Lecture 13 — Nonlinear Control Synthesis Cont'd

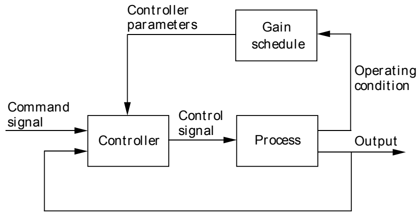
Today's Goal: *To understand the meaning of the concepts*

- *Gain scheduling*
- *Internal model control*
- *Model predictive control*
- *Nonlinear observers*
- *Lie brackets*

Material:

- Lecture notes
- Internal model, more info in e.g.,
  - Section 8.4 in [Glad&Ljung]
  - Ch 12.1 in [Khalil]

Gain Scheduling

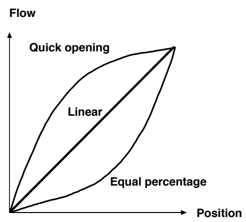


Example of scheduling variables

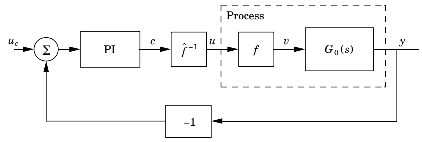
- Production rate
- Machine speed
- Mach number and dynamic pressure

Compare structure with adaptive control!

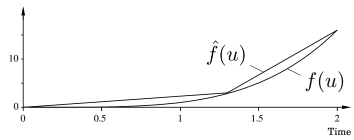
Valve Characteristics



Nonlinear Valve

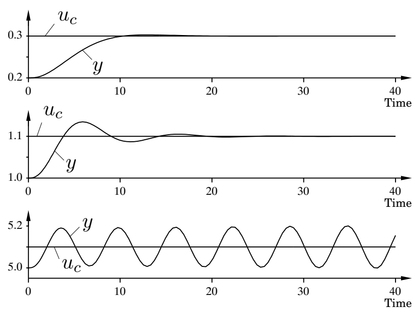


Valve characteristics



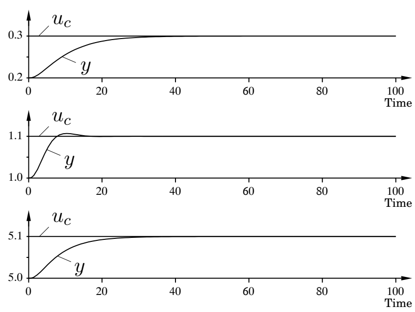
Results

Without gain scheduling



Results

With gain scheduling



Gain Scheduling

- state dependent controller parameters.
  - $K = K(q)$
- design controllers for a number of operating points.
  - use the closest controller.

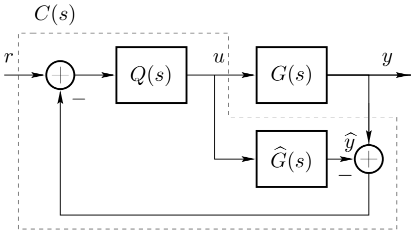
Problems:

- How should you switch between different controllers?
  - Bumpless transfer
- Switching between stabilizing controllers can cause instability.

Outline

- ○ Gain scheduling
- • **Internal model control**
- ○ Model predictive control
- ○ Nonlinear observers
- ○ Lie brackets

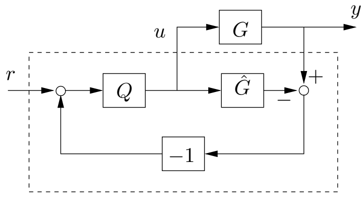
Internal Model Control

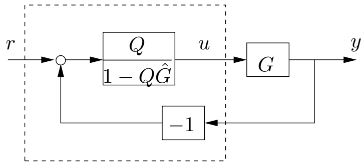


Feedback from model error $y - \widehat{y}$.

Design: Choose $\widehat{G} \approx G$ and Q stable with $Q \approx G^{-1}$.

Two equivalent diagrams





Example

$$G(s) = \frac{1}{1 + sT_1}$$

Choose

$$Q = \frac{1 + sT_1}{1 + \tau s}$$

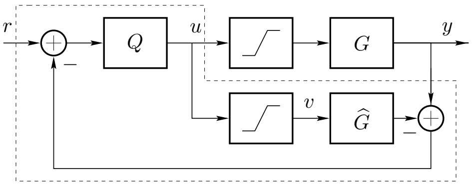
Gives the PI controller

$$C = \frac{1 + sT_1}{s\tau} = \frac{T_1}{\tau}\left(1 + \frac{1}{T_1 s}\right)$$

Internal Model Control Can Give Problems

- Unstable G
- $Q \not\approx G^{-1}$ due to RHP zeros
- Cancellation of process poles may show up in some signals

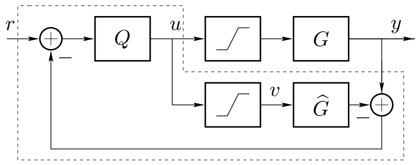
Internal Model Control with Static Nonlinearity



Include the nonlinearity in the model in the controller.

Choose $Q \approx G^{-1}$.

Example (cont'd)



Assume $r = 0$ and $\widehat{G} = G$:

$$u = -Q(y - \widehat{G}v) = -\frac{1 + sT_1}{1 + \tau s} y + \frac{1}{1 + \tau s} v$$

Same as before if $|u| \leq u_{\text{max}}$: Integrating controller.

If $|u| > u_{\text{max}}$ then

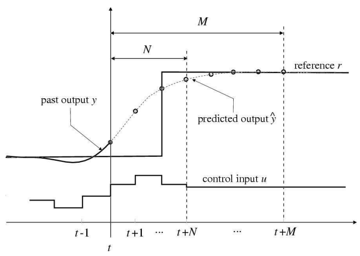
$$u = -\frac{1 + sT_1}{1 + \tau s} y \pm \frac{u_{\text{max}}}{1 + \tau s}$$

No integration. (A way to implement anti-windup.)

Outline

- Gain scheduling
- Internal model control
- **Model predictive control**
- Nonlinear observers
- Lie brackets

Model Predictive Control – MPC



1. Derive the future controls $u(t+j), \quad j = 0, 1, \ldots, N-1$ that give an optimal predicted response.
2. Apply the first control $u(t)$.
3. Start over from 1 at next sample.

What is Optimal?

Minimize a cost function, V, of inputs and predicted outputs.

$$V = V(U_t, Y_t), \quad U_t = \begin{bmatrix} u(t+N-1) \\ \vdots \\ u(t) \end{bmatrix}, \quad Y_t = \begin{bmatrix} \hat{y}(t+M|t) \\ \vdots \\ \hat{y}(t+1|t) \end{bmatrix}$$

V often quadratic

$$V(U_t, Y_t) = Y_t^T Q_y Y_t + U_t^T Q_u U_t \tag{1}$$

$\Longrightarrow$ linear controller

$$u(t) = -L\hat{x}(t|t)$$

Model Predictive Control

+ Flexible method
  * Many types of models for prediction:
    - state space, input–output, step response, FIR filters
  * MIMO
  * Time delays
+ Can include constraints on input signal and states
+ Can include future reference and disturbance information
– On-line optimization needed
– Stability (and performance) analysis can be complicated

Typical application:
Chemical processes with slow sampling (minutes)

A predictor for Linear Systems

Discrete-time model

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) + B_v v_1(t) \\ y(t) &= Cx(t) + v_2(t) \end{aligned} \qquad t = 0, 1, \dots$$

Predictor (v unknown)

$$\begin{aligned} \hat{x}(t+k+1|t) &= A\hat{x}(t+k|t) + Bu(t+k) \\ \hat{y}(t+k|t) &= C\hat{x}(t+k|t) \end{aligned}$$

The M-step predictor for Linear Systems

$\hat{x}(t|t)$ is predicted by a standard Kalman filter, using outputs up to time t, and inputs up to time $t-1$.

Future predicted outputs are given by

$$\begin{bmatrix} \hat{y}(t+M|t) \\ \vdots \\ \hat{y}(t+1|t) \end{bmatrix} = \begin{bmatrix} CA^M \\ \vdots \\ CA \end{bmatrix} \hat{x}(t|t) + \begin{bmatrix} CB & CAB & CA^2B & \dots \\ 0 & CB & CAB & \dots \\ \vdots & \ddots & \ddots & \vdots \end{bmatrix} \begin{bmatrix} u(t+M-1) \\ \vdots \\ u(t+N-1) \\ \vdots \\ u(t) \end{bmatrix}$$

$$Y_t = D_x \hat{x}(t|t) + D_u U_t$$

Limitations

Limitations on control signals, states and outputs,

$$|u(t)| \le C_u \quad |x_i(t)| \le C_{x_i} \quad |y(t)| \le C_y,$$

leads to linear programming or quadratic optimization.

Efficient optimization software exists.

Design Parameters

- Model
- M (look on settling time)
- N as long as computational time allows
- If $N < M-1$ assumption on $u(t+N), \dots, u(t+M-1)$ needed (e.g., $=0$, $= u(t+N-1)$.)
- Q_y, Q_u (trade-offs between control effort etc)
- C_y, C_u limitations often given
- Sampling time

Product: ABB Advant

Example–Motor

$$A = \begin{pmatrix} 1 & 0.139 \\ 0 & 0.861 \end{pmatrix}, \quad B = \begin{pmatrix} 0.214 \\ 2.786 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \end{pmatrix}$$

Minimize $V(U_t) = \|Y_t - R\|$ where $R = \begin{pmatrix} r \\ \vdots \\ r \end{pmatrix}$, r=reference,

$M = 8$, $N = 2$, $u(t+2) = u(t+3) = u(t+7) = \dots = 0$

Example–Motor

$$Y_t = \begin{pmatrix} CA^8 \\ \vdots \\ CA \end{pmatrix} x(t) + \begin{pmatrix} CA^6B & CA^7B \\ \vdots & \vdots \\ 0 & CB \end{pmatrix} \begin{pmatrix} u(t+1) \\ u(t) \end{pmatrix}$$
$$= D_x x(t) + D_u U_t$$

Solution without control constraints

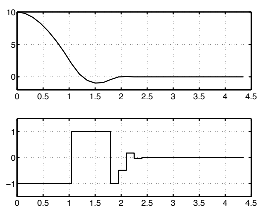
$$\begin{aligned} U_t &= -(D_u^T D_u)^{-1} D_u^T D_x x + (D_u^T D_u)^{-1} D_u^T R = \\ &= -\begin{pmatrix} -2.50 & -0.18 \\ 2.77 & 0.51 \end{pmatrix} \begin{pmatrix} x_1(t) - r \\ x_2(t) \end{pmatrix} \end{aligned}$$

Use

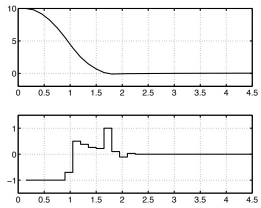
$$u(t) = -2.77(x_1(t) - r) - 0.51 x_2(t)$$

Example–Motor–Results

No control constraints in optimization (but in simulation)



Control constraints $|u(t)| \le 1$ in optimization.



Outline

- Gain scheduling
- Internal model control
- Model predictive control
- **Nonlinear observers**
- Lie brackets

Nonlinear Observers

What if x is not measurable?

$$\dot{x} = f(x,u), \quad y = h(x)$$

Simplest observer (open loop – only works for as. stable systems).

$$\dot{\widehat{x}} = f(\widehat{x}, u)$$

Correction, as in linear case,

$$\dot{\widehat{x}} = f(\widehat{x}, u) + K(y - h(\widehat{x}))$$

Choices of K

- Linearize f at x_0, find K for the linearization
- Linearize f at $\widehat{x}(t)$, find $K(t)$ for the linearization

Second case is called *Extended Kalman Filter*

A Nonlinear Observer for the Pendulum

Control tasks:

1. Swing up
2. Catch
3. Stabilize in upward position

The observer must to be valid for a complete revolution

A Nonlinear Observer for the Pendulum

$$\frac{d^2\theta}{dt^2} = \sin\theta + u\cos\theta$$

$x_1 = \theta$, $x_2 = \frac{d\theta}{dt} \Longrightarrow$

$$\frac{dx_1}{dt} = x_2$$
$$\frac{dx_2}{dt} = \sin x_1 + u\cos x_1$$

Observer structure:

$$\frac{d\hat{x}_1}{dt} = \hat{x}_2 \qquad +k_1(x_1 - \hat{x}_1)$$
$$\frac{d\hat{x}_2}{dt} = \sin\hat{x}_1 + u\cos\hat{x}_1 \qquad +k_2(x_1 - \hat{x}_1)$$

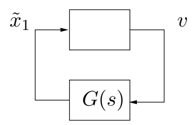
A Nonlinear Observer for the Pendulum

Introduce the error $\tilde{x} = \hat{x} - x$

$$\begin{cases} \dfrac{d\tilde{x}_1}{dt} = -k_1\tilde{x}_1 + \tilde{x}_2 \\ \dfrac{d\tilde{x}_2}{dt} = \sin\hat{x}_1 - \sin x_1 + u(\cos\hat{x}_1 - \cos x_1) - k_2\tilde{x}_1 \end{cases}$$

$$\frac{d}{dt}\begin{bmatrix}\tilde{x}_1\\ \tilde{x}_2\end{bmatrix} = \begin{bmatrix}-k_1 & 1\\ -k_2 & 0\end{bmatrix}\begin{bmatrix}\tilde{x}_1\\ \tilde{x}_2\end{bmatrix} + \begin{bmatrix}0\\1\end{bmatrix}v$$

$$v = 2\sin\frac{\tilde{x}_1}{2}\Big(\cos(x_1 + \frac{\tilde{x}_1}{2}) - u\sin(x_1 + \frac{\tilde{x}_1}{2})\Big)$$



Stability with Small Gain Theorem

The linear block:

$$G(s) = \frac{1}{s^2 + k_1 s + k_2} = \frac{1}{s^2 + 2\zeta\omega_0 s + \omega_0^2}$$

With $\zeta \ge \frac{1}{\sqrt{2}}$, this gives

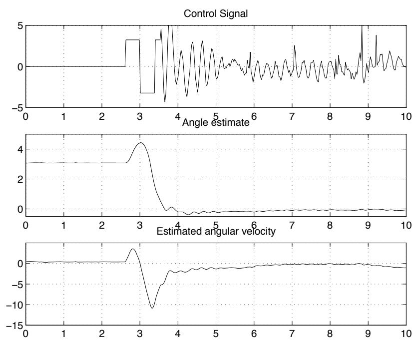
$$\gamma_G = \max|G(i\omega)| = |G(0)| = \frac{1}{\omega_0^2}$$

Moreover

$$|v| = \left|2\sin\frac{\tilde{x}_1}{2}\Big(\cos(x_1 + \frac{\tilde{x}_1}{2}) - u\sin(x_1 + \frac{\tilde{x}_1}{2})\Big)\right| \le |\tilde{x}_1|\sqrt{1 + u_{\max}^2}$$

so the observer is stable by the small gain theorem provided that $k_2 = \omega_0^2$ is selected to satisfy $\frac{1}{\omega_0^2}\sqrt{1 + u_{\max}^2} \le 1$.

A Nonlinear Observer for the Pendulum



Outline

- ○ Gain scheduling
- ○ Internal model control
- ○ Model predictive control
- ○ Nonlinear observers
- • **Lie brackets**

Controllability

Linear case

$$\dot{x} = Ax + Bu$$

All controllability definitions coincide

$$0 \to x(T),$$
$$x(0) \to 0,$$
$$x(0) \to x(T)$$

T either fixed or free

Rank condition System is controllable iff

$$W_n = \begin{pmatrix} B & AB & \dots & A^{n-1}B \end{pmatrix} \quad \text{full rank}$$

Is there a corresponding result for nonlinear systems?

Lie Brackets

Lie bracket between $f(x)$ and $g(x)$ is defined by

$$[f, g] = \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g$$

Example:

$$f = \begin{pmatrix} \cos x_2 \\ x_1 \end{pmatrix}, \qquad g = \begin{pmatrix} x_1 \\ 1 \end{pmatrix},$$

$$\begin{aligned} [f, g] &= \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \cos x_2 \\ x_1 \end{pmatrix} - \begin{pmatrix} 0 & -\sin x_2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos x_2 + \sin x_2 \\ -x_1 \end{pmatrix} \end{aligned}$$

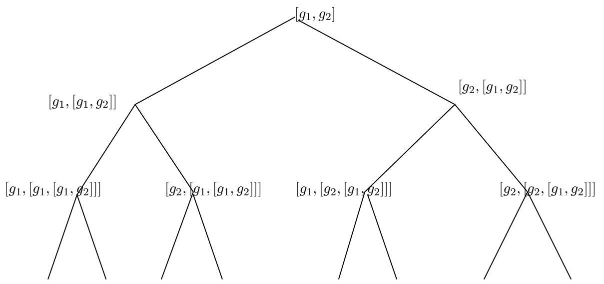
Why interesting?

$$\dot{x} = g_1(x)u_1 + g_2(x)u_2$$

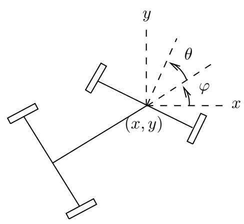
- The motion $(u_1, u_2) = \begin{cases} (1, 0), & t \in [0, \epsilon] \\ (0, 1), & t \in [\epsilon, 2\epsilon] \\ (-1, 0), & t \in [2\epsilon, 3\epsilon] \\ (0, -1), & t \in [3\epsilon, 4\epsilon] \end{cases}$

  gives motion $x(4\epsilon) = x(0) + \epsilon^2[g_1, g_2] + O(\epsilon^3)$
- $\Phi^t_{[g_1,g_2]} = \lim_{n\to\infty} (\Phi^{\sqrt{\frac{t}{n}}}_{-g_2} \Phi^{\sqrt{\frac{t}{n}}}_{-g_1} \Phi^{\sqrt{\frac{t}{n}}}_{g_2} \Phi^{\sqrt{\frac{t}{n}}}_{g_1})^n$
- The system is controllable if the **Lie bracket tree** has full rank (controllable=the states you can reach from $x = 0$ at fixed time T contains a ball around $x = 0$)

The Lie Bracket Tree



Parking Your Car Using Lie-Brackets



$$\frac{d}{dt}\begin{pmatrix} x \\ y \\ \varphi \\ \theta \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_1 + \begin{pmatrix} \cos(\varphi + \theta) \\ \sin(\varphi + \theta) \\ \sin(\theta) \\ 0 \end{pmatrix} u_2$$

Parking the Car

Can the car be moved sideways?

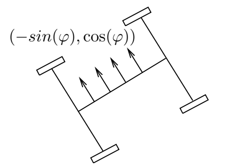
Sideways: in the $(-\sin(\varphi), \cos(\varphi), 0, 0)^T$-direction?

$$\begin{aligned} [g_1, g_2] &= \frac{\partial g_2}{\partial x} g_1 - \frac{\partial g_1}{\partial x} g_2 \\ &= \begin{pmatrix} 0 & 0 & -\sin(\varphi+\theta) & -\sin(\varphi+\theta) \\ 0 & 0 & \cos(\varphi+\theta) & \cos(\varphi+\theta) \\ 0 & 0 & 0 & \cos(\theta) \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} - 0 \\ &= \begin{pmatrix} -\sin(\varphi+\theta) \\ \cos(\varphi+\theta) \\ \cos(\theta) \\ 0 \end{pmatrix} =: g_3 = \text{“wriggle”} \end{aligned}$$

Once More

$$\begin{aligned} [g_3, g_2] &= \frac{\partial g_2}{\partial x} g_3 - \frac{\partial g_3}{\partial x} g_2 = \dots \\ &= \begin{pmatrix} -\sin(\varphi) \\ \cos(\varphi) \\ 0 \\ 0 \end{pmatrix} = \text{“sideways”} \end{aligned}$$

The motion $[g_3, g_2]$ takes the car sideways.



The Parking Theorem

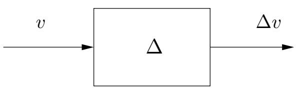
You can get out of any parking lot that is bigger than your car. Use the following control sequence:

Wriggle, Drive, –Wriggle(this requires a cool head), –Drive (repeat).

Outline

- ◦ Gain scheduling
- ◦ Internal model control
- ◦ Model predictive control
- ◦ Nonlinear observers
- ◦ Lie brackets
- • **Extra: Integral quadratic constraints**

Integral Quadratic Constraint



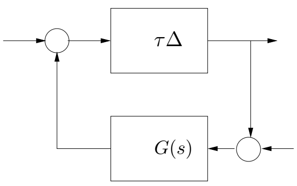
The (possibly nonlinear) operator Δ on $\mathbf{L}_2^m[0,\infty)$ is said to *satisfy the IQC defined by* Π if

$$\int_{-\infty}^{\infty} \begin{bmatrix} \widehat{v}(i\omega) \\ \widehat{(\Delta v)}(i\omega) \end{bmatrix}^* \Pi(i\omega) \begin{bmatrix} \widehat{v}(i\omega) \\ \widehat{(\Delta v)}(i\omega) \end{bmatrix} d\omega \geq 0$$

for all $v \in \mathbf{L}_2[0,\infty)$.

Δ structure	$\Pi(i\omega)$	Condition
Δ passive	$\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$	
$\|\Delta(i\omega)\| \leq 1$	$\begin{bmatrix} x(i\omega)I & 0 \\ 0 & -x(i\omega)I \end{bmatrix}$	$x(i\omega) \geq 0$
$\delta \in [-1,1]$	$\begin{bmatrix} X(i\omega) & Y(i\omega) \\ Y(i\omega)^* & -X(i\omega) \end{bmatrix}$	$X = X^* \geq 0$ $Y = -Y^*$
$\delta(t) \in [-1,1]$	$\begin{bmatrix} X & Y \\ Y^T & -X \end{bmatrix}$	
$\Delta(s) = e^{-\theta s} - 1$	$\begin{bmatrix} x(i\omega)\rho(\omega)^2 & 0 \\ 0 & -x(i\omega) \end{bmatrix}$	$\rho(\omega) =$ $2\max_{\lvert\theta\rvert\leq\theta_0} \sin(\theta\omega/2)$

IQC Stability Theorem



Let $G(s)$ be stable and proper and let Δ be causal.

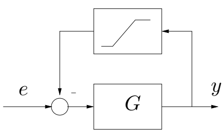
For all $\tau \in [0,1]$, suppose the loop is well posed and $\tau\Delta$ satisfies the IQC defined by $\Pi(i\omega)$. If

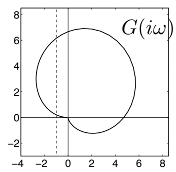
$$\begin{bmatrix} G(i\omega) \\ I \end{bmatrix}^* \Pi(i\omega) \begin{bmatrix} G(i\omega) \\ I \end{bmatrix} < 0 \quad \text{for } \omega \in [0,\infty]$$

then the feedback system is input/output stable.

A Matlab toolbox for system analysis

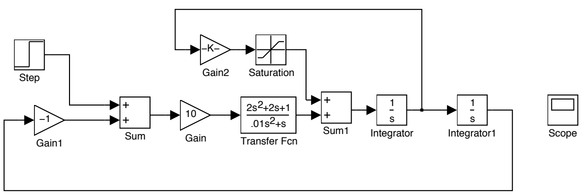
`http://www.ee.mu.oz.au/staff/cykao/`



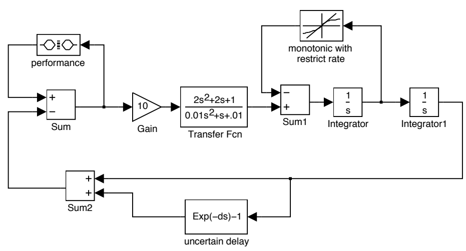



```
>> abst_init_iqc;
>> G = tf([10 0 0],[1 2 2 1]);
>> e = signal
>> w = signal
>> y = -G*(e+w)
>> w==iqc_monotonic(y)
>> iqc_gain_tbx(e,y)
```

A servo with friction

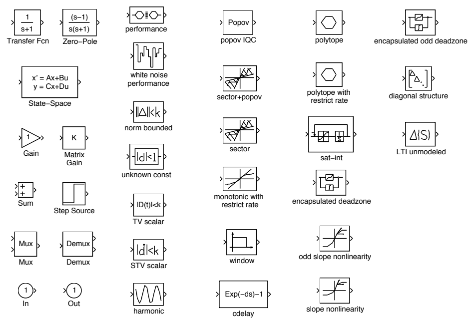


An analysis model defined graphically



```
iqc_gui('fricSYSTEM')

extracting information from fricSYSTEM ...

    scalar inputs:  5
    states:         10
    simple q-forms: 7

    LMI #1    size = 1    states: 0
    LMI #2    size = 1    states: 0
    LMI #3    size = 1    states: 0
    LMI #4    size = 1    states: 0
    LMI #5    size = 1    states: 0

  Solving with 62 decision variables ...

  ans =    4.7139
```

A library of analysis objects



The friction example in text format

```
d=signal;                          % disturbance signal
e=signal;                          % error signal
w1=signal;                         % friction force
w2=signal;                         % delay perturbation
u=signal;                          % control force
v=tf(1,[1 0])*(u-w1)               % velocity
x=tf(1,[1 0])*v;                   % position
e==d-x-w2;
u==10*tf([2 2 1],[0.01 1 0.01])*e;
w1==iqc_monotonic(v,0,[1 5],10)
w2==iqc_cdelay(x,.01)
iqc_gain_tbx(d,e)
```

Summary

- Gain scheduling
- Internal model control
- Model predictive control
- Nonlinear observers
- Lie brackets
- Extra: Integral quadratic constraints

Next: Lecture 14

- Course Summary