LUNDS UNIVERSITET

Lecture 11

FRTN10 Multivariable Control

Automatic Control LTH, 2019

Course Outline

L1–L5 Specifications, models and loop-shaping by hand

L6–L8 Limitations on achievable performance

L9–L11 Controller optimization: analytic approach

9. Linear-quadratic control
10. Kalman filtering
11. **LQG control**

L12–L14 Controller optimization: numerical approach

L15 Course review

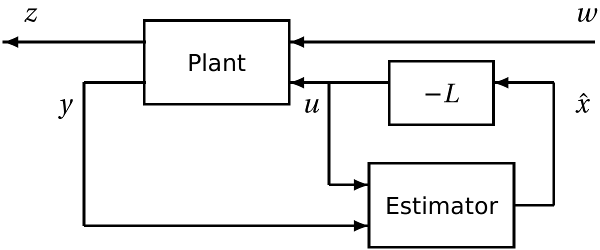
L11: LQG control

1. LQG
2. Tuning the LQG controller
3. Robustness of LQG
4. Integral action, reference values

L11: LQG control

1. LQG
2. Tuning the LQG controller
3. Robustness of LQG
4. Integral action, reference values

Optimal output feedback – LQG



Plant: $\begin{cases} \dot{x} = Ax + Bu + w_1 \\ y = Cx + w_2 \end{cases}$ $\quad \Phi_w = \begin{pmatrix} R_1 & R_{12} \\ R_{12}^T & R_2 \end{pmatrix}$

Controller: $\begin{cases} \frac{d}{dt}\hat{x}(t) = A\hat{x}(t) + Bu(t) + K[y(t) - C\hat{x}(t)] \\ u(t) = -L\hat{x}(t) \end{cases}$

Minimize$_{L,K}$ $\quad \mathrm{E}\,|z|^2 = \mathrm{E}\left(x^T Q_1 x + 2x^T Q_{12} u + u^T Q_2 u\right)$

The separation principle

In LQG control, the optimal state feedback and the optimal observer are independent and can be designed separately.

- The optimal state feedback gain L is independent of the state uncertainty
  - $\tilde{x}$ is zero-mean Gaussian $\Rightarrow$ using $u = -L\hat{x}$ is optimal ("certainty equivalance")
- The optimal Kalman filter gain K is independent of the control objective
  - From the Kalman filter's view, u is deterministic and does not affect the state uncertainty

LQG – summary

Given white noise $\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$ with intensity $\begin{pmatrix} R_1 & R_{12} \\ R_{12}^T & R_2 \end{pmatrix}$ and the linear plant

$$\dot{x}(t) = Ax(t) + Bu(t) + w_1(k)$$
$$y(t) = Cx(t) + w_2(t)$$

consider controllers of the form

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K\big(y(t) - C\hat{x}(t)\big)$$
$$u(t) = -L\hat{x}(t)$$

The stationary variance

$$\mathrm{E}\left(x^T Q_1 x + 2x^T Q_{12} u + u^T Q_2 u\right)$$

is minimized when

$$L = Q_2^{-1}(SB + Q_{12})^T \qquad K = (PC^T + R_{12})R_2^{-1}$$
$$0 = Q_1 + A^T S + SA - (SB + Q_{12})Q_2^{-1}(SB + Q_{12})^T$$
$$0 = R_1 + AP + PA^T - (PC^T + R_{12})R_2^{-1}(PC^T + R_{12})^T$$

The minimal variance is $\mathrm{tr}(SR_1) + \mathrm{tr}[PL^T(B^T SB + Q_2)L]$

Example – LQG control of an integrator

Consider the problem to minimize $\mathrm{E}(Q_1 x^2 + Q_2 u^2)$ for

$$\begin{cases} \dot{x}(t) = u(t) + w_1(t) \\ y(t) = x(t) + w_2(t) \end{cases} \qquad \Phi_w = \begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix}$$

The observer-based controller

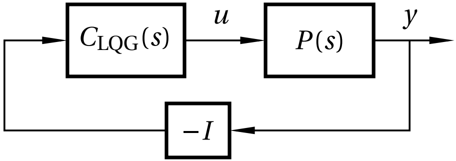
$$\begin{cases} \frac{d}{dt}\hat{x}(t) = A\hat{x}(t) + Bu(t) + K[y(t) - C\hat{x}(t)] \\ u(t) = -L\hat{x}(t) \end{cases}$$

is optimal with K and L computed as follows:

$$0 = Q_1 - S^2/Q_2 \quad \Rightarrow \quad S = \sqrt{Q_1 Q_2} \quad \Rightarrow \quad L = S/Q_2 = \sqrt{Q_1/Q_2}$$

$$0 = R_1 - P^2/R_2 \quad \Rightarrow \quad P = \sqrt{R_1 R_2} \quad \Rightarrow \quad K = P/R_2 = \sqrt{R_1/R_2}$$

The LQG controller



The controller transfer function (from $-y$ to u) is given by

$$C_{\mathrm{LQG}}(s) = L(sI - A + BL + KC)^{-1}K$$

- Same order as the plant model

Several options in Matlab:

- `L = lqr(..), K = lqe(..), C_LQG = reg(P,L,K)`
- `L = lqr(..), obs = kalman(..), C_LQG = lqgreg(obs,L)`
- `C_LQG = lqg(P,Q,R)`

L11: LQG control

1. LQG
2. Tuning the LQG controller
3. Robustness of LQG
4. Integral action, reference values

How to choose the cost function

- Only in rare instances does a quadratic cost function follow directly from the design specifications
- In most cases, the cost function must be iteratively tuned by the designer to achieve the desired closed-loop behavior

How to choose the cost function

- Only in rare instances does a quadratic cost function follow directly from the design specifications
- In most cases, the cost function must be iteratively tuned by the designer to achieve the desired closed-loop behavior

Some possible starting points:

- Only penalize the outputs $y = Cx$ and the inputs u; put $Q_1 = C^T C$, $Q_2 = \rho I$, and $Q_{12} = 0$
- Make the diagonal elements equal to the inverse value of the square of the allowed deviations:

$$Q_1 = \begin{pmatrix} \frac{1}{(x_1^{\max})^2} & \cdots & 0 \\ \vdots & \ddots & \\ 0 & & \frac{1}{(x_n^{\max})^2} \end{pmatrix}, \; Q_2 = \begin{pmatrix} \frac{1}{(u_1^{\max})^2} & \cdots & 0 \\ \vdots & \ddots & \\ 0 & & \frac{1}{(u_m^{\max})^2} \end{pmatrix}, \; Q_{12} = 0$$

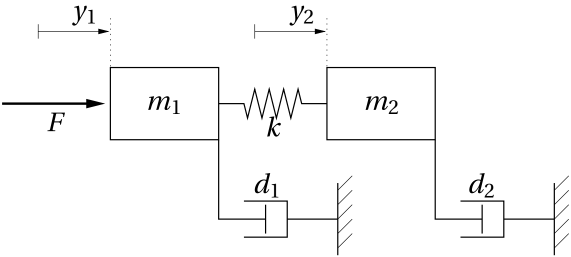
Tuning the cost function

- To achieve higher bandwidth (more aggressive control), decrease Q_2 or increase Q_1
- To increase the damping of a state x_j, add penalty on $\dot{x}_j^2$
- (Advanced) To make a state x_j behave more like $\dot{x}_j = -\alpha x_j$, add penalty on $(\dot{x}_j + \alpha x_j)^2$

Note that

$$
\begin{aligned}
\dot{x}_j^2 &= (A_j x + B_j u)^T (A_j x + B_j u) \\
&= x^T (A_j^T A_j) x + 2x^T (A_j^T B_j) u + u^T (B_j^T B_j) u
\end{aligned}
$$

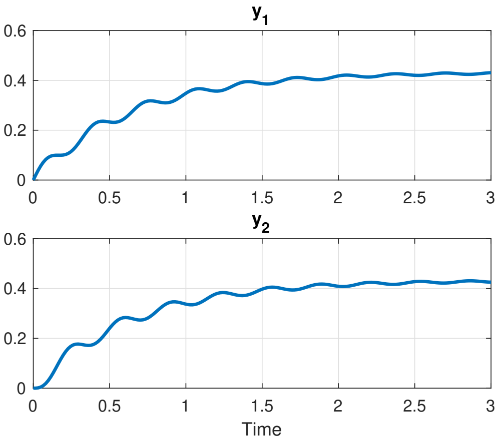
Example – LQ control of flexible servo



$$
\begin{aligned}
m_1 \frac{d^2 y_1}{dt^2} &= -d_1 \frac{dy_1}{dt} - k(y_1 - y_2) + F(t) \\
m_2 \frac{d^2 y_2}{dt^2} &= -d_2 \frac{dy_2}{dt} + k(y_1 - y_2)
\end{aligned}
$$

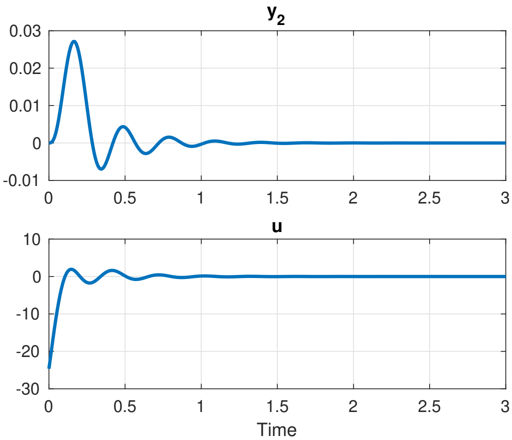
Open-loop response

Response to impulse input disturbance:



First iteration

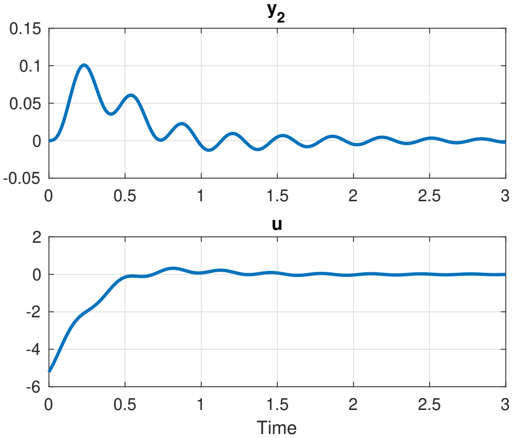
Minimize $\mathrm{E}(y_2^2 + u^2) = \mathrm{E}(x^T(C_2^T C_2)x + u^2)$



Too fast—control signal too aggressive

Second iteration

Minimize $\mathrm{E}\,(x^T(C_2^TC_2)x + 100u^2)$

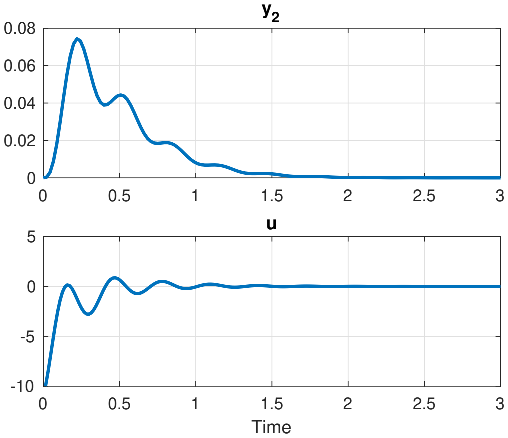


Good speed, needs improved damping

Third iteration

Minimize

$$\mathrm{E}\,(y_2^2 + 0.1\,\dot{y}_2^2 + 100u^2) = \mathrm{E}\,(x^T(C_2^T C_2 + 0.1(C_2 A)^T(C_2 A))x + 100u^2)$$



Better damping, but more aggressive control signal

Tuning the Kalman filter

- The real noise properties are seldom known
- As a starting point put $R_1 = BB^T$, $R_2 = \rho I$, $R_{12} = 0$
- If the controller is too sensitive to measurement noise, increase R_2 or decrease R_1
- If the robustness of the closed loop degrades too much when using the Kalman filter for output feedback, decrease R_2 or increase R_1

L11: LQG control

1. LQG
2. Tuning the LQG controller
3. **Robustness of LQG**
4. Integral action, reference values

Robustness of LQG controllers

Guaranteed Margins for LQG Regulators

JOHN C. DOYLE

Abstract—There are none.

INTRODUCTION

Considerable attention has been given lately to the issue of robustness of linear–quadratic (LQ) regulators. The recent work by Safonov and Athans [1] has extended to the multivariable case the now well-known guarantee of 60° phase and 6 dB gain margin for such controllers. However, for even the single-input, single-output case there has remained the question of whether there exist any guaranteed margins for the full LQG (Kalman filter in the loop) regulator. By counterexample, this note answers that question; there are none.

[*IEEE Transactions on Automatic Control*, 23:4, 1978]

Example (Doyle & Stein, 1979)

Benign minimum-phase SISO plant (no fundamental limitations):

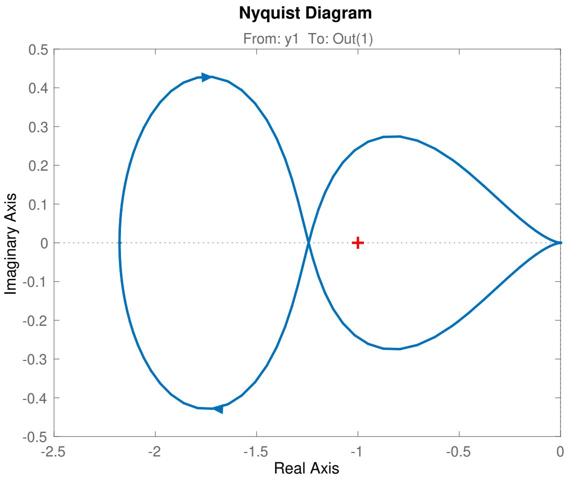
$$A = \begin{pmatrix} -4 & -3 \\ 1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad G = \begin{pmatrix} 61 \\ -35 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 2 \end{pmatrix}$$

$$Q_1 = 80 \begin{pmatrix} 1 & \sqrt{35} \\ \sqrt{35} & 35 \end{pmatrix}, \quad Q_2 = 1, \quad R_1 = 1, \quad R_2 = 1$$

gives

- Control poles: $-7 \pm 2i$
- Observer poles: $-7.02 \pm 1.95i$

Example (Doyle & Stein, 1979)



$$M_s = 4.8,\ \varphi_m = 14.8^\circ$$

Loop transfer recovery

The robustness of an LQG controller can often be improved by either

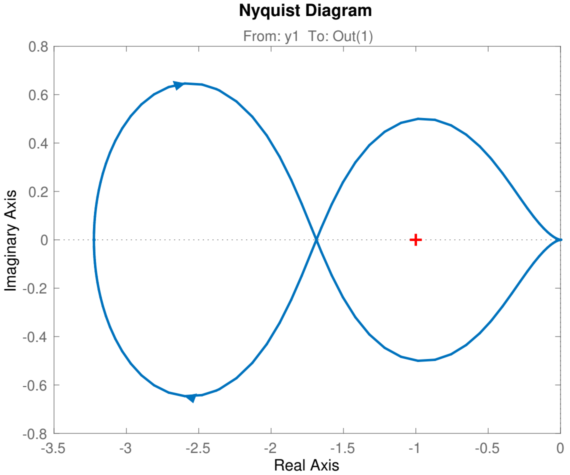
- adding a penalty proportional to $C^T C$ to Q_1
- adding a penalty proportional to BB^T to R_1

Makes the loop transfer function more similar to the state feedback (LQ) loop gain

Price: Higher controller gain, more amplification of noise

Doyle & Stein's example with LTR

$$Q_1^{new} = Q_1 + 400C^T C$$



$$M_s = 2.0,\ \varphi_m = 30°$$

L11: LQG control

Integral action via explicit integration

Add explicit integrators $\dot{x}_i = r - y$ to track reference values without error.

Gives extended plant model

$$\begin{pmatrix} \dot{x} \\ \dot{x}_i \end{pmatrix} = \begin{pmatrix} A & 0 \\ -C & 0 \end{pmatrix} \begin{pmatrix} x \\ x_i \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} u + \begin{pmatrix} 0 \\ I \end{pmatrix} r + \begin{pmatrix} I \\ 0 \end{pmatrix} v_1$$

Extended state feedback law from LQ design:

$$u = -\begin{pmatrix} L & L_i \end{pmatrix} \begin{pmatrix} x \\ x_i \end{pmatrix}$$

Including a penalty on x_i in the LQ design makes $y \to r$ in case of a constant load disturbance or step reference change

(Matlab: `lqi`, `lqgtrack`, `lqg`)

Linear-quadratic-integral control in Matlab

```
lqi computes an optimal state-feedback control law for the tracking
loop shown below. For a plant SYS with state-space equations
dx/dt = Ax + Bu, y = Cx + Du, the state-feedback control is of the form
u = - K [x; xi] where xi is the integrator output.

                                 .--------------------.
                              x  |    .---.           | x
                                 '--->|   |           |
         e = r-y  .----------.        |-K |       .---'---.
r ---->O----------|Integrator|------->|   |------->|  SYS  |-----> y
       ^          '----------' xi     '---'  u    '-------'  |
       |-                                                    |
       |                                                     |
       '-----------------------------------------------------'


[K,S,E] = lqi(SYS,Q,R,N) calculates the optimal gain matrix K given a
state-space model SYS of the plant and weighting matrices Q,R,N. The
control law u = -K z = -K [x;xi] minimizes the cost function

          J(u) = Integral {z'Qz + u'Ru + 2*z'Nu}
```

Reference handling without integration

Simple solution using feedforward from r:

$$u(t) = -L\hat{x}(t) + L_r r(t)$$

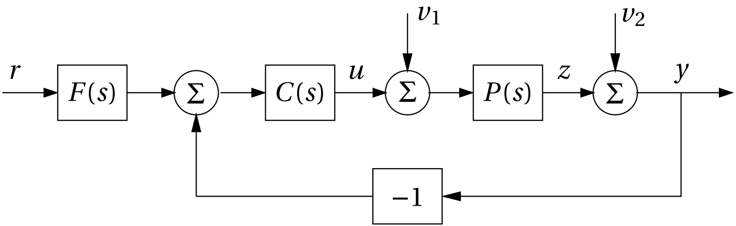
Assuming we want to achieve $y = r$, select (for a square plant)

$$L_r = [C(BL - A)^{-1}B]^{-1}$$

to ensure static gain I from r to y

A reference filter to further shape $G_{yr}(s)$ can be added if needed

LQG example – Control of DC-servo



Process: $P(s) = \dfrac{20}{s(s+1)}$

Cost function: $J = \mathrm{E}\,(z^2 + u^2)$

White noise intensities: $R_1 = 1$, $R_2 = 1$, $R_{12} = 0$

LQG design

State-space model (ignoring r):

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \overbrace{\begin{bmatrix} 0 & 0 \\ 1 & -1 \end{bmatrix}}^{A} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \overbrace{\begin{bmatrix} 20 \\ 0 \end{bmatrix}}^{B} u + \overbrace{\begin{bmatrix} 20 \\ 0 \end{bmatrix}}^{G} v_1$$

$$y = \underbrace{\begin{bmatrix} 0 & 1 \end{bmatrix}}_{C} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + v_2 \qquad\qquad z = x_2$$

Cost matrices:

$$Q_1 = C^T C = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad Q_2 = 1$$

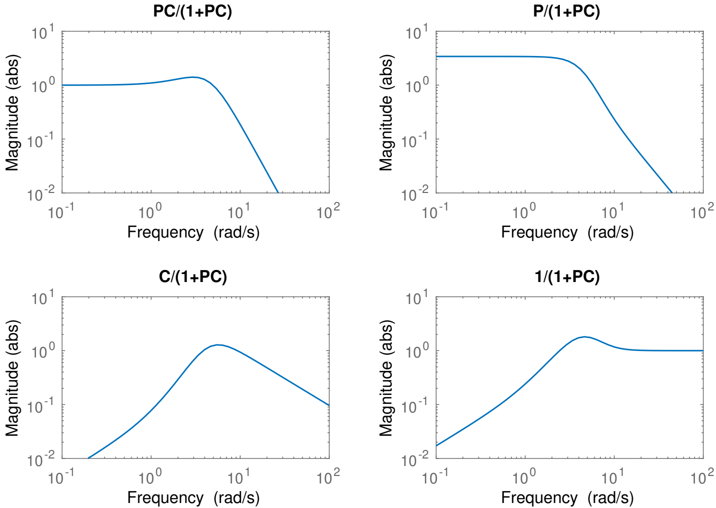
Solving the Riccati equations gives the optimal controller

$$\dot{\hat{x}} = (A - BL - KC)\hat{x} + Ky \qquad\qquad u = -L\hat{x}$$

where

$$L = \begin{bmatrix} 0.2702 & 0.7298 \end{bmatrix} \qquad\qquad K = \begin{bmatrix} 20.0000 \\ 5.4031 \end{bmatrix}$$

Gang of four



$\frac{P}{1+PC}$ reveals poor low-frequency disturbance rejection

Integral action

Add explicit integrator $\dot{x}_i = r - y$ and extend the model:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_i \end{bmatrix} = \overbrace{\begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & -1 & 0 \end{bmatrix}}^{A_e} \begin{bmatrix} x_1 \\ x_2 \\ x_i \end{bmatrix} + \overbrace{\begin{bmatrix} 20 \\ 0 \\ 0 \end{bmatrix}}^{B_e} u + \overbrace{\begin{bmatrix} 20 \\ 0 \\ 0 \end{bmatrix}}^{G_e} v_1$$

Minimization of $\mathrm{E}\left(x_2^2 + 0.01 x_i^2 + u^2\right)$ gives the optimal state feedback

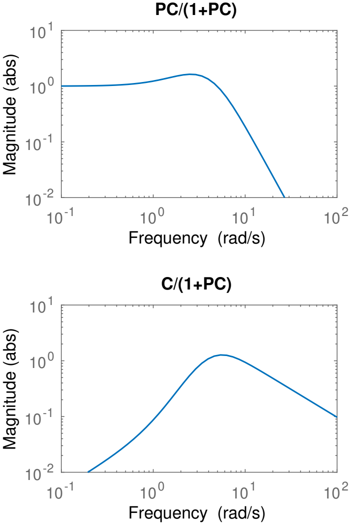
$$u = -L_e \begin{bmatrix} \hat{x} & x_i \end{bmatrix}$$

where

$$L_e = \begin{bmatrix} 0.2751 & 0.7569 & -0.1 \end{bmatrix}$$

We can use the same Kalman filter as before (x_i is known)

Gang of four with integral action



Matlab code for DC-servo example

```
A = [0 0; 1 -1];
B = [20; 0];
G = [20; 0];
C = [0 1];
sys = ss(A,B,C,0);
Q1 = C'*C;
Q2 = 1;
R1 = 1;
R2 = 1;

%% Design LQG controller
L = lqr(A,B,Q1,Q2)        % Calculate LQ feedback gain
K = lqe(A,G,C,R1,R2)      % Calculate Kalman gain
ctrl = -reg(sys,L,K);     % Form LQG regulator
```

Matlab code for DC-servo example, cont'd

```
%% Design LQG controller with integral action, version 1
Qi = 1;                               % Integral state weight
Le = lqi(sys,blkdiag(Q1,Qi),Q2)      % Calculate LQI feedback gain
sysk = ss(A,[B G],C,[0 0]);           % Define system with noise
obs = kalman(sysk,R1,R2);             % Calculate Kalman filter
ctrl_i1 = lqgtrack(obs,Le)            % Form LQG regulator


%% Design LQG controller with integral action, version 2
Qi = 1;
ctrl_i2 = lqg(sys,blkdiag(Q1,Q2),blkdiag(G*R1*G',R2),Qi)
```

Summary of LQG

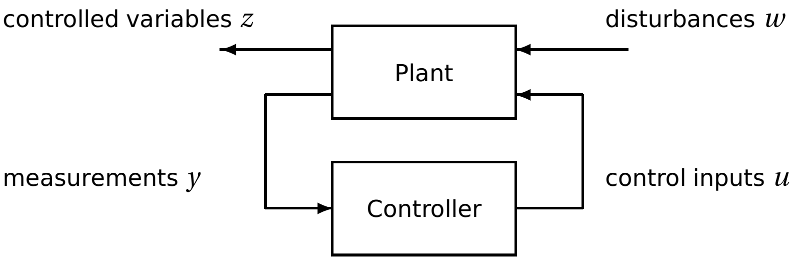
Advantages

- Works fine with MIMO models
- Observer structure ties to reality
- Always stabilizing
- Well developed theory, analytic solutions

Disadvantages

- Requires detailed state-space model
- Potentially high-order controllers (same order as plant model)
- Sometimes hard to choose design weights
- No robustness guarantees – must always check the resulting controller!
- Quadratic criterion (H_2) not always the most suitable design requirement

Alternative norms for optimization



Common alternative: H_∞ optimal control:

$$\text{Minimize} \quad \sup_{\omega} \|G_{zw}(i\omega)\|$$

Can be solved using a couple of Riccati equations, similar to the LQG problem (Matlab: `hinfsyn`)

Lecture 11 – summary

- LQG design can produce a stabilizing controller for any controllable and observable linear MIMO plant
- Cost function and noise model must be tuned to obtain the desired closed-loop performance
  - Remember to respect fundamental limitations
- No robustness guarantees – always check the result!

Next section of the course: Optimization of controllers using numerical methods