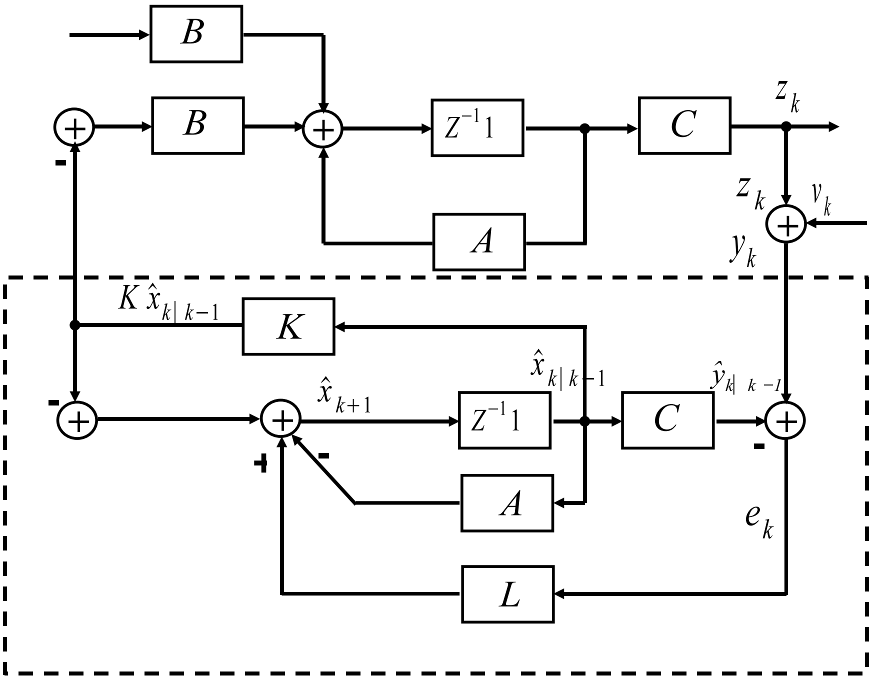


Fig. 56:

LQG Controller $\|\hat{x}_{k+1|k} = A\hat{x}_{k|k-1} + Bu_k + L(y_k - C\hat{x}_{k|k-1}), u_k = -Kk|\hat{k}-1$

$k = -[R+B^TSB]^{-1} + s = A^TSA + Q - A^TSB[R+B^TSB]^{-1}B^TSA$

$L = APC^T[V+CPC^T]^{-1} + P = APA^T + BWB^T[V+CPC^T]^{-1}CPC^T$

Want a controller which takes as input noisy measurements,y, and produces as output a Feedback signal,u, which will minimize excursions of the regulated plant outputs (if no pole-zero cancellation, then this is equivalent to minimizing state excursions.)

Also want to achieve “regulation" with as little actuator efort,u, as possible.

Problem statement (Mathematically) Find:Controller $H(z) = C_c(zI - A_c)^{-1}B_c + D_c$ Controller: $x_c(k+$

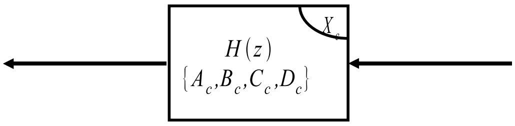


Fig. 57:

$1) = A_cx(k+1) = A_cx(k+1) + B_cy(k), y_c(k) = C_cx_c(k)$ which will minimize the cost, where

$J_{LQG} =_{k\longrightarrow\infty}^{Limit} E[x_k^TQx_k$ Rms “state" excursions $+u_k^TRu_k]$ Rms “actuator" effort.

$\|x_{k+1} = Ax_k + (-Bu_k) + B_ww_k, z_k = Cx_k, y_k = Cx_k + v_k\|$

Remarks:

1. Q and R are weighting matrices that allow trading off rms u and rms x
2. If $Q = C^T\rho C; \rho > 0$ then trade off rms z vs rms u

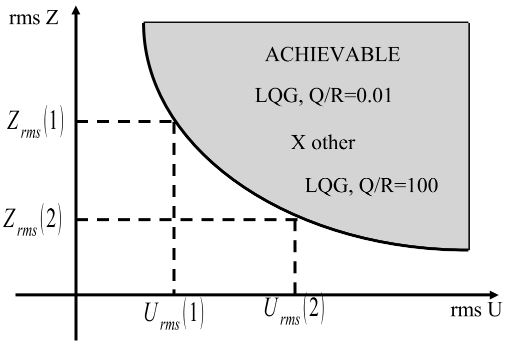


Fig. 58:

3. In the stochastic LQR case, the only difference is that now we don't have complete state information $y_i = Cx_i + v_i$ we have only noisy observations i.e can't use full state feedback

   Idea: Could we use estimated state Feedback? (i.e.$-K\hat{x}_{k|k-1}$)
4. We can let $\frac{Q}{R}$ ratio vary and we will obtain family of LQG controllers. can plot rms Z rms u for each one.
   So by specifying (1) system model,(2) noise variance,(3) optimally criterion J_{LQG} and plotting trade off curve completely specifies limit of performance of System i.e which combinations of (Z_{rms}, U_{rms}) are achievable by any controller good "benchmark curve".

9 System Identification

System identification is a general term to describe mathematical tools and algorithms that build dynamical models from measured data. A dynamical mathematical model in this context is a mathematical description of the dynamic behavior of a system or process.
What is System identification? Using experimental data obtained from input/output relations to model dynamic systems.

Different approaches to system identification depending on model class

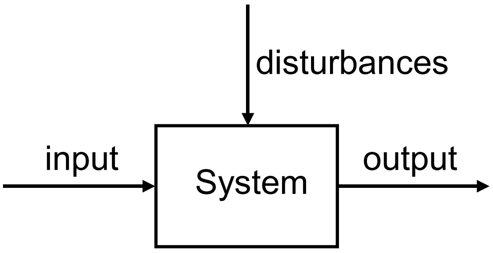


Fig. 59:

1. Linear/Non-Linear
2. parametric/Non-Parametric
   - Non-Parametric methods try to estimate a generic model (step responses, impulse responses, frequency responses)
   - Parametric methods estimate parameters in a user-specified model (transfer functions, state-space matrices)

Identification Steps: System identification is the experimental approach to process modeling.

System identification includes the following steps:

1. **Experimental design:** its purpose is to obtain good experimental data, and it includes the choice of the measured variables and of the character of the input Signals.
2. **Selection of model structure:** A suitable model structure is chosen using prior knowledge and trial and error.
3. **Choice of the criterion to fit:** A suitable cost function is chosen, which reflects how well the model fits the experimental data.
4. **Parameter estimation:** An optimization problem is solved to obtain the numerical values of the model parameters.
5. **Model validation:** The model is tested in order to reveal any inadequacies.

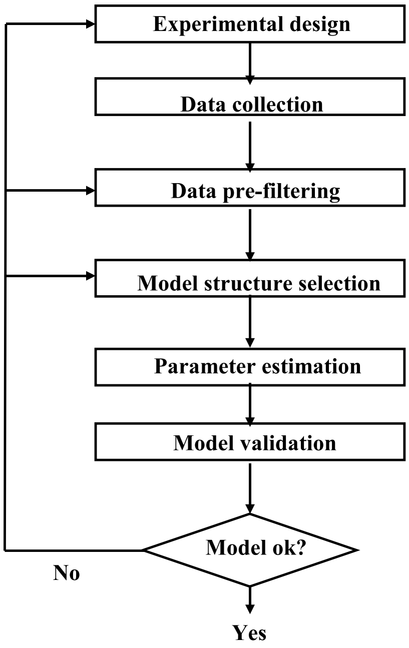


Fig. 60:

Different mathematical models: Model Description:

1. Transfer functions
2. State-Space models
3. Block diagrams(e.g.Simulink)

Notation for continuous time models: Complex Laplace variable s and differential operator p

$$\dot{x}(t) = \frac{\partial x(t)}{\partial t} = px(t)$$

Notation for discrete time models: Complex z transform variable and shift operator q

$$x(k+1) = qx(k)$$

Experiments and data collection: Often good to use a two-stage approach

1. Preliminary experiments
   - Step/Impulse response tests to get basic understanding of system dynamics
   - Linearity, static gains, time delays constants, sampling interval
2. Data collection for model estimation
   - Carefully designed experiment to enable good model fit
   - Operating point, input signal type, number of data points to collect

Preliminary experiments-step response: Useful for obtaining qualitative information about the system

- Dead time (delay)
- Static gain
- Time-constants
- Resonance frequency

Sample time can be determined from time constants

Rule-of-thumb: 4 – 10 sample points over the rise time.

Design experiment for model estimation: Input signal should excite all relevant frequencies

- Estimated model more accurate in frequency ranges where input has high energy
- Pseudo-Random Binary Sequence(PRBS) is usually a good choice

Trade-off in selection of signal amplitude

- Large amplitude gives high signal-to-noise ratio(SNR), low parameter variance
- Most systems are non-linear for large input amplitudes.

Big difference between estimation of system under closed loop control or not!

Collecting data: Sampling time selection and anti-alias are central!

Data pre-filtering:
Remove:

- Transients needed to reach desired operating point
- Mean values of input and output values, i.e. work with

$$\Delta u[t] = u[t] - \frac{1}{N}\Sigma_{t=1}^{N} u[t]$$

$$\Delta y[t] = y[t] - \frac{1}{N}\Sigma_{t=1}^{N} y[t]$$

- Trends (use 'detrend' in MATLAB)
- Outliers (obvious erroneous data points)

General Model Structure

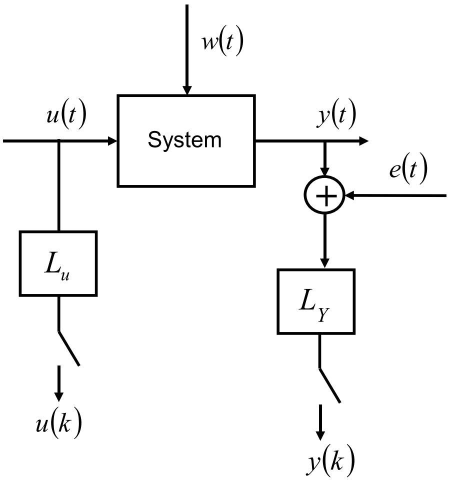


Fig. 61:

1. Systems described in terms of differential equations and transfer functions.
2. Provides insight into the system physics and a compact model structure general-linear polynomial model or the general-linear model.

$$y(k) = q^{-k}G(q^{-1},\Theta)u(k) + H(q^{-1},\Theta)e(k)$$

   - $u(k)$and$y(k)$ are the input and output of the system respectively
   - $e(k)$ is zero-mean white noise, or the disturbance of the system.
   - $G(q^{-1},\Theta)$ is the transfer function of the stochastic part of the system.
   - $H(q^{-1},\Theta$ is the transfer function of the stochastic part of the system.
   - General-Linear model structure:

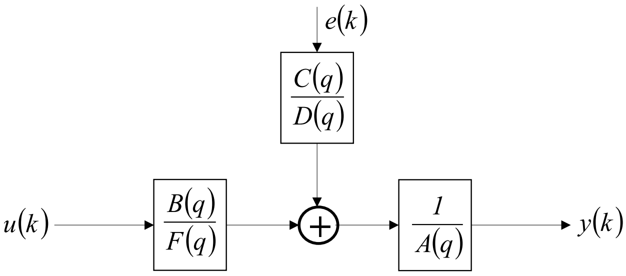


Fig. 62:

3. Provides flexibility for system and stochastic dynamics, Simpler models that are a subset of the Linear model structure.

   Model structures based on input-output:

Model	$\tilde{p}(q)$	$\tilde{p}_e(q)$
ARX	$\frac{B(q)}{A(q)}$	$\frac{1}{A(q)}$
ARMAX	$\frac{B(q)}{A(q)}$	$\frac{C(q)}{A(q)}$
BOX-JENKINS	$\frac{B(q)}{F(q)}$	$\frac{C(q)}{D(q)}$
Output Error	$\frac{B(q)}{F(q)}$	1
FIR	$B(q)$	1

$$A(q)y[k] = \frac{B(q)}{F(q)}u[k] + \frac{C(q)}{D(q)}y[k]\text{or}y[k] = \tilde{p}(q)u[k] + \tilde{p}_e(q)e[k]$$

Provides flexibility for system and stochastic dynamics. Simpler models that are a subset of the General Linear model structure.

Parametric models:

1. Each of these methods has their own advantages and disadvantages and is commonly used in real-world applications
2. Choice of the model structure to use depends on the dynamics and the noise characteristics of the system.
3. Using a model with more freedom or parameters is not always better as it can result in the modeling of nonexistent dynamics and noise characteristics.
4. This is where physical insight into a system is helpful.

   AR Model

   ARX Model

   ARMAX Model

   Box-Jenkins Model

   Output-Error Model

   State-Space Model

9.1 AR Model:

1. Process model where outputs are only dependent on previous outputs.
2. No system inputs or disturbances are used in the modeling
3. Class of solvable problems is limited. For signals not for systems
4. Time series analysis, such as linear prediction coding commonly use the AR model.

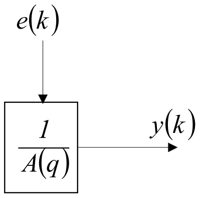


Fig. 63: AR Model Structure

9.2 ARX Model:

Advantages:

- Is the most efficient of the polynomial estimation methods → solving linear regression equations in analytic form.
- Unique solution, that satisfies the global minimum of the loss function.
- Preferable, especially when the model order is high.

Disadvantages:

- Disturbances are part of the system dynamics.
- The transfer function of the deterministic part $G(q^{-1}, \Theta)$ of the system and the transfer function of the stochastic part $H(q^{-1}, \Theta)$ of the system have the same set of poles.
- This coupling can be unrealistic.
- The system dynamics and stochastic dynamics of the system do not share the same set of poles all the time.
- However, you can reduce this disadvantages if you have a good signal-to-noise-ratio.
- Set the model order higher than the actual model order to minimize the equation error, especially when the signal-to-noise ration is low.
- However, increasing the model order can change some dynamic characteristics of the model,such as the stability of the model.

$$y[k] = \frac{B(q)}{A(q)}u[k] + \frac{1}{A(q)}e[k]$$

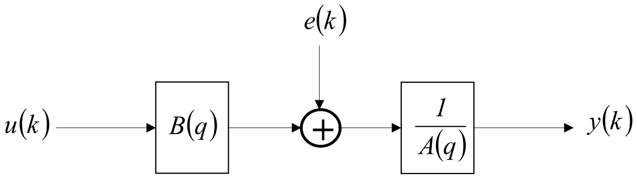


Fig. 64: ARX Model Structure

9.3 ARMAX Model:

1. Includes disturbances dynamics! ARMAX models are useful when you have dominating disturbances that enter early in the process,such as at the input.
   For example, a wind gust affecting an aircraft is a dominating disturbance early in the process.
2. More flexibility in the handling of disturbance modeling.

$$y[k] = \frac{B(q)}{A(q)}u[k] + \frac{C(q)}{A(q)}e[k]$$

9.4 Box Jenkins Model:

- provides a complete model with disturbance properties modeled separately from system dynamics.

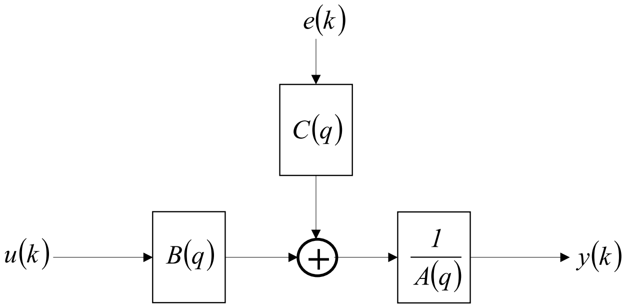


Fig. 65: ARMAX Model Structure

- The Box-Jenkins model is useful when you have disturbances that enter late in the process.

  For example, measurement noise on the output is a disturbance late in the process(not output error model).

$$y[k] = \frac{B(q)}{F(q)}u[k] + \frac{C(Q)}{D(q)}e[k]$$

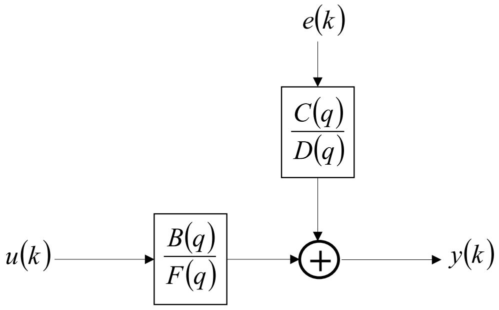


Fig. 66: Box-Jenkins Model Structure

9.5 Output Error Model:

- Describes the system dynamics separately
- No parameters are used for modeling the disturbance characteristics.

$$y[k] = \frac{B(q)}{F(q)}u[k] + e[k]$$

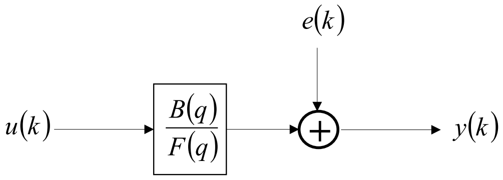


Fig. 67: Output Error Model Structure

9.6 State Space Model:

1. All previous methods are based on minimizing a performance function.
2. For complex high order systems the classical methods can suffer from several problems.
3. Find many local minima in the performance function and do not converge to global minima.
4. The user will need to specify complicated parametrization. They also may suffer potential problems with numerical instability and excessive computation time to execute the iterative numerical minimization methods needed.
5. In addition, modern control methods require a state-space model of the system.
6. For cases such as these the State-Space (SS) identification method is the appropriate model structure.

   Hint: When the model order is high, use an ARX model because the algorithm involved in ARX model estimation is fast and efficient when the number of data points is very large. The state-space model estimation with a large number of data points is slow and requires a large amount of memory. If you must use a state-space model, for example in modern control methods, reduce the sampling rate of the signal in case the sampling rate is unnecessarily high.

   $$x(k+1) = Ax(k) + Bu(k) + Ke(k)$$

   $$y(k) = Cx(k) + Du(k) + e(k)$$

   $x(k)$: state vector (Dim. setting you need to provide for the state-space model)

   $y(k)$: system output

   $u(k)$: System input

   $e(k)$: Stochastic error. A, B, C, D, K are system matrices.
7. In general, they provides a more complete representation of the system, especially for complex MIMO systems, because similar to first principle models.
8. The identification procedure does not involve nonlinear optimization so the estimation reaches a solution regardless of the initial guess.
9. parameter settings for the state-space model are simpler.
10. You need to select only the order / states, of the model.
11. prior knowledge of the system or by analyzing the singular values of A.

Determine Parameters: Determining the delay and model order for the prediction error methods, ARMAX, BJ, and OE, is typically a trial-and-error process.

The following is a useful set of steps that can lead to a suitable model (this is not the only methodology you can use, nor is this a comprehensive procedure).

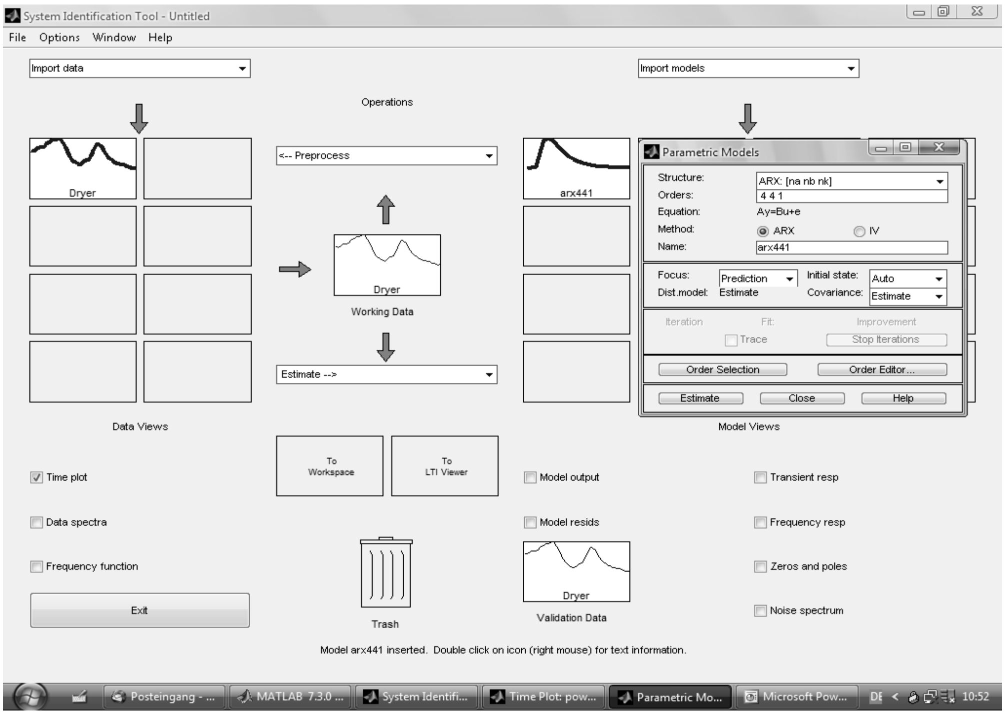
1. Obtain useful information about the model order by observing the number of resonance peaks in the nonparametric frequency response function. Normally, the number of peaks in the magnitude response equals half the order of $A(q), F(q)$
2. Obtain a reasonable estimate of delay using correlation analysis and/or by testing reasonable values in a medium size ARX model. Choose the delay that provides the best model fit based on prediction errors or other fit criterion
3. Test various ARX model orders with this delay choosing those that provide the best fit.

4. Since the ARX model describes both the system dynamics and noise properties using the same set of poles, the resulting model may be unnecessarily high in order.
5. By plotting the zeros and poles (with the uncertainty intervals) and looking for cancelations you can reduce the model order.The resulting order of the poles and zeros are a good starting point for ARMAX, OE and/or BJ models with these orders used as the B and F model parameters and first or second order models for the noise characteristics.
6. Are there additional signals? Measurements can be incorporated as extra input signals!
7. If you cannot obtain a suitable model following these steps additional physical insight into the problem might be necessary (Compensating for nonlinear sensors or actuators and handling of important physical non - linearities are often necessary in addition to using a ready-made model).

From the prediction error standpoint, the higher the order of the model is, the better the model fits the data because the model has more degrees of freedom. However, you need more computation time and memory for higher orders. The parsimony principle advocates choosing the model with the smallest degree of freedom, or number of parameters, if all the models fit the data well and pass the verification test.

Conclusion:

- Variety of model structures available
- Choice is based upon an understanding of the system identification method and algorithm.
- System and disturbance are important
- Handling a wide range of system dynamics without knowing system physics
- Reduction of engineering effort
- Identification Tools (Matlab, Matrix or LabVIEW) are available for developing, prototyping and deploying control algorithms.



Parametric Models

Structure: ARX: [na nb nk]
Orders: 4 4 1
Equation: Ay=Bu+e
Method: ● ARX ○ IV
Name: arx441

Focus: Prediction
Dist.model: Estimate
Initial state: Auto
Covariance: Estimate

☑ Time plot
□ Data spectra
□ Frequency function
□ Model output
□ Model resids
□ Transient resp
□ Frequency resp
□ Zeros and poles
□ Noise spectrum

Model arx441 inserted. Double click on icon (right mouse) for text information.

Fig. 68: Matlab System Identification Tool

10 Non-Linear Systems

In mathematics, a nonlinear system is a system which is not linear i.e. a system which does not satisfy the superposition principle. Less technically, a nonlinear system is any problem where the variable(s) to be solved for cannot be written as a linear sum of independent components. A nonhomogenous system, which is linear apart from the presence of a function of the independent variables, is nonlinear according to a strict definition, but such systems are usually studied alongside linear systems, because they can be transformed to a linear system as long as a particular solution is known.
Generally, nonlinear problems are difficult (if possible) to solve and are much less understandable than linear problems. Even if not exactly solvable, the outcome of a linear problem is rather predictable, while the outcome of a nonlinear is inherently not.
Properties of non-linear Systems: Some properties of non-linear dynamic systems are:

- They do not follow the principle of **superposition** (linearity and homogeneity).
- They may have multiple isolated equilibrium points (**linear systems** can have only one).
- They may exhibit properties such as **limit-cycle**, **bifurcation**, **chaos**
- Finite escape time: The state of an **unstable** nonlinear system can go to infinity in finite time.
- For a **sinusoidal** input, the output signal may contain many harmonics and sub-harmonics with various amplitudes and phase differences (a linear system's output will only contain the sinusoid at the output).

Analysis and Control of Non-linear Systems: There are several well-developed techniques for analyzing nonlinear feedback systems:

- Describing function method
- Phase Plane method
- Lyapunov stability analysis
- Singular Perturbation method
- Popov criterion (described in The Lur'e Problem below)
- Center manifold theorem
- Small-gain theorem
- Passivity analysis

Control Design of Non-linear Systems: Control design techniques for non-linear systems also exist. These can be subdivided into techniques which attempt to treat the system as a linear system in a limited range of operation and use (well-known) linear design techniques for each region:

- Gain Scheduling
- Adaptive control

Those that attempt to introduce auxiliary nonlinear feedback in such a way that the system can be treated as linear for purposes of control design:

- Feedback linearization
- And Lyapunov based methods:
- Lyapunov Redesign
- Back-stepping
- Sliding mode control

10.1 Describing Function Method:

- Procedure for analyzing Non-Linear Control problems based on quasi-linearization
- Replacement of the non-linear system by a system that is linear except for a dependence on the amplitude of the input waveform.must be carried out for a specific family of input waveforms.
- An example might be the family of sine-wave inputs; in this case the system would be characterized by an SIDF or sine input describing function $H(A,\omega)$ giving the system response to an input consisting of a sine wave of amplitude A and frequency ω (This SIDF is a generalization of the transfer function $H(\omega)$ used to characterize linear systems).
- Other types of describing functions that have been used are DF's for level inputs and for Gaussian noise inputs. the DF's often suffice to answer specific questions about control and stability.

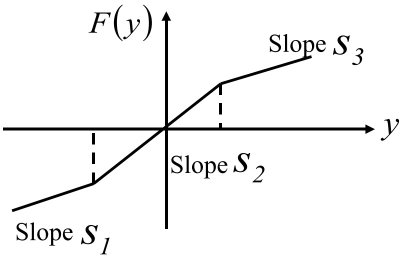


Fig. 69: Example piecewise linear function

10.2 Phase plane method:

- Refers to graphically determining the existence of limit cycles.
- The Phase Plane, applicable for second order systems only, is a plot with axes being the values of the two state variables,$x2vsx1$
- Vectors representing the derivatives at representative points are drawn. With enough of these arrows in place the system behavior over the entire plane can be visualized and limit cycles can be easily identified.

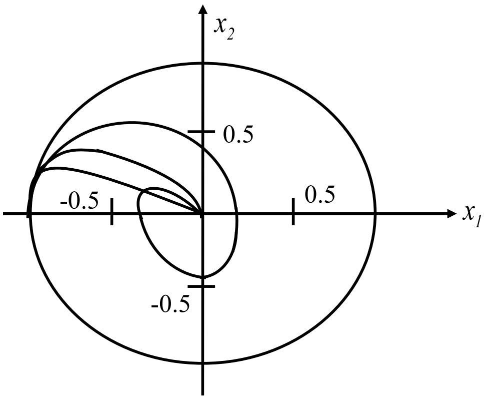


Fig. 70:

10.3 Lyapunov Stability:

- Lyapunov stability occurs in the study of dynamical systems. In simple terms, if all solutions of the dynamical system that start out near an equilibrium point x_e stay near x_e forever, then x_e is Lyapunov stable.

- More strongly, if all solutions that start out near x_e converge to x_e, then x_e is asymptotically stable.
- The notion of exponential stability guarantees a minimal rate of decay, i.e., an estimate of how quickly the solutions converge.
- The idea of Lyapunov stability can be extended to infinite-dimensional manifolds, where it is known as **Structural stability**, which concerns the behavior of different but "nearby" solutions to differential equations.

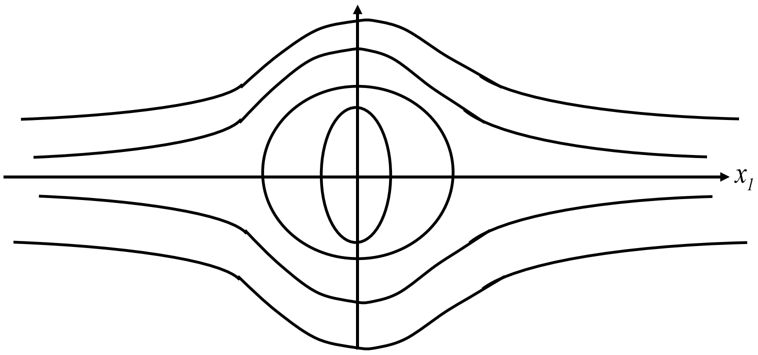


Fig. 71:

10.4 The Lur'e problem:

1. Control systems have a forward path that is linear and time-invariant, and a feedback path that contains a memory-less, possibly time-varying, static non-linearity.

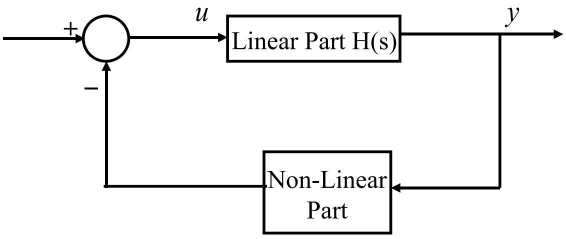


Fig. 72:

2. The linear part can be characterized by four matrices (A,B,C,D), while the non-linear part is $\Phi(y)$ with a sector non-linearity.
3. by A.I.Lur'e

10.5 Popov criterion:

1. The sub-class of Lure's systems studied by **Popov** is described by:

$$\dot{x} = Ax + bu, \dot{\xi} = u$$

$$y = cx + d\xi, u = -\Phi(y)$$

2. where $x \varepsilon R\ n, \xi, u, y$ are scalars and A, b, c, d have commensurate dimensions. The non-linear element $\Phi : R \to R$ is a time-invariant nonlinearity belonging to open sector $(0, \infty)$. This means that

$$\Phi(0) = 0, y\Phi(y) > 0 \forall y \neq 0;$$

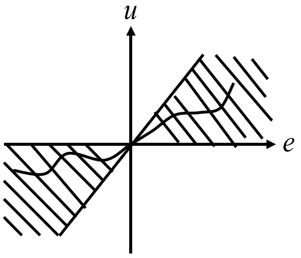


Fig. 73:

Popov Stability Criteria:: →Absolute stability The marked region in figure is called sector

$$0 \leq F(e) \leq ke \ \text{ for } \ e > 0$$

$$ke \leq F(e) \leq 0 \ \text{ for } \ e < 0$$

or shorter

$$0 \leq \frac{F(e)}{e} \leq k$$

The standard control loop is absolute stable in sector $[0,k]$, if the closed loop only has global asymptotic stable equilibrium for any characteristic $F(e)$ lying in this sector.

Prerequisities:

$$G(s) = \frac{1}{s^p}\frac{1+b_1s+...+b_ms^m}{1+a_1s+....a_ns^n}, m < n+p$$

No common zeros of numerators and denominators

$F(e)$ is unique and at least steady in intervals $F(0) = 0$

One Distinguishes the main case $p = 0$, i.e without integrator eigen value and the singular case $p \neq 0$

Cases	p	sector	Inequality
Ia	o	$\frac{F(e)}{e} \geq 0$	$\Re((1+qj\omega)G(j\omega)) > 0$
Ib	0	$k \geq \frac{F(e)}{e} \geq 0$	$\Re((1+qj\omega)G(j\omega)) > -\frac{1}{k}$
IIa	$1,...,n-m$	$k \geq \frac{F(e)}{e} \geq 0$	$\Re((1+qj\omega)G(j\omega)) > -\frac{1}{k}$
IIb	$1,...,n-m$	$\frac{F(e)}{e} > 0$	$\Re((1+qj\omega)G(j\omega)) \geq 0$

- The control loop is absolute stable in the given sector, when a real number q is found for all $\omega \leq 0$ apply on the right side of inequality.
- The Popov criteria is a sufficient condition for absolute stability of the control loop.

Geometric interprettion of the Popov inequality:

- For the case Ib and IIa, the inequality can be rewritten as follows

$$\Re((1+qj\omega)G(j\omega)) > -\frac{1}{k}$$

$$\Re(G(j\omega)) + q\Re(j\omega(G(j\omega))) > -\frac{1}{k}$$

$$\Re(G(j\omega)) + q\Re(j\omega\Re(G(j\omega)) > -\omega\Im(G(j\omega))) > -\frac{1}{k}$$

$$\Re(G(j\omega)) - q\omega\Im(G(j\omega)) > -\frac{1}{k} \longrightarrow 1$$

– One defines the modified frequency plot

$$G_p(j\omega) = \Re(G(j\omega)) + j\omega\Im(G(j\omega))$$

– This frequency plot is called Popov curve, so equation 1 can be represented as

$$\Im(G_p(j\omega)) < \frac{1}{q}(\Re(G_p(j\omega)) + \frac{1}{k})$$

Example of Popov criteria: The non-linearity is shown as a three-point characteristic, which already hosts the amplification K of the linear subsystem. Popov frequency response locus $Z = Lp(j\omega)$ The line

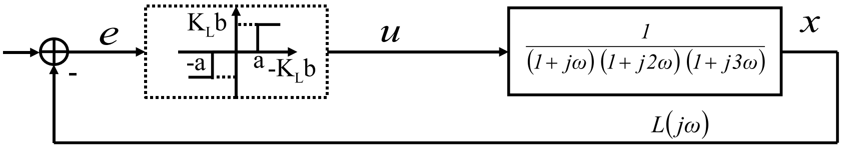


Fig. 74:

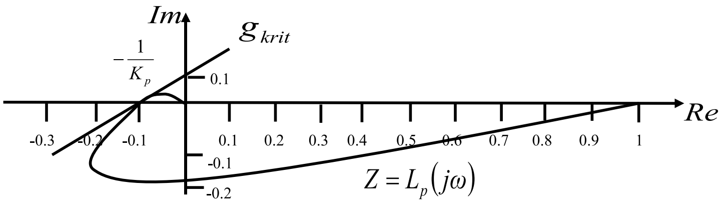


Fig. 75:

g_{krit} is tangential to the Popov frequency response locus curve and intersects the real axis at 0.1 The Popov sector is defined by the line K_pe To verify Popov criteria the gradient of three-point characteristic

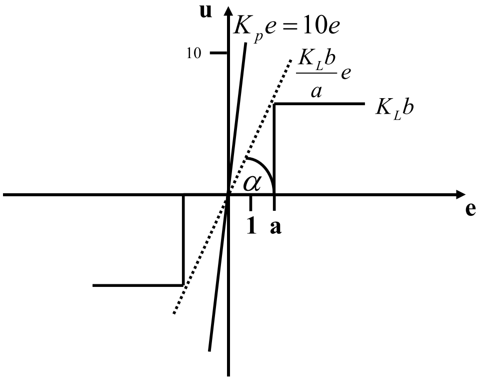


Fig. 76:

must be smaller the calculated gradient

$$tan\alpha = \frac{K_L b}{a}$$

$$\frac{K_L b}{a} < 10$$

If this is true for the given inequality, then is idle state global asymptotic stable

11 Non-Linear Control

Non-linear control is a sub-division of control engineering which deals with the control of non-linear systems. The behaviour of a non-linear system cannot be described as a linear function of the state of that system or the input variables to that system. For linear systems, there are many well-established control techniques, for example root-locus, Bode plot, Nyquist criterion, state-feedback, pole-placement etc.

11.1 Gain Scheduling:

It is an approach to control of **non-linear systems** that uses a family of **linear controllers**, each of which provides satisfactory control for a different operating point of the system.

One or more observable variables, called the scheduling variables, are used to determine what operating region the system is currently in and to enable the appropriate linear controller.

Example: In an aircraft flight control system, the altitude and Mach number might be the scheduling variables, with different linear controller parameters available (and automatically plugged into the controller) for various combinations of these two variables.

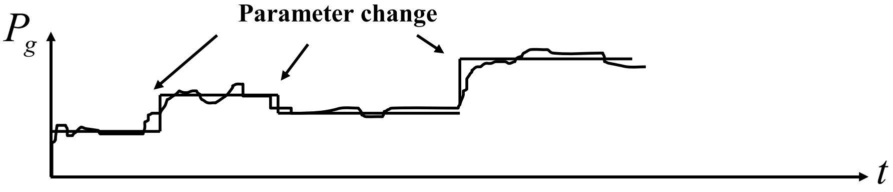


Fig. 77:

11.2 Adaptive Control:

1. modifying the controller parameters to compensate slow timeŰvarying behavior or uncertainties.
2. For example, as an aircraft flies, its mass will slowly decrease as a result of fuel consumption; we need a control law that adapts itself to such changing conditions.
3. Adaptive control does not need a priori information about the bounds on these uncertain or time-varying parameters;
4. Robust control guarantees that if the changes are within given bounds the control law need not be changed.
5. while adaptive control is precisely concerned with control law changes.

There are several broad categories of feedback adaptive control (classification can vary):

Dual Adaptive Controllers Optimal Controllers. – Suboptimal Dual Controllers

Nondual Adaptive Controllers. – Model Reference Adaptive Controllers (MRACs) [incorporate a reference model defining desired closed loop performance]

MRAC

MIA

- Gradient Optimization MRACs [use local rule for adjusting parameters when performance differs from reference]
- Stability Optimized MRACs.
- Model Identification Adaptive Controllers (MIACs) [perform **System identification** while the system is running].
- Cautious Adaptive Controllers [use current SI to modify control law, allowing for SI uncertainty]. Nonparametric Adaptive Controllers
- Parametric Adaptive Controllers
- Explicit Parameter Adaptive Controllers
- Implicit Parameter Adaptive Controllers

11.3 Feedback Linearization:

1. Common approach used in controlling nonlinear systems.
2. transformation of the nonlinear system into an equivalent linear system through a change of variables and a suitable control input.
3. Feedback linearization may be applied to nonlinear systems of the following form:

   $$\dot{x} = f(x) + g(x)u$$

   $$y = h(x)$$

   Where x is the state vector, u is the vector of inputs, and y is the vector of outputs.
4. The goal, then, is to develop a control input u that renders either the input-output map linear, or results in a linearization of the full state of the system.

Existence of a state Fb control?

$$u = \alpha(x) + \beta(x)y$$

And variable change?

$$z = T(x)$$

Pendulum Example: Task:Stabilization of origin of pendulum eq

$$\dot{x_1} = x_2$$

$$\dot{x_2} = a[sin(x_1 + \delta) - sin(\delta)] - bx_2 + cu$$

1. Choose u as follows to cancel nonlinear term:

   $$u = \frac{a}{c}[sin(x_1 + \delta) - sin(\delta)] + \frac{v}{c}$$

   From the above two equations: Linear System

   $$\dot{x_1} = x_2$$

   $$\dot{x_2} = -b.x_2 + v$$
2. Stabilizing fb controller

   $$v = -k_1 x_1 - k_2 x_2$$

   Closed loop

   $$\dot{x}_1 = x_2$$

   $$\dot{x}_2 = k_1 x_1 - (k_2 + b)x_2$$

3. Re-substituting to or fb control law

$$u = \frac{a}{c}[sin(x_1 + \delta) - sin(\delta)] - \frac{1}{c}(k_1 x_1 + k_2 x_2)$$

11.4 Sliding Mode Control:

In Control theory sliding mode control is a type of variable structure control where the dynamics of a nonlinear system is altered via application of a high-frequency switching control. This is a state feedback control scheme where the feedback is not a continuous function of time.

This control scheme involves following two steps:

- Selection of a hyper surface or a manifold such that the system trajectory exhibits desirable behavior when confined to this manifold.
- Finding feed-back gains so that the system trajectory intersects and stays on the manifold.

Advantages:

- Controllers with switching elements are easy to realize.
- The controller outputs can be restricted to a certain region of operation.
- Can also be used inside the region of operation to enhance the closed loop dynamics

Two types of switching controllers

- Switching between constant values
- Switching between state or output dependent values

Design of a switching controller therefore needs:

- The definition of a switching condition
- The choice of the control law inside the areas separated by the switching law

Phase plot of a closed loop system with sliding mode controller

$$(y = 0, \dot{y} = 1 \ \text{ and } \ m = 1)$$

Consider a linear system

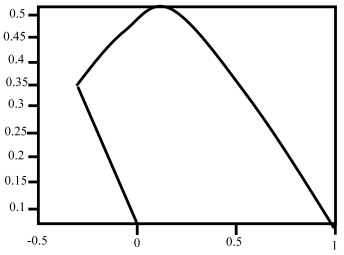


Fig. 78:

$$\ddot{y} = u$$

The control law

$$u = -ky \ \text{ with } \ k > 0$$

It is clear that controller is not able to stabilise the semi stable system that has two open loop eigen values of zero. The root locus plot with two branches along the imaginary axis.

Switching function:

$$s(y,\dot{y}) = my + \dot{y}$$

With a scalar $m > 0$ then a variable structure control law

$$u = -1 \;\; s(y,\dot{y}) > 0, 1 \;\; s(y,\dot{y}) < 0$$

Phase portrait with example: Consider a second order system

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = h(x) + g(x)u$$

$g(x)$ and $h(x)$ are unknown non-linear functions.

On the manifolds $s = 0$ motion

$$s = a_1 x_1 + x_2 = 0 \rightarrow \dot{x} = -ax$$

Independent of $g(x)$ and $h(x)$

Goal:state FB → stabilize the origin.

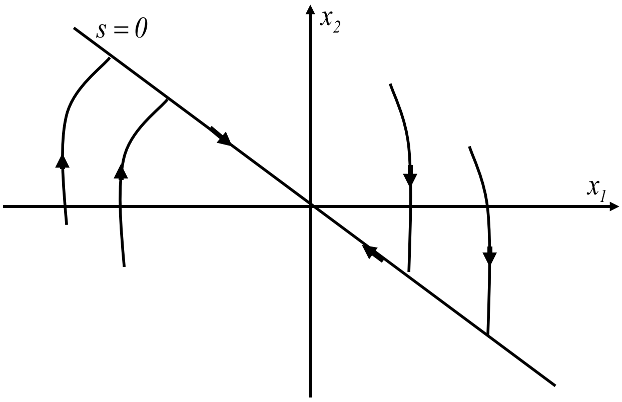


Fig. 79:

Chattering on sliding manifold:

- Caused by delays (no ideal would run on manifold).
- Region $s < 0 \rightarrow$ Trajectory reverses
- Delay leads to crossing again

Disadvantages:

- Low control accuracy.
- Power losses
- Excitation of un modeled of dynamics
- Can result in instabilities.

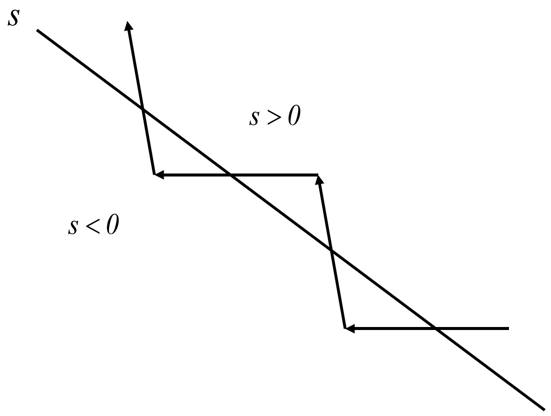


Fig. 80:

References

[1] Chen, Chi-Tsong, *“Linear System Theory and Design,"* 3rd. ed., Oxford University Press (ISBN $0-19-511777-8$), 1999.

[2] Khalil, Hassan K, *“Nonlinear Systems,"* 3rd. ed., Prentice Hall (ISBN $0-13-067389-7$)

[3] Nise, Norman S, *“Control Systems Engineering,"* 4th ed., John Wiley and Sons, Inc. (ISBN $0-471-44577-0$), 2004.

[4] Hinrichsen, Diederich and Pritchard, Anthony J., *“Mathematical Systems Theory I, Modelling, State Space Analysis, Stability and Robustness,"* Springer. (ISBN $978-3-540-44125-0$), 2005

[5] Sontag, Eduardo D, *“Mathematical Control Theory: Deterministic Finite Dimensional Systems. Second Edition. Springer,"*(ISBN $0-387-984895$), 1999

[6] Christopher Kilian, *“Modern Control Technology. Thompson Delmar Learning,"* (ISBN $1-4018-5806-6$), 2005.

[7] Vannevar Bush, *“ Operational Circuit Analysis,"* John Wiley and Sons, Inc..

[8] Robert F. Stengel, *“Optimal Control and Estimation. "* Dover Publications. (ISBN $0-486-68200-5$, ISBN$-13: 978-0-486-68200-6$, 1994.

[9] Franklin et al, *“ Feedback Control of Dynamic Systems"* 4, New Jersey: Prentice Hall. (ISBN $0-13-032393-4$), 2002.

[10] Joseph L. Hellerstein, Dawn M. Tilbury, and Sujay Parekh , *“Feedback Control of Computing Systems "* John Wiley and Sons. (ISBN $0-47-126637-X$), (ISBN$-13: 978-0-471-26637-2$),2004.

[11] Diederich Hinrichsen and Anthony J. Pritchard , *“Mathematical Systems Theory I - Modelling, State Space Analysis, Stability and Robustness. "* Springer. (ISBN $0-978-3-540-44125-0$, 2005.

[12] Sontag, Eduardo, *“Mathematical Control Theory: Deterministic Finite Dimensional Systems "* Springer.Second Edition. Springer. (ISBN $0-387-984895$), 1998.

[13] Gordon E. Carlson , *“Signal and Linear Systems Analysis with Matlab second edition, "* Wiley, (ISBN $0-471-12465-6$), 1998.

[14] John G. Proakis and Dimitris G. Mandalokis , *“Digital Signal Processing Principals, Algorithms and Applications "* third edition, Prentice Hall,(ISBN 0133737624), 1996.

[15] D. Ronald Fannin, William H. Tranter, and Rodger E. Ziemer , *“Signals and Systems Continuous and Discrete "* fourth edition, Prentice Hall, (ISBN $0-13-496456-X$), 1998.

[16] Boaz Porat, *“A Course in Digital Signal Processing, "* Wiley, (ISBN 0471149616).

[17] Lennart Ljung, *“System Identification Ů Theory For the User, "* , 2nd ed, PTR Prentice Hall, Upper

Saddle River, N.J., 1999.

[18] Jer-Nan Juang, *“Applied System Identification, "* ,Prentice Hall, Upper Saddle River, N.J., 1994.

[19] A. I. Lur’e and V. N. Postnikov, *“On the theory of stability of control systems, Applied mathematics and mechanics, "* , 8(3), 1944, (in Russian).

[20] M. Vidyasagar, *“Nonlinear Systems Analysis, "* ,second edition, Prentice Hall, Englewood Cliffs, New Jersey 07632.

[21] A. Isidori, *“ Nonlinear Control Systems, "* , third edition, Springer Verlag, London, 1995.

[22] H. K. Khalil, *“ Nonlinear Systems,"* , third edition, Prentice Hall, Upper Saddle River, New Jersey, 2002.