Contents

A1    Syllabus, Spring 2014
0     Introduction
1     Dynamic Programming
3     Probability Theory
4     Least squares (LS) estimation
5     Stochastic state estimation (Kalman Filter)
6     Linear Quadratic Gaussian (LQG) Control
7     Principles of Feedback Design
8     Discretization and Implementation of Continuous-time Design
9     LQG/Loop Transfer Recovery (LTR)
10    LQ with Frequency Shaped Cost Function (FSLQ)
11    Feedforward Control: Zero Phase Error Tracking
12    Preview Control
13    Internal Model Principle and Repetitive Control
14    Disturbance Observer
15    System Identification and Recursive
16    Stability of Parameter Adaptation Algorithms
17    PAA with Parallel Predictors
18    Parameter Convergence in PAAs
19    Adaptive Control based on Pole Assignment
ME 233: Advanced Control Systems II
Spring 2014

ME233 discusses advanced control methodologies and their applications to engineering systems. Methodologies include but are not limited to: Linear Quadratic Optimal Control, Kalman Filter, Discretization, Linear Quadratic Gaussian Problem, Loop Transfer Recovery, System Identification, Adaptive Control and Model Reference Adaptive Systems, Self Tuning Regulators, Repetitive Control, and Disturbance Observers.

Instructor: Xu Chen, maxchen@berkeley.edu
Office: 5112 Etcheverry Hall
Office Hour: Tu, Th 1:00pm – 2:30pm in 5112 Etcheverry Hall

Teaching Assistant: Changliu Liu, changliuliu@berkeley.edu
Office: M, W 10:00am – 11:00am in 136 Hesse Hall

Lectures: Tu, Th 8:00am - 9:30pm in Rm. 3113 Etcheverry Hall
Discussion: Fri. 10am-11am in Rm 1165 Etcheverry Hall

Prerequisites: ME C 232 (syllabus on course website) or its equivalence

Course website: http://www.me.berkeley.edu/ME233/sp14/ and bCourses.berkeley.edu

Remark: lecture videos are webcasted to Youtube and iTunes-U (links on the course website)

Grading: Two Midterm Exams (open one-page summary sheet for each exam) 2*20 %
Final Examination (open notes) 40 %
Homework (see policy on course website) 20 %

Class Notes: ME233 Class Notes by M. Tomizuka (Parts I and II)
They can be purchased at Copy Central, 48 Shattuck Square, Berkeley

Tentative Schedule (Subject to change):

<table>
<thead>
<tr>
<th>Week</th>
<th>Days</th>
<th>Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>2/4, 2/6</td>
<td>Principle of Least Squares estimation; Stochastic State Estimation (Kalman Filter).</td>
</tr>
<tr>
<td>4</td>
<td>2/11, 2/13</td>
<td>Stochastic Estimation (continuation)</td>
</tr>
<tr>
<td>5</td>
<td>2/18, 2/20</td>
<td>Linear Stochastic Control (Linear Quadratic Gaussian (LQG) Problem); Singular values; Introduction to linear multivariable control.</td>
</tr>
<tr>
<td>6</td>
<td>2/25, 2/27</td>
<td>Linear multivariable control; Loop Transfer Recovery</td>
</tr>
<tr>
<td>7</td>
<td>3/4, 3/6</td>
<td>Frequency-shaped LQ; in-class Midterm I on 3/4/2014</td>
</tr>
<tr>
<td>8</td>
<td>3/11, 3/13</td>
<td>Feedforward and preview control; Internal Model Principle and Repetitive Control.</td>
</tr>
<tr>
<td>9</td>
<td>3/18, 3/20</td>
<td>Disturbance Observer</td>
</tr>
<tr>
<td></td>
<td>3/25, 3/27</td>
<td>SPRING RECESS</td>
</tr>
<tr>
<td>10</td>
<td>4/1, 4/3</td>
<td>System Identification and Adaptive Control</td>
</tr>
<tr>
<td>11</td>
<td>4/8, 4/10</td>
<td>Parameter Estimation Algorithms</td>
</tr>
<tr>
<td>12</td>
<td>4/15, 4/17</td>
<td>Stability analysis of adaptive systems; in-class Midterm II on 4/15/2014</td>
</tr>
<tr>
<td>13</td>
<td>4/22, 4/24</td>
<td>Parallel Adaptation Algorithms; Parameter Convergence</td>
</tr>
<tr>
<td>14</td>
<td>4/29, 5/1</td>
<td>Direct and Indirect Adaptive Control; Adaptive Prediction</td>
</tr>
</tbody>
</table>

Final Examination: May 15 (Th) 2014, 7-10 pm

Please notify the instructor in writing by the second week of the semester, if you have any potential conflict(s) about the class schedule, or if you need special accommodations such as: disability-related accommodations, emergency medical information you wish to discuss with the instructor, or special arrangements in case the building must be evacuated.
Introduction

Big picture

ME 233 talks about advanced and practical control theories, including but not limited to:

▶ dynamic programming
▶ optimal estimation (Kalman Filter) and stochastic control
▶ SISO and MIMO feedback design principles
▶ digital control: implementation and design
▶ feedforward design techniques: preview control, zero phase error tracking, etc
▶ feedback design techniques: LQG/LTR, internal model principle, repetitive control, disturbance observer
▶ system identification
▶ adaptive control
▶ ...

Syllabus

Requirements
Teaching staff and class notes

▶ instructor:
  ▶ Xu Chen, 2013 UC Berkeley Ph.D., maxchen@berkeley.edu
  ▶ office hour: Tu Thur 1pm-2:30pm at 5112 Etcheverry Hall

▶ teaching assistant:
  ▶ Changliu Liu, changliuliu@berkeley.edu
  ▶ office hour: M, W 10:00am – 11:00am in 136 Hesse Hall

▶ class notes:
  ▶ ME233 Class Notes by M. Tomizuka (Parts I and II); Both can be purchased at Copy Central, 48 Shattuck Square, Berkeley

Requirements and evaluations

▶ website (case sensitive):
  ▶ www.me.berkeley.edu/ME233/sp14
  ▶ bcourses.berkeley.edu

▶ prerequisites: ME C 232 or its equivalence
▶ lectures: Tu Thur 8-9:30am, 3113 Etcheverry Hall
▶ discussions: Fri. 10-11am, 1165 Etcheverry Hall
▶ homework (20%)
▶ two in-class midterms (20% each): Mar. 4, 2014 and Apr. 15, 2014; one-page handwritten summary sheets allowed
▶ one final exam (40%): May 15 2014 (Th), 7 pm -10 pm; open notes
Prerequisites (ME 232 table of contents)

- Laplace and Z transformations
- Models and Modeling of linear dynamical systems: transfer functions, state space models
- Solutions of linear state equations
- Stability: poles, eigenvalues, Lyapunov stability
- Controllability and observability
- State and output feedbacks, pole assignment via state feedback
- State estimation and observer, observer state feedback control
- Linear Quadratic (LQ) Optimal Control, LQR properties, Riccati equation

Remark

ME233 will be webcasted:

- Berkeley’s YouTube channel (http://www.youtube.com/ucberkeley)
- iTunes U (http://itunes.berkeley.edu/)
- webcast.berkeley (http://webcast.berkeley.edu)

links will be posted on course website when available
References (also on course website)

- **Probability**
  - Bertsekas, Introduction to Probability, Athena Scientific
  - Yates and Goodman, Probability and Stochastic Processes, second edition, Willey

- **Linear Quadratic Optimal Control**
  - Bryson and Ho, Applied Optimal Control: Optimization, Estimation, and Control, Wiley

- **Stochastic Control Theory and Optimal Filtering**
  - Lewis and Xie and Popa, Optimal and Robust Estimation, Second Edition CRC
  - Grewal and Andrews, Kalman Filter, Theory and Practice, Prentice Hall
  - Astrom, Introduction to Stochastic Control Theory, Dover Books on Engineering (paperback), New York, 2006

- **Adaptive Control**
  - Astrom and Wittenmark, Adaptive Control, Addison Wesley, 2nd Ed., 1995
  - Goodwin and Sin, Adaptive Filtering Prediction and Control, Prentice Hall, 1984
  - Krstic, Kanellakopoulos, and Kokotovic, Nonlinear and Adaptive Control Design, Willey
Lecture 1: Dynamic Programming

Dynamic programming (DP)

introduction:

- history: developed in the 1950’s by Richard Bellman
- “programming”: ~“planning” (has nothing to do with computers)
- a useful concept with lots of applications
- IEEE Global History Network: “A breakthrough which set the stage for the application of functional equation techniques in a wide spectrum of fields..."
Essentials of dynamic programming

- key idea: solve a complex and difficult problem via solving a collection of sub problems

Example (Path planning)
goal: obtain minimum cost path from $S$ to $E$

- observation: if node $C$ is on the optimal path, then the path from node $C$ to node $E$ must be optimal as well
solution: backward analysis

\[
\begin{align*}
\text{dist}(E) &= \min \{ \text{dist}(B) + 2, \text{dist}(D) + 1 \} \\
\text{dist}(B) &= \text{dist}(A) + 6 \\
\text{dist}(D) &= \min \{ \text{dist}(B) + 1, \text{dist}(C) + 3 \} \\
\text{dist}(C) &= 2 \\
\text{dist}(A) &= \min \{ 1, \text{dist}(C) + 4 \}
\end{align*}
\]
Essentials of dynamic programming

summary (Bellman’s principle of optimality): “From any point on an optimal trajectory, the remaining trajectory is optimal for the corresponding problem initiated at that point.”

General optimal control problems

general discrete-time plant:

\[ x(k+1) = f(x(k), u(k), k) \]

state constraint: \( x(k) \in X \subset \mathbb{R}^n \)
input constraint: \( u(k) \in U \subset \mathbb{R}^m \)

performance index:

\[ J = S(x(N)) + \sum_{k=0}^{N-1} L(x(k), u(k), k) \]

\( S \& L \)-real, scalar-valued functions; \( N \)-final time (optimization horizon)

goal: obtain the optimal control sequence

\[ \{u^o(0), u^o(1), \ldots, u^o(N-1)\} \]
Dynamic programming for optimal control

- define: \( U_k \triangleq \{ u(k), u(k+1), \ldots, u(N-1) \} \)
- optimal cost to go at time \( k \):

\[
J^o_k (x(k)) \triangleq \min_{U_k} \left\{ S(x(N)) + \sum_{j=k}^{N-1} L(x(j), u(j), j) \right\}
\]

\[
= \min_{u(k)} \min_{U_{k+1}} \left\{ L(x(k), u(k), k) + \left[ S(x(N)) + \sum_{j=k+1}^{N-1} L(x(j), u(j), j) \right] \right\}
\]

\[
= \min_{u(k)} \left\{ L(x(k), u(k), k) + \min_{U_{k+1}} \left[ S(x(N)) + \sum_{j=k+1}^{N-1} L(x(j), u(j), j) \right] \right\}
\]

\[
= \min_{u(k)} \left\{ L(x(k), u(k), k) + J^o_{k+1} (x(k+1)) \right\}
\]

(1)

- boundary condition: \( J^o_N (x(N)) = S(x(N)) \)
- The problem can now be solved by solving a sequence of problems \( J^o_{N-1}, J^o_{N-2}, \ldots, J^o_1, J^o \).

Solving discrete-time finite-horizon LQ via DP

- system dynamics:

\[
x(k+1) = A(k)x(k) + B(k)u(k), \quad x(k_0) = x_0
\]

(2)

- performance index:

\[
J = \frac{1}{2} x^T(N) S x(N) + \frac{1}{2} \sum_{k=k_0}^{N-1} \left\{ x^T(k) Q(k) x(k) + u^T(k) R(k) u(k) \right\}
\]

\[
Q(k) = Q^T(k) \succeq 0, \quad S = S^T \succeq 0, \quad R(k) = R^T(k) \succ 0
\]

- optimal cost to go:

\[
J^o_k (x(k)) = \min_{u(k)} \left\{ \frac{1}{2} x^T(k) Q(k) x(k) + \frac{1}{2} u^T(k) R(k) u(k) + J^o_{k+1} (x(k+1)) \right\}
\]

with boundary condition: \( J^o_N (x(N)) = \frac{1}{2} x^T(N) S x(N) \)
Facts about quadratic functions

- consider
  \[ f(u) = \frac{1}{2} u^T M u + p^T u + q, \quad M = M^T \]  

- optimality (maximum when \( M \) is negative definite; minimum when \( M \) is positive definite) is achieved when
  \[ \frac{\partial f}{\partial u} = Mu + p = 0 \Rightarrow u^o = -M^{-1}p \]  

- and the optimal cost is
  \[ f^o = f(u^o) = -\frac{1}{2} p^T M^{-1}p + q \]

From \( J^o_N \) to \( J^o_{N-1} \) in discrete-time LQ

- by definition:
  \[ J^o_{N-1}(x(N-1)) = \min_{u(N-1)} \left\{ \frac{1}{2} x^T(N) S x(N) + \frac{1}{2} \left[ x^T(N-1) Q(N-1) x(N-1) + u^T(N-1) R(N-1) u(N-1) \right] \right\} \]

- using the system dynamics (2) gives
  \[ J^o_{N-1}(x(N-1)) = \frac{1}{2} \min_{u(N-1)} \left\{ x^T(N-1) Q(N-1) x(N-1) \right. \]
  \[ + u^T(N-1) R(N-1) u(N-1) + \left[ A(N-1) x(N-1) + B(N-1) u(N-1) \right]^T \]
  \[ \times S \left[ A(N-1) x(N-1) + B(N-1) u(N-1) \right] \]

- optimal control by letting \( \partial J^o_{N-1} / \partial u(N-1) = 0 \):
  \[ u^o(N-1) = - \left[ R(N-1) + B^T(N-1) S B(N-1) \right]^{-1} B^T(N-1) S A(N-1) x(N-1) \]

state feedback gain: \( K(N-1) \)
Optimality at \( N \) and \( N - 1 \)

at time \( N \): optimal cost is

\[
J^o_N(x(N)) = \frac{1}{2} x^T(N) S x(N) \triangleq \frac{1}{2} x^T(N) P(N) x(N)
\]

at time \( N - 1 \):

\[
J^o_{N-1}(x(N-1)) = \frac{1}{2} \min_{u(N-1)} \{ x^T(N-1) Q(N-1) x(N-1) 
+ u^T(N-1) R(N-1) u(N-1) + [A(N-1) x(N-1) + B(N-1) u(N-1)]^T 
\times S [A(N-1) x(N-1) + B(N-1) u(N-1)] \}
\]

optimal cost to go [by using (5)] is

\[
J^o_{N-1}(x(N-1)) = \frac{1}{2} x^T(N-1) \left\{ Q(N-1) + A^T(N-1) S A(N-1)
- (\ldots)^T \left[ R(N-1) + B^T(N-1) S B(N-1) \right]^{-1} B^T(N-1) S A(N-1) \right\} x(N-1)
\triangleq \frac{1}{2} x^T(N-1) P(N-1) x(N-1)
\]

Summary: from \( N \) to \( N - 1 \)

at \( N \):

\[
J^o_N(x(N)) = \frac{1}{2} x^T(N) S x(N) = \frac{1}{2} x^T(N) P(N) x(N)
\]

at \( N - 1 \):

\[
J^o_{N-1}(x(N-1)) = \frac{1}{2} x^T(N-1) P(N-1) x(N-1)
\]

with (\( S \) has been replaced with \( P(N) \) here)

\[
P(N-1) = Q(N-1) + A^T(N-1) P(N) A(N-1)
- (\ldots)^T \left[ R(N-1) + B^T(N-1) P(N) B(N-1) \right]^{-1} B^T(N-1) P(N) A(N-1)
\]

and state-feedback law

\[
u^o(N-1) = - \left[ R(N-1) + B^T(N-1) P(N) B(N-1) \right]^{-1} 
\times B^T(N-1) P(N) A(N-1) x(N-1)
\]
Induction from $k+1$ to $k$

- Assume at $k+1$:
  \[ J_{k+1}^o(x(k+1)) = \frac{1}{2} x^T (k+1) P(k+1) x(k+1) \]

- Analogous as the case from $N$ to $N-1$, we can get, at $k$:
  \[ J_k^o(x(k)) = \frac{1}{2} x^T (k) P(k) x(k) \]

with Riccati equation

\[
P(k) = A^T(k) P(k+1) A(k) + Q(k) \\
- A^T(k) P(k+1) B(k) \left[ R(k) + B^T(k) P(k+1) B(k) \right]^{-1} B^T(k) P(k+1) A(k) \]

and state-feedback law

\[
u^o(k) = - \left[ R(k) + B^T(k) P(k+1) B(k) \right]^{-1} B^T(k) P(k+1) A(k) x(k) \]

Implementation

- Optimal state-feedback control law:
  \[
u^o(k) = - \left[ R(k) + B^T(k) P(k+1) B(k) \right]^{-1} B^T(k) P(k+1) A(k) x(k) \]

- Riccati equation:
  \[
P(k) = A^T(k) P(k+1) A(k) + Q(k) \]
  \[
  - A^T(k) P(k+1) B(k) \left[ R(k) + B^T(k) P(k+1) B(k) \right]^{-1} B^T(k) P(k+1) A(k) \]

with the boundary condition $P(N) = S$.

- $u^o(k)$ depends on
  - the state vector $x(k)$
  - system matrices $A(k)$ and $B(k)$ and the cost matrix $R(k)$
  - $P(k+1)$, which depends on $Q(k+2)$, $A(k+1)$, $B(k+1)$, and $P(k+2)$...

- Iterating gives: $u(0)$ depends on $\{A(k), B(k), R(k), Q(k+1)\}_{k=0}^{N-1}$

In practice, $P(k)$ can be computed offline since they do not require information of $x(k)$.
Lecture 3: Review of Probability Theory

Connection with control systems
Random variable, distribution
Multiple random variables
Random process, filtering a random process

Big picture
why are we learning this:

We have been very familiar with deterministic systems:

\[ x(k+1) = Ax(k) + Bu(k) \]

In practice, we commonly have:

\[ x(k+1) = Ax(k) + Bu(k) + B_w w(k) \]

where \( w(k) \) is the noise term that we have been neglecting. With the introduction of \( w(k) \), we need to equip ourselves with some additional tool sets to understand and analyze the problem.
Sample space, events and probability axioms

- experiment: a situation whose outcome depends on chance
- trial: each time we do an experiment we call that a trial

Example (Throwing a fair dice)

possible outcomes in one trail: getting a ONE, getting a TWO, ...

- sample space $\Omega$: includes all the possible outcomes
- probability: discusses how likely things, or more formally, events, happen
- an event $S_i$: includes some (maybe 1, maybe more, maybe none) outcomes of the sample space. e.g., the event that it won’t rain tomorrow; the event that getting odd numbers when throwing a dice

Sample space, events and probability axioms

- $\Pr\{S_j\} \geq 0$
- $\Pr\{\Omega\} = 1$
- if $S_i \cap S_j = \emptyset$ (empty set), then $\Pr\{S_i \cup S_j\} = \Pr\{S_i\} + \Pr\{S_j\}$

Example (Throwing a fair dice)

the sample space:

$$\Omega = \{\text{getting a ONE, getting a TWO, \ldots, getting a SIX}\}$$

the event $S_1$ of observing an even number:

$$S_1 = \{\omega_2, \omega_4, \omega_6\}$$

$$\Pr\{S_1\} = \frac{1}{6} + \frac{1}{6} + \frac{1}{6} = \frac{1}{2}$$
Random variables

to better measure probabilities, we introduce random variables (r.v.’s)

- r.v.: a real valued function $X(\omega)$ defined on $\Omega$; $\forall x \in \mathbb{R}$ there defined the (probability) cumulative distribution function (cdf)

$$F(x) = \Pr\{X \leq x\}$$

- cdf $F(x)$: non-decreasing, $0 \leq F(x) \leq 1$, $F(-\infty) = 0$, $F(\infty) = 1$

Example (Throwing a fair dice)

can define $X$: the obtained number of the dice

$$X(\omega_1) = 1, \ X(\omega_2) = 2, \ X(\omega_3) = 3, \ X(\omega_4) = 4, \ldots$$

can also define $X$: indicator of whether the obtained number is even

$$X(\omega_1) = X(\omega_3) = X(\omega_5) = 0, \ X(\omega_2) = X(\omega_4) = X(\omega_6) = 1$$

Probability density and moments of distributions

- probability density function (pdf):

$$p(x) = \frac{dF(x)}{dx}$$

$$\Pr(a < X \leq b) = \int_a^b p(x) \, dx, \ a < b$$

sometimes we write $p_X(x)$ to emphasize that it is for the r.v. $X$

- mean, or expected value (first moment):

$$m_X = E[X] = \int_{-\infty}^{\infty} xp_X(x) \, dx$$

- variance (second moment):

$$\text{Var}[X] = E[(X - m_X)^2] = \int_{-\infty}^{\infty} (x - m_X)^2 p_X(x) \, dx$$

- standard deviation (std): $\sigma = \sqrt{\text{Var}[X]}$

- exercise: prove that $\text{Var}[X] = E[X^2] - (E[X])^2$
Example distributions

uniform distribution

▶ a r.v. uniformly distributed between \( x_{\text{min}} \) and \( x_{\text{max}} \)

▶ probability density function:

\[
p(x) = \frac{1}{x_{\text{max}} - x_{\text{min}}}
\]

▶ cumulative distribution function:

\[
F(x) = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}, \quad x_{\text{min}} \leq x \leq x_{\text{max}}
\]

▶ mean and variance:

\[
E[X] = \frac{1}{2} (x_{\text{max}} + x_{\text{min}}), \quad \text{Var}[X] = \frac{(x_{\text{max}} - x_{\text{min}})^2}{12}
\]

Example distributions

Gaussian/normal distribution

▶ importance: sum of independent r.v.s \( \rightarrow \) a Gaussian distribution

▶ probability density function:

\[
p(x) = \frac{1}{\sigma_X \sqrt{2\pi}} \exp \left( -\frac{(x - m_X)^2}{2\sigma_X^2} \right)
\]

▶ pdf fully characterized by \( m_X \) and \( \sigma_X \). Hence a normal distribution is usually denoted as \( \mathcal{N}(m_X, \sigma_X) \)

▶ nice properties: if \( X \) is Gaussian and \( Y \) is a linear function of \( X \), then \( Y \) is Gaussian
Example distributions

Gaussian/normal distribution

**Central Limit Theorem**: if $X_1, X_2, \ldots$ are independent identically distributed random variables with mean $m_X$ and variance $\sigma^2_X$, then

$$Z_n = \frac{\sum_{k=1}^{n}(X_k - m_X)}{\sqrt{n}\sigma^2_X}$$

converges in distribution to a normal random variable $X \sim N(0, 1)$

**example**: sum of uniformly distributed random variables in $[0, 1]$

X1 = rand(1,1e5);
X2 = rand(1,1e5);
X3 = rand(1,1e5);
Z = X1 + X2;
[fz,x] = hist(Z,100);
w_fz = x(end)/length(fz);
fz = fz/sum(fz)/w_fz;
figure, bar(x,fz)
xlabel 'x'; ylabel 'p_Z(x)';
Y = X1 + X2 + X3;
% ...

Lecture 3: Review of Probability Theory

Multiple random variables

joint probability

for the same sample space $\Omega$, multiple r.v.’s can be defined

- joint probability: $\Pr(X = x, Y = y)$
- joint cdf:
  $$F(x, y) = \Pr(X \leq x, Y \leq y)$$
- joint pdf: $p(x, y) = \frac{\partial^2}{\partial x \partial y} F(x, y)$
- covariance:
  $$\text{Cov}(X, Y) = \Sigma_{XY} = E[(X - m_X)(Y - m_Y)] = E[XY] - E[X] E[Y]$$
  $$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - m_X)(y - m_Y) p(x, y) \, dx \, dy$$

- uncorrelated: $\Sigma_{XY} = 0$
- independent random variables satisfy:
  $$F(x, y) = \Pr(X \leq x, Y \leq y) = \Pr(X \leq x) \Pr(Y \leq y) = F_X(x) F_Y(y)$$
  $$p(x, y) = p_X(x) p_Y(y)$$

Lecture 3: Review of Probability Theory
Multiple random variables

more about correlation
correlation coefficient:
\[ \rho(X, Y) = \frac{\text{Cov}(X, Y)}{\sqrt{\text{Var}(X)\text{Var}(Y)}} \]

\(X\) and \(Y\) are uncorrelated if \(\rho(X, Y) = 0\)

- independent \(\Rightarrow\) uncorrelated; uncorrelated \(\not\Rightarrow\) independent
- uncorrelated indicates \(\text{Cov}(X, Y) = E[XY] - E[X]E[Y] = 0\), which is weaker than \(X\) and \(Y\) being independent

**Example**

\(X\)–uniformly distributed on \([-1, 1]\). Construct \(Y\): if \(X \leq 0\) then \(Y = -X\); if \(X > 0\) then \(Y = X\). \(X\) and \(Y\) are uncorrelated due to

- \(E[X] = 0\), \(E[Y] = \frac{1}{2}\)
- \(E[XY] = 0\)

however \(X\) and \(Y\) are clearly dependent

Multiple random variables

random vector

- vector of r.v.'s:
  \[ Z = \begin{bmatrix} X \\ Y \end{bmatrix} \]

- mean:
  \[ m_Z = \begin{bmatrix} m_X \\ m_Y \end{bmatrix} \]

- covariance matrix:
  \[
  \Sigma = E[(Z - m_Z)(Z - m_Z)^T] = \begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{bmatrix}
  \]
  \[= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \begin{bmatrix} (X - m_X)^2 & (X - m_X)(Y - m_Y) \\ (Y - m_Y)(X - m_X) & (Y - m_Y)^2 \end{bmatrix} p(x,y) \, dx \, dy \]
Conditional distributions

- joint pdf to single pdf:
  \[ p_X(x) = \int_{-\infty}^{\infty} p(x, y) \, dy \]

- conditional pdf:
  \[ p_X(x | y_1) = p_X(x | Y = y_1) = \frac{p(x, y_1)}{p_Y(y_1)} \]

- conditional mean:
  \[ E[X|y_1] = \int_{-\infty}^{\infty} x p_X(x | y_1) \, dx \]

- note: independent \( \Rightarrow p_X(x | y_1) = p_X(x) \)

- properties of conditional mean:
  \[ E_Y[E[X|y]] = E[X] \]

Multiple random variables

Gaussian random vectors

Gaussian r.v. is particularly important and interesting as its pdf is mathematically sound

Special case: two independent Gaussian r.v. \( X_1 \) and \( X_2 \)

\[
p(x_1, x_2) = p_{X_1}(x_1) p_{X_2}(x_2) = \frac{1}{\sigma_{X_1} \sqrt{2\pi}} e^{-\left(\frac{x_1 - m_{X_1}}{2\sigma_{X_1}^2}\right)^2} \frac{1}{\sigma_{X_2} \sqrt{2\pi}} e^{-\left(\frac{x_2 - m_{X_2}}{2\sigma_{X_2}^2}\right)^2} = \frac{1}{\sigma_{X_1} \sigma_{X_2} (\sqrt{2\pi})^2} \exp\left\{ -\frac{1}{2} \begin{bmatrix} x_1 - m_{X_1} \\ x_2 - m_{X_2} \end{bmatrix}^T \begin{bmatrix} \sigma_{X_1}^2 & 0 \\ 0 & \sigma_{X_2}^2 \end{bmatrix}^{-1} \begin{bmatrix} x_1 - m_{X_1} \\ x_2 - m_{X_2} \end{bmatrix} \right\}
\]

We can use the random vector notation: \( X = [X_1, X_2]^T \)

\[
\Sigma = \begin{bmatrix} \sigma_{X_1}^2 & 0 \\ 0 & \sigma_{X_2}^2 \end{bmatrix}
\]

and write

\[
p_X(x) = \frac{1}{(\sqrt{2\pi})^2 \sqrt{\det \Sigma}} \exp\left\{ -\frac{1}{2} [X - m_X]^T \Sigma^{-1} [X - m_X] \right\}
\]
General Gaussian random vectors

pdf for a $n$-dimensional jointly distributed Gaussian random vector $X$:

$$p_X(x) = \frac{1}{(\sqrt{2\pi})^n \sqrt{\det \Sigma}} \exp \left\{ -\frac{1}{2} [X - m_X]^T \Sigma^{-1} [X - m_X] \right\}$$ (1)

joint pdf for 2 Gaussian random vectors $X$ ($n$-dimensional) and $Y$ ($m$-dimensional):

$$p(x, y) = \frac{1}{(\sqrt{2\pi})^{n+m} \sqrt{\det \Sigma}} \exp \left\{ -\frac{1}{2} \begin{bmatrix} x - m_X \\ y - m_Y \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} x - m_X \\ y - m_Y \end{bmatrix} \right\}$$ (2)

$$\Sigma = \begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{bmatrix}$$

where $\Sigma_{XY}$ is the cross covariance (matrix) between $X$ and $Y$

$$\Sigma_{XY} = E[(X - m_X)(Y - m_Y)^T] = E[(Y - m_Y)(X - m_X)^T]^T = \Sigma_{YX}^T$$

conditional mean and covariance

important facts about conditional mean and covariance:

$$m_{X|Y} = m_X + \Sigma_{XY} \Sigma_{YY}^{-1} [y - m_Y]$$

$$\Sigma_{X|Y} = \Sigma_{XX} - \Sigma_{XY} \Sigma_{YY}^{-1} \Sigma_{YX}$$

proof uses $p(x, y) = p(x|y) p(y)$, (1), and (2)

▶ getting $\det \Sigma$ and the inverse $\Sigma^{-1}$: do a transformation

$$\begin{bmatrix} I & -\Sigma_{XY} \Sigma_{YY}^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{bmatrix} \begin{bmatrix} I \\ -\Sigma_{YY}^{-1} \Sigma_{YX} \end{bmatrix} = \begin{bmatrix} \Sigma_{XX} - \Sigma_{XY} \Sigma_{YY}^{-1} \Sigma_{YX} & 0 \\ 0 & \Sigma_{YY} \end{bmatrix}$$ (3)

hence

$$\det \Sigma = \det \Sigma_{YY} \det (\Sigma_{XX} - \Sigma_{XY} \Sigma_{YY}^{-1} \Sigma_{YX})$$ (4)
General Gaussian random vectors

computing the inverse $\Sigma^{-1}$: 

$\Sigma^{-1} = \left[ \begin{array}{cc} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{array} \right]^{-1}$

$\Sigma^{-1} = \left[ \begin{array}{cc} I & 0 \\ -\Sigma_{YY}^{-1}\Sigma_{YX} & I \end{array} \right] \left[ \begin{array}{cc} \Sigma_{XX} - \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX} & 0 \\ 0 & \Sigma_{YY} \end{array} \right]^{-1} \left[ \begin{array}{cc} I & -\Sigma_{XY}\Sigma_{YY}^{-1} \\ 0 & I \end{array} \right]$ 

hence in (2):

$\left[ \begin{array}{c} x - m_X \\ y - m_Y \end{array} \right]^T \Sigma^{-1} \left[ \begin{array}{c} x - m_X \\ y - m_Y \end{array} \right] = \left[ \begin{array}{c} x - (m_X + \Sigma_{XY}\Sigma_{YY}^{-1}[y - m_Y]) \\ y - m_Y \end{array} \right]$ 

hence in (2):

$p(x, y) = p(x | y)p(y) \Rightarrow p(x | y) = \frac{p(x, y)}{p(y)}$

▶ using (4) and (5) in (2), we get

$p(x | y) = \frac{1}{(\sqrt{2\pi})^n \sqrt{\det(\Sigma_{XX} - \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX})}} \times \exp \left\{ -\frac{1}{2} \left[ \begin{array}{c} x - (m_X + \Sigma_{XY}\Sigma_{YY}^{-1}[y - m_Y]) \\ \ldots \end{array} \right]^T \left[ \Sigma_{XX} - \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX} \right]^{-1} \left[ \begin{array}{c} x - (m_X + \Sigma_{XY}\Sigma_{YY}^{-1}[y - m_Y]) \\ \ldots \end{array} \right] \right\}$

hence $X | y$ is also Gaussian, with

$m_{X|y} = m_X + \Sigma_{XY}\Sigma_{YY}^{-1}[y - m_Y]$

$\Sigma_{X|y} = \Sigma_{XX} - \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX}$
Random process

- discrete-time random process: a random variable evolving with time \( \{x(k), k = 1, 2, \ldots \} \)
- a stack of random vectors: \( x(k) = [x(1), x(2), \ldots]^T \)

\[
x(k) \text{ in experiment } \omega_1
\]

\[
x(k) \text{ in } \omega_2
\]

\[
x(k) \text{ in } \omega_3
\]

\[
\vdots
\]

Random process

\( x(k) = [x(1), x(2), \ldots]^T \):

- complete probabilistic properties defined by the joint pdf \( p(x(1), x(2), \ldots) \), which is usually difficult to get
- usually sufficient to know the mean \( \mathbb{E}[x(k)] = m_x(k) \) and auto-covariance:

\[
\mathbb{E}[(x(j) - m_x(j))(x(k) - m_x(k))] = \Sigma_{xx}(j, k) \tag{6}
\]

- sometimes \( \Sigma_{xx}(j, k) \) is also written as \( \chi_{xx}(j, k) \)
Random process

let $x(k)$ be a 1-d random process

- time average of $x(k)$:
  
  $$
  \bar{x}(k) = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{j=-N}^{N} x(j)
  $$

- ensemble average:
  
  $$
  E[x(k)] = m_{x(k)}
  $$

- **ergodic** random process: for all moments of the distribution, the ensemble averages equal the time averages
  
  $$
  E[x(k)] = \bar{x}(k), \quad \Sigma_{xx}(j, k) = [x(j) - m_x][x(k) - m_x], \ldots
  $$

- ergodicity: not easy to test but many processes in practice are ergodic; extremely important as large samples can be expensive to collect in practice

- one necessary condition for ergodicity is stationarity

Random process

stationarity: tells whether the statistics characteristics changes w.r.t. time

- **stationary in the strict sense**: probability distribution does not change w.r.t. time
  
  $$
  Pr \{ x(k_1) \leq x_1, \ldots, x(k_n) \leq x_n \} = Pr \{ x(k_1 + l) \leq x_1, \ldots, x(k_n + l) \leq x_n \}
  $$

- **stationary in the week/wide sense**: mean does not dependent on time
  
  $$
  E[x(k)] = m_x = \text{costant}
  $$

  and the auto-covariance (6) depends only on the time difference $l = j - k$

  - can hence write
    
    $$
    E[(x(k) - m_x)(x(k + l) - m_x)] = \Sigma_{xx}(l) = X_{xx}(l)
    $$

  - for stationary and ergodic random processes:
    
    $$
    \Sigma_{xx}(l) = E[(x(k) - m_x)(x(k + l) - m_x)] = (x(k) - m_x)(x(k + l) - m_x)
    $$
Random process

covariance and correlation for stationary ergodic processes

- we will assume stationarity and ergodicity unless otherwise stated
- **auto-correlation**: \( R_{xx}(l) = E[x(k)x(k+l)] \).

- **cross-covariance**:

\[
\Sigma_{xy}(l) = X_{xy}(l) = E[(x(k) - m_x)(y(k+l) - m_y)]
\]

- property (using ergodicity):

\[
\Sigma_{xy}(l) = X_{xy}(l) = \frac{(x(k) - m_x)(y(k+l) - m_y)}{(y(k+l) - m_y)(x(k) - m_x)} = X_{yx}(-l) = \Sigma_{yx}(-l)
\]

Random process

white noise

- **white noise**: a purely random process with \( x(k) \) not correlated with \( x(j) \) at all if \( k \neq j \):

\[
X_{xx}(0) = \sigma_{xx}^2, \quad X_{xx}(l) = 0 \quad \forall l \neq 0
\]

- non-stationary zero mean white noise:

\[
E[x(k)x(j)] = Q(k)\delta_{kj}, \quad \delta_{kj} = \begin{cases} 1 & , k = j \\ 0 & , k \neq j \end{cases}
\]
Random process

auto-covariance and spectral density

- **spectral density**: the Fourier transform of auto-covariance

\[
\Phi_{xx}(\omega) = \sum_{l=-\infty}^{\infty} X_{xx}(l) e^{-j\omega l}, \quad X_{xx}(l) = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega l} \Phi_{xx}(\omega) d\omega
\]

- **cross spectral density**:

\[
\Phi_{xy}(\omega) = \sum_{l=-\infty}^{\infty} X_{xy}(l) e^{-j\omega l}, \quad X_{xy}(l) = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega l} \Phi_{xy}(\omega) d\omega
\]

properties:

- the variance of \(x\) is the area under the spectral density curve

\[
\text{Var}[x] = E[(x - E[x])^2] = X_{xx}(0) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Phi_{xy}(\omega) d\omega
\]

- \(X_{xx}(0) \geq |X_{xx}(l)|, \ \forall l\)

Filtering a random process

passing a random process \(u(k)\) through an LTI system (convolution) generates another random process:

\[
y(k) = g(k) * u(k) = \sum_{i=-\infty}^{\infty} g(i) u(k - i)
\]

- if \(u\) is zero mean and ergodic, then

\[
X_{uy}(l) = u(k) \sum_{i=-\infty}^{\infty} u(k + l - i) g(i) = \sum_{i=-\infty}^{\infty} u(k) u(k + l - i) g(i) = \sum_{i=-\infty}^{\infty} X_{uu}(l - i) g(i) = g(l) * X_{uu}(l)
\]

similarly

\[
X_{yy}(l) = \sum_{i=-\infty}^{\infty} X_{yu}(l - i) g(i) = g(l) * X_{yu}(l)
\]

- in pictures:

\[
X_{uu}(l) \xrightarrow{G(z)} X_{uy}(l); \quad X_{yu}(l) \xrightarrow{G(z)} X_{yy}(l)
\]
Filtering a random process
input-output spectral density relation
for a general LTI system

\[ u(k) \rightarrow G(z) = \frac{b_n z^n + b_{n-1} z^{n-1} + \ldots + b_0}{z^n + a_{n-1} z^{n-1} + \ldots + a_0} \rightarrow y(k) \]

\[ Y(z) = G(z) U(z) \iff Y(e^{j\omega}) = G(e^{j\omega}) U(e^{j\omega}) \]

▶ auto-covariance relation in the last slide:

\[ X_{uu}(l) \rightarrow G(z) \rightarrow X_{uy}(l) ; \quad X_{yu}(l) \rightarrow G(z) \rightarrow X_{yy}(l) \]

\[ X_{yu}(l) = X_{uy}(-l) = g(-l) * X_{uu}(-l) = g(-l) * X_{uu}(l) \]

hence

\[ \Phi_{yy}(\omega) = G(e^{j\omega}) G(e^{-j\omega}) \Phi_{uu}(\omega) = |G(e^{j\omega})|^2 \Phi_{uu}(\omega) \]

Filtering a random process
MIMO case:

▶ if \( u \) and \( y \) are vectors, \( G(z) \) becomes a transfer function matrix

▶ dimensions play important roles:

\[ X_{uy}(l) = E \left[ (u(k) - m_u)(y(k + l) - m_y)^T \right] = X_{yu}(-l)^T \]

\[ X_{uu}(l) \rightarrow G(z) \rightarrow X_{uy}(l) ; \quad X_{yu}(l) \rightarrow G(z) \rightarrow X_{yy}(l) \]

\[ X_{yy}(l) = g(l) * X_{yu}(l) = g(l) * X_{uy}^T(-l) \]

\[ = g(l) * [g(-l) * X_{uu}(-l)]^T \]

\[ \Phi_{yy}(e^{j\omega}) = G(e^{j\omega}) \cdot \Phi_{uu}(e^{j\omega}) G^T(e^{-j\omega}) \]
Filtering a random process in state space

consider: \( w(k) \)–zero mean, white, \( E[w(k)w(k)^T] = W(k) \) and

\[
x(k+1) = A(k)x(k) + B_w(k)w(k)
\]

(7)

assume random initial state \( x(0) \) (uncorrelated to \( w(k) \)):

\[
E[x(0)] = m_x, \quad E[(x(0) - m_x)(x(0) - m_x)^T] = X_0
\]

▶ mean of state vector \( x(k) \):

\[
m_x(k+1) = A(k)m_x(k), \quad m_x(0) = m_x
\]

(8)

▶ covariance \( X(k) = X_{xx}(k,k) \):

\[
X(k+1) = A(k)X(k)A^T(k) + B_w(k)W(k)B_w^T(k), \quad X(0) = X_0
\]

▶ intuition: covariance is a “second-moment” statistical property

Lecture 3: Review of Probability Theory

Filtering a random process in state space

dynamics of the mean:

\[
m_x(k+1) = A(k)m_x(k), \quad m_x(0) = m_x
\]

dynamics of the covariance:

\[
X(k+1) = A(k)X(k)A^T(k) + B_w(k)W(k)B_w^T(k), \quad X(0) = X_0
\]

▶ (steady state) if \( A(k) = A \) and is stable, \( B_w(k) = B_w \), and \( w(k) \) is stationary \( W(k) = W \), then

\[
m_x(k) \to 0, \quad X(k) \to \text{a steady state} \ X_{ss}
\]

\[
X_{ss} = AX_{ss}A^T + B_w WB_w^T: \text{ discrete-time Lyapunov Eq.} \quad (9)
\]

\[
X_{ss}(l) = E[x(k)x^T(k+l)] = X_{ss}(A^T)^l
\]

\[
X_{ss}(-l) = X_{ss}(l)^T = A^lX_{ss}
\]
Filtering a random process in state space

Example
first-order system

\[ x(k+1) = ax(x) + \sqrt{1 - a^2} w(k), \quad E[w(k)] = 0, \quad E[w(k)w(j)] = W \delta_{kj} \]

with \(|a| < 1\) and \(x(0)\) uncorrelated with \(w(k)\).

steady-state variance equation (9) becomes

\[ X_{ss} = a^2 X_{ss} + (1 - a^2) W \Rightarrow X_{ss} = W \]

and

\[ X(l) = X(-l) = a^l X_{ss} = a^l W \]

Filtering a random process in state space

Example

\[ x(k+1) = ax(x) + \sqrt{1 - a^2} w(k), \quad E[w(k)] = 0, \quad E[w(k)w(j)] = W \delta_{kj} \]

\[ X(l) = X(-l) = a^l X_{ss} = a^l W \]
Filtering a random process

continuous-time case

similar results hold in the continuous-time case:

\[ u(t) \rightarrow G(s) \rightarrow y(t) \]

- spectral density (SISO case):
  \[
  \Phi_{yy}(j\omega) = G(j\omega) G(-j\omega) \Phi_{uu}(j\omega) = |G(j\omega)|^2 \Phi_{uu}(j\omega)
  \]

- mean and covariance dynamics:
  \[
  \frac{dx(t)}{dt} = Ax(t) + B_w w(t), \quad E[w(t)] = 0, \quad Cov[w(t)] = W
  \]
  \[
  \frac{dm_x(t)}{dt} = Am_x(t), \quad m_x(0) = m_{x0}
  \]
  \[
  \frac{dX(t)}{dt} = AX + XA^T + B_w W B_w^T
  \]

- steady state: \( X_{ss}(\tau) = X_{ss} e^{A^T \tau}; \quad X_{ss}(-\tau) = e^{A \tau} X_{ss} \) where
  \[
  AX_{ss} + X_{ss} A^T = -B_w W B_w^T : \text{continuous-time Lyapunov Eq.}
  \]

Appendix: Lyapunov equations

- discrete-time case:
  \[
  A^T PA - P = -Q
  \]
  has the following unique solution iff \( \lambda_i(A) \lambda_j(A) \neq 1 \) for all \( i,j = 1, \ldots, n \):
  \[
  P = \sum_{k=0}^{\infty} \left(A^T\right)^k Q A^k
  \]

- continuous-time case:
  \[
  A^T P + PA = -Q
  \]
  has the following unique solution iff \( \lambda_i(A) + \bar{\lambda}_j(A) \neq 0 \) for all \( i,j = 1, \ldots, n \):
  \[
  P = \int_{0}^{\infty} e^{A^T t} Q e^{A t} dt
  \]
Summary

1. Big picture

2. Basic concepts: sample space, events, probability axioms, random variable, pdf, cdf, probability distributions

3. Multiple random variables
   random vector, joint probability and distribution, conditional probability
   Gaussian case

4. Random process
Big picture

general least squares estimation:

► given: jointly distributed $x$ ($n$-dimensional) & $y$ ($m$-dimensional)

► goal: find the optimal estimate $\hat{x}$ that minimizes

$$E \left[ \|x - \hat{x}\|^2 \mid y = y_1 \right] = E \left[ (x - \hat{x})^T (x - \hat{x}) \mid y = y_1 \right]$$

► solution: consider

$$J(z) = E \left[ \|x - z\|^2 \mid y = y_1 \right] = E \left[ x^T x \mid y = y_1 \right] - 2z^T E[x \mid y = y_1] + z^T z$$

which is quadratic in $z$. For optimal cost,

$$\frac{\partial}{\partial z} J(z) = 0 \Rightarrow z = E[x \mid y = y_1] \triangleq \hat{x}$$

hence

$$\hat{x} = E[x \mid y = y_1] = \int_{-\infty}^{\infty} x p_{x \mid y}(x \mid y_1) \, dx$$

$$J_{\text{min}} = J(\hat{x}) = \text{Tr} \left\{ E \left[ (x - \hat{x})(x - \hat{x})^T \mid y = y_1 \right] \right\}$$
Big picture

general least squares estimation:

\[
\hat{x} = E[x|y = y_1] = \int_{-\infty}^{\infty} xp_{x|y}(x|y_1) \, dx
\]

achieves the minimization of

\[
E[||x - \hat{x}||^2 | y = y_1]
\]

solution concepts:

- the solution holds for any probability distribution in \( y \)
- for each \( y_1 \), \( E[x|y = y_1] \) is different
- if no specific value of \( y \) is given, \( \hat{x} \) is a function of the random vector/variable \( y \), written as

\[
\hat{x} = E[x|y]
\]

Least square estimation in the Gaussian case

Why Gaussian?

- Gaussian is common in practice:
  - macroscopic random phenomena = superposition of microscopic random effects (Central limit theorem)
- Gaussian distribution has nice properties that make it mathematically feasible to solve many practical problems:
  - pdf is solely determined by the mean and the variance/covariance
  - linear functions of a Gaussian random process are still Gaussian
  - the output of an LTI system is a Gaussian random process if the input is Gaussian
  - if two jointly Gaussian distributed random variables are uncorrelated, then they are independent
  - \( X_1 \) and \( X_2 \) jointly Gaussian \( \Rightarrow X_1|X_2 \) and \( X_2|X_1 \) are also Gaussian
Least square estimation in the Gaussian case

Why Gaussian?

Gaussian and white:

▶ they are different concepts
▶ there can be Gaussian white noise, Poisson white noise, etc
▶ Gaussian white noise is used a lot since it is a good approximation to many practical noises

Least square estimation in the Gaussian case

the solution

problem (re-stated): \( x, y \)–Gaussian distributed

\[
\text{minimize } E \left[ \| x - \hat{x} \|^2 \mid y \right]
\]

solution:

\[
\hat{x} = E[x \mid y] = E[x] + X_{xy}X_{yy}^{-1}(y - E[y])
\]

properties:

▶ the estimation is unbiased: \( E[\hat{x}] = E[x] \)
▶ \( y \) is Gaussian \( \Rightarrow \hat{x} \) is Gaussian; and \( x - \hat{x} \) is also Gaussian
▶ covariance of \( \hat{x} \):

\[
E \left[ (\hat{x} - E[\hat{x}])(\hat{x} - E[\hat{x}])^T \right] = E \left\{ (y - E[y]) \left[ X_{xy}X_{yy}^{-1}(y - E[y]) \right]^T \right\} = X_{xy}X_{yy}^{-1}X_{yx}
\]

▶ estimation error \( \tilde{x} \triangleq x - \hat{x} \): zero mean and

\[
\text{Cov}[\tilde{x}] = E \left[ (x - E[x \mid y])(x - E[x \mid y])^T \right] = X_{xx} - X_{xy}X_{yy}^{-1}X_{yx}
\]
**Least square estimation in the Gaussian case**

\[
\hat{x} = E[x|y] = E[x] + X_{xy}X_{yy}^{-1}(y - E[y])
\]

\(E[x|y]\) is a better estimate than \(E[x]\):  
- the estimation is unbiased: \(E[\hat{x}] = E[x]\)
- estimation error \(\tilde{x} \triangleq x - \hat{x}\): zero mean and

\[
\text{Cov}[x - \hat{x}] = X_{xx} - X_{xy}X_{yy}^{-1}X_{yx} \preceq \text{Cov}[x - E[X]]
\]

---

**Properties of least square estimate (Gaussian case)**

two random vectors \(x\) and \(y\)

**Property 1:**
(i) the estimation error \(\tilde{x} = x - \hat{x}\) is uncorrelated with \(y\)
(ii) \(\tilde{x}\) and \(\hat{x}\) are orthogonal:

\[
E[(x - \hat{x})^T\hat{x}] = 0
\]

**proof** of (i):

\[
E[\tilde{x}(y - m_y)^T] = E[(x - E[x] - X_{xy}X_{yy}^{-1}(y - m_y))(y - m_y)^T] \\
= X_{xy} - X_{xy}X_{yy}^{-1}X_{yx} = 0
\]

---
Properties of least square estimate (Gaussian case)

two random vectors $x$ and $y$

**proof** of (ii): $E[\tilde{x}^T \hat{x}] = E[(x - \hat{x})^T (E[x] + X_{xy}X_{yy}^{-1}(y - m_y))] = E[\tilde{x}^T] E[x] + E[(x - \hat{x})^T X_{xy}X_{yy}^{-1}(y - m_y)]$ where $E[\tilde{x}^T] = 0$ and

$$E[(x - \hat{x})^T X_{xy}X_{yy}^{-1}(y - m_y)] = \text{Tr}\left\{ E[X_{xy}X_{yy}^{-1}(y - m_y)(x - \hat{x})^T]\right\}$$

$$= \text{Tr}\left\{ X_{xy}X_{yy}^{-1}E[(y - m_y)(x - \hat{x})^T]\right\} = 0 \text{ because of (i)}$$

▶ note: $\text{Tr}\left\{BA\right\} = \text{Tr}\left\{AB\right\}$. Consider, e.g. $A = [a, b], B = \begin{bmatrix} c \\ d \end{bmatrix}$

---

Properties of least square estimate (Gaussian case)

two random vectors $x$ and $y$

**Property 1** (re-stated):

(i) the estimation error $\tilde{x} = x - \hat{x}$ is uncorrelated with $y$

(ii) $\tilde{x}$ and $\hat{x}$ are orthogonal:

$$E[(x - \hat{x})^T \hat{x}] = 0$$

▶ intuition: least square estimation is a projection
Properties of least square estimate (Gaussian case)
three random vectors \(x, y, z\), where \(y\) and \(z\) are uncorrelated

**Property 2:** let \(y\) and \(z\) be Gaussian and uncorrelated, then

(i) the optimal estimate of \(x\) is

\[
E[x|y, z] = E[x] + \left( E[x|y] - E[x] \right) + \left( E[x|z] - E[x] \right)
\]

= \(E[x|y] + (E[x|z] - E[x])\)

Alternatively, let \(\hat{x}_{|y} \triangleq E[x|y]\), \(\tilde{x}_{|y} = x - E[x|y] = x - \hat{x}_{|y}\), then

\[
E[x|y, z] = E[x|y] + E[\tilde{x}_{|y}|z]
\]

(ii) the estimation error covariance is

\[
X_{xx} - X_{xy}X_{yy}^{-1}X_{yx} - X_{xz}X_{zz}^{-1}X_{zx} = X_{\tilde{x}\tilde{x}} - X_{\tilde{x}z}X_{zz}^{-1}X_{z\tilde{x}}
\]

where \(X_{\tilde{x}\tilde{x}} = E[\tilde{x}_{|y}\tilde{x}_{|y}^T]\) and \(X_{\tilde{x}z} = E[\tilde{x}_{|y}(z - m_z)^T]\)

Proof of (i): let \(w = [y, z]^T\)

\[
E[x|w] = E[x] + \left[ X_{xy} \quad X_{xz} \right] \left[ X_{yy} \quad X_{yz} \right]^{-1} \begin{bmatrix} y - E[y] \\ z - E[z] \end{bmatrix}
\]

Using \(X_{yz} = 0\) yields

\[
E[x|w] = E[x] + \frac{X_{xy}X_{yy}^{-1}(y - E[y])}{E[x|y] - E[x]} + \frac{X_{xz}X_{zz}^{-1}(z - E[z])}{E[x|z] - E[x]}
\]

= \(E[x|y] + E[(\hat{x}_{|y} + \tilde{x}_{|y})|z] - E[x]\)

= \(E[x|y] + E[\tilde{x}_{|y}|z]\)

where \(E[\tilde{x}_{|y}|z] = E[E[x|y]|z] = E[x]\) as \(y\) and \(z\) are independent
Properties of least square estimate (Gaussian case)

three random vectors $x$, $y$, and $z$, where $y$ and $z$ are uncorrelated

**proof** of (ii): let $w = [y, z]^T$, the estimation error covariance is

$$X_{xx} - X_{xw} X_{ww}^{-1} X_{wx} = X_{xx} - X_{xy} X_{yy}^{-1} X_{yx} - X_{xz} X_{zz}^{-1} X_{zx}$$

additionally

$$X_{xz} = E \left[ (x - E[x]) (z - E[z])^T \right] = E \left[ (\hat{x}_{|y} + \tilde{x}_{|y} - E[x]) (z - E[z])^T \right]$$

$$= E \left[ (\hat{x}_{|y} - E[x]) (z - E[z])^T \right] + E \left[ \tilde{x}_{|y} (z - E[z])^T \right]$$

but $\hat{x}_{|y} - E[x]$ is a linear function of $y$, which is uncorrelated with $z$, hence $E \left[ (\hat{x}_{|y} - E[x]) (z - E[z])^T \right] = 0$ and $X_{xz} = X_{\tilde{x}_{|y} z}$

---

Properties of least square estimate (Gaussian case)

three random vectors $x$, $y$, and $z$, where $y$ and $z$ are uncorrelated

**Property 2** (re-stated): let $y$ and $z$ be Gaussian and uncorrelated

(i) the optimal estimate of $x$ is

$$E[x|y, z] = E[x|y] + E[\tilde{x}_{|y}|z]$$

(ii) the estimation error covariance is

$$X_{\tilde{x}\tilde{x}} - X_{\tilde{x}z} X_{zz}^{-1} X_{z\tilde{x}}$$

▶ intuition:

![Graphical representation of intuition](image-url)
Properties of least square estimate (Gaussian case)

three random vectors $x$, $y$, and $z$, where $y$ and $z$ are correlated

**Property 3:** let $y$ and $z$ be Gaussian and correlated, then

(i) the optimal estimate of $x$ is

$$E[x|y,z] = E[x|y] + E[\tilde{x}_y|\tilde{z}_y]$$

where $\tilde{z}_y = z - \hat{z}_y = z - E[z|y]$ and $\tilde{x}_y = x - \hat{x}_y = x - E[x|y]$ (ii) the estimation error covariance is

$$X_{\tilde{x}_y|\tilde{z}_y} - X_{\tilde{x}_y\tilde{z}_y} X_{\tilde{z}_y\tilde{z}_y}^{-1} X_{\tilde{z}_y|\tilde{z}_y}$$

▶ intuition:

$E[\tilde{x}_y|\tilde{z}_y]$

Application of the three properties

Consider

![Diagram](image)

Given $[y(0), y(1), \ldots, y(k)]^T$, we want to estimate the state $x(k)$

▶ the properties give a recursive way to compute

$$\hat{x}(k)|\{y(0), y(1), \ldots, y(k)\}$$
Example

Consider estimating the velocity $x$ of a motor, with

\[
E[x] = m_x = 10 \text{ rad/s}
\]

\[
\text{Var}[x] = 2 \text{ rad}^2/\text{s}^2
\]

There are two (tachometer) sensors available:

$\triangleright \ y_1 = x + v_1: \ E[v_1] = 0, \ E[v_1^2] = 1 \text{ rad}^2/\text{s}^2$

$\triangleright \ y_2 = x + v_2: \ E[v_2] = 0, \ E[v_2^2] = 1 \text{ rad}^2/\text{s}^2$

where $v_1$ and $v_2$ are independent, Gaussian, $E[v_1 v_2] = 0$ and $x$ is independent of $v_i$, $E[(x - E[x]) v_i] = 0$

Example

$\triangleright$ best estimate of $x$ using only $y_1$:

\[
X_{xy_1} = E[(x - m_x)(y_1 - m_{y_1})] = E[(x - m_x)(x - m_x + v_1)]
\]

\[
= X_{xx} + E[(x - m_x) v_1] = 2
\]

\[
X_{y_1y_1} = E[(y_1 - m_{y_1})(y_1 - m_{y_1})] = E[(x - m_x + v_1)(x - m_x + v_1)]
\]

\[
= X_{xx} + E[v_1^2] = 3
\]

\[
\hat{x}_{|y_1} = E[x] + X_{xy_1} X_{y_1y_1}^{-1} (y_1 - E[y_1]) = 10 + \frac{2}{3} (y_1 - 10)
\]

$\triangleright$ similarly, best estimate of $x$ using only $y_2$: $\hat{x}_{|y_2} = 10 + \frac{2}{3} (y_2 - 10)$
Example

- best estimate of $x$ using $y_1$ and $y_2$ (direct approach): let $y = [y_1, y_2]^T$

\[
X_{xy} = E \left[ (x - m_x) \left[ \begin{array}{c} y_1 - m_{y_1} \\ y_2 - m_{y_2} \end{array} \right]^T \right] = [2, 2]
\]

\[
X_{yy} = E \left[ \left[ \begin{array}{c} y_1 - m_{y_1} \\ y_2 - m_{y_2} \end{array} \right] \left[ \begin{array}{cc} y_1 - m_{y_1} & y_2 - m_{y_2} \end{array} \right] \right] = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}
\]

\[
\hat{x}_{|y} = E[x] + X_{xy}X_{yy}^{-1}(y - m_y) = 10 + [2, 2] \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}^{-1} \begin{bmatrix} y_1 - 10 \\ y_2 - 10 \end{bmatrix}
\]

- note: $X_{yy}^{-1}$ is expensive to compute at high dimensions

Example

- best estimate of $x$ using $y_1$ and $y_2$ (alternative approach using Property 3):

\[
E[x|y_1, y_2] = E[x|y_1] + E[\tilde{x}_1|y_1]E[\tilde{y}_2|y_1]
\]

which involves just the scalar computations:

\[
E[x|y_1] = 10 + \frac{2}{3} (y_1 - 10), \quad \tilde{x}_{1|y_1} = x - E[x|y_1] = \frac{1}{3} (x - 10) + \frac{2}{3} v_1
\]

\[
\tilde{y}_{2|y_1} = y_2 - E[y_2|y_1] = y_2 - \left[ E[y_2] + X_{y_2|y_1} \frac{1}{X_{y_1|y_1}} (y_1 - m_{y_1}) \right] = (y_2 - 10) - \frac{2}{3} (y_1 - 10)
\]

\[
X_{\tilde{x}_1|y_1, \tilde{y}_2|y_1} = E \left[ \left( \frac{1}{3} (x - 10) + \frac{2}{3} v_1 \right) \left( y_2 - 10 - \frac{2}{3} (y_1 - 10) \right)^T \right] = \frac{1}{9} \text{Var}[x] + \frac{4}{9} \text{Var}[v_1] = \frac{2}{3}
\]

\[
X_{\tilde{y}_2|y_1, \tilde{y}_2|y_1} = \frac{1}{9} \text{Var}[x] + \text{Var}[v_2] + \frac{4}{9} \text{Var}[v_1] = \frac{5}{3}
\]

\[
E[\tilde{x}_1|y_2|y_1] = E[\tilde{x}_1|y_1] + X_{\tilde{x}_1|y_1, \tilde{y}_2|y_1} \frac{1}{X_{\tilde{y}_2|y_1, \tilde{y}_2|y_1}} [\tilde{y}_2|y_1 - E[\tilde{y}_2|y_1]]
\]

\[
= 10 + \frac{2}{5} (y_1 - 10) + \frac{2}{5} (y_2 - 10)
\]
Summary

1. Big picture
\[ \hat{x} = E[x|y] \text{ minimizes } J = E[||x - \hat{x}||^2 | y] \]

2. Solution in the Gaussian case
Why Gaussian?
\[ \hat{x} = E[x|y] = E[x] + X_{xy}X_{yy}^{-1}(y - E[y]) \]

3. Properties of least square estimate (Gaussian case)
two random vectors \( x \) and \( y \)
three random vectors \( x, y \) and \( z \): \( y \) and \( z \) are uncorrelated
three random vectors \( x, y \) and \( z \): \( y \) and \( z \) are correlated

* Appendix: trace of a matrix

- the trace of a \( n \times n \) matrix is given by \( \text{Tr}(A) = \sum_{i=1}^{n} a_{ii} \)
- trace is the matrix inner product:
  \[ \langle A, B \rangle = \text{Tr}(A^T B) = \text{Tr}(B^T A) = \langle B, A \rangle \] (1)
- take a three-column example: write the matrices in the column vector form \( B = [b_1, b_2, b_3] \), \( A = [a_1, a_2, a_3] \), then,
  \[ A^T B = \begin{bmatrix} a_1^T b_1 & * & * \\ * & a_2^T b_2 & * \\ * & * & a_3^T b_3 \end{bmatrix} \] (2)
  \[ \text{Tr}(A^T B) = a_1^T b_1 + a_2^T b_2 + a_3^T b_3 = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}^T \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \] (3)
  which is the inner product of the two long stacked vectors.
- we frequently use the inner-product equality \( \langle A, B \rangle = \langle B, A \rangle \)
Big picture
why are we learning this?
▶ state estimation in deterministic case:

Plant: \( x(k+1) = Ax(k) + Bu(k), \ y(k) = Cx(k) \)
Observer: \( \hat{x}(k+1) = A\hat{x}(k) + Bu(k) + L(y(k) - C\hat{x}(k)) \)

▶ \( L \) designed based on the error \( (e(k) = x(k) - \hat{x}(k)) \) dynamics:
\[
e(k+1) = (A - LC)e(k)
\] (1)
to reach fast convergence of \( \lim_{k \to \infty} e(k) = 0 \)
▶ \( L \) is not optimal when there is noise in the plant; actually \( \lim_{k \to \infty} e(k) = 0 \) isn’t even a valid goal when there is noise
▶ Kalman Filter provides optimal state estimation under input and output noises
Problem statement

plant: \( x(k+1) = A(k)x(k) + B(k)u(k) + B_w(k)w(k) \)
\( y(k) = C(k)x(k) + v(k) \)

- \( w(k) \)–s-dimensional input noise; \( v(k) \)–r-dimensional measurement noise; \( x(0) \)–unknown initial state
- assumptions: \( x(0), w(k), \) and \( v(k) \) are independent and Gaussian distributed; \( w(k) \) and \( v(k) \) are white:
  \[
  E[x(0)] = x_o, \quad E[(x(0) - x_o)(x(0) - x_o)^T] = X_0 \\
  E[w(k)] = 0, \quad E[v(k)] = 0, \quad E[w(k)v^T(j)] = 0 \quad \forall k, j \\
  E[w(k)w^T(j)] = W(k)\delta_{kj}, \quad E[v(k)v^T(j)] = V(k)\delta_{kj}
  \]

Problem statement

- goal:
  \[
  \text{minimize } E[\|x(k) - \hat{x}(k)\|^2|Y_j], \quad Y_j = \{y(0), y(1), \ldots, y(j)\}
  \]
- solution:
  \[ \hat{x}(k) = E[x(k)|Y_j] \]
- three classes of problems:
  - \( k > j \): prediction problem
  - \( k = j \): filtering problem
  - \( k < j \): smoothing problem
History

Rudolf Kalman:

- obtained B.S. in 1953 and M.S. in 1954 from MIT, and Ph.D. in 1957 from Columbia University, all in Electrical Engineering
- developed and implemented Kalman Filter in 1960, during the Apollo program, and furthermore in various famous programs including the NASA Space Shuttle, Navy submarines, etc.
- was awarded the National Medal of Science on Oct. 7, 2009 from U.S. president Barack Obama

Useful facts

assume $x$ is Gaussian distributed

- if $y = Ax + B$ then
  \[
  X_{xy} = E \left[ (x - E[x]) (y - E[y])^T \right] = X_{xx} A^T
  \]
  \[
  X_{yy} = E \left[ (y - E[y]) (y - E[y])^T \right] = AX_{xx} A^T
  \]  
  (2)

- if $y = Ax + B$ and $y' = A' x + B'$ then
  \[
  X_{yy'} = AX_{xx} \left( A' \right)^T, \quad X_{y'y} = A' X_{xx} A^T
  \]  
  (3)

- if $y = Ax + Bv; \ v$ is Gaussian and independent of $x$, then
  \[
  X_{yy} = AX_{xx} A^T + BX_{vv} B^T
  \]  
  (4)

- if $y = Ax + Bv, \ y' = A' x + B' v; \ v$ is Gaussian and dependent of $x$, then
  \[
  X_{yy'} = AX_{xx} \left( A' \right)^T + AX_{xx} \left( B' \right)^T + BX_{vx} \left( A' \right)^T + BX_{vv} \left( B' \right)^T
  \]  
  (5)
Derivation of Kalman Filter

- goal:
  \[ \text{minimize } E \left[ \| x(k) - \hat{x}(k) \|^2 \right| \gamma_k], \quad \gamma_k = \{y(0), y(1), \ldots, y(k)\} \]

- the best estimate is the conditional expectation
  \[ E[x(k) | \gamma_k] = E[x(k) | \{\gamma_{k-1}, y(k)\}] = E[x(k) | \gamma_{k-1}] + E[\check{x}(k) | \gamma_{k-1}] \tilde{y}(k) | \gamma_{k-1}] \]

- introduce some notations:
  - a priori estimation \( \hat{x}(k|k-1) = E[x(k) | \gamma_{k-1}] = \hat{x}(k) | y(0), \ldots, y(k-1) \)
  - a posteriori estimation \( \hat{x}(k|k) = E[x(k) | \gamma_k] = \hat{x}(k) | y(0), \ldots, y(k) \)
  - a priori covariance \( M(k) = E[\check{x}(k) | \gamma_{k-1} \check{x}^T(k) | \gamma_{k-1}] \)
  - a posteriori covariance \( Z(k) = E[\check{x}(k) | \gamma_k \check{x}^T(k) | \gamma_k] \)

Derivation of Kalman Filter

KF gain update

to get \( E[\check{x}(k) | \gamma_{k-1} | \tilde{y}(k) | \gamma_{k-1}] \) in

\[ E[x(k) | \gamma_k] = E[x(k) | \gamma_{k-1}] + E[\check{x}(k) | \gamma_{k-1}] \tilde{y}(k) | \gamma_{k-1}] \]

we need \( X_{\check{x}(k)|\gamma_{k-1}} \tilde{y}(k)|\gamma_{k-1} \) and \( X_{\tilde{y}(k)|\gamma_{k-1}} \tilde{y}(k)|\gamma_{k-1} \)

\( y(k) = C(k) x(k) + v(k) \) gives

\[ \hat{y}(k) | \gamma_{k-1} = C(k) \hat{x}(k|k-1) + \hat{v}(k) | \gamma_{k-1} = C(k) \hat{x}(k|k-1) \]

\[ \Rightarrow \tilde{y}(k) | \gamma_{k-1} = C(k) \check{x}(k|k-1) + v(k) \]

hence

\[ X_{\check{x}(k)|\gamma_{k-1}} \tilde{y}(k)|\gamma_{k-1} = M(k) C^T(k) \] (6)

\[ X_{\tilde{y}(k)|\gamma_{k-1}} \tilde{y}(k)|\gamma_{k-1} = C(k) M(k) C^T(k) + V(k) \] (7)
Derivation of Kalman Filter

KF gain update

\[ \tilde{y}(k) \mid Y_{k-1} = C(k) \tilde{x}(k|k-1) + \nu(k) \]

unbiased estimation: \( E[\hat{x}(k|k-1)] = E[x] \Rightarrow \)

\[ E[\tilde{y}(k) \mid Y_{k-1}] = E[\tilde{x}(k) \mid Y_{k-1}] + E[\nu(k) \mid Y_{k-1}] = 0 \]

thus

\[ E[\tilde{x}(k) \mid Y_{k-1} \middle| \tilde{y}(k) \mid Y_{k-1}] \]

\[ = E[\tilde{x}(k) \mid Y_{k-1}] + X_{\tilde{x}(k) \mid Y_{k-1}} X_{\tilde{y}(k) \mid Y_{k-1}}^{-1} \tilde{y}(k) \mid Y_{k-1} \tilde{y}(k) \mid Y_{k-1} - 0 \]

\[ = M(k) C^T(k) \left[ C(k) M(k) C^T(k) + V(k) \right]^{-1} (y(k) - \hat{y}(k) \mid Y_{k-1}) \]

now becomes

\[ \hat{x}(k|k) = \hat{x}(k|k-1) + M(k) C^T(k) \left[ C(k) M(k) C^T(k) + V(k) \right]^{-1} (y(k) - C \hat{x}(k|k-1)) \]

\[ F(k) = M(k) C^T(k) \left[ C(k) M(k) C^T(k) + V(k) \right]^{-1} \]

namely

\[ \begin{aligned} \hat{x}(k|k) &= \hat{x}(k|k-1) + F(k) (y(k) - C(k) \hat{x}(k|k-1)) \\ F(k) &= M(k) C^T(k) \left[ C(k) M(k) C^T(k) + V(k) \right]^{-1} \end{aligned} \] (8)
Derivation of Kalman Filter

KF covariance update

now for the variance update:

\[
E \left[ \tilde{x}(k) | Y_k \right] = E \left[ \tilde{x}(k) | \{Y_{k-1}, y(k)\} \right] \\
= E \left[ \tilde{x}(k) | Y_{k-1} \right] - X_{\tilde{x}(k)|Y_{k-1}} X_{\tilde{y}(k)|Y_{k-1}}^{-1} X_{\tilde{y}(k)|Y_{k-1}} \tilde{x}(k)|Y_{k-1}
\]

or, using the introduced notations,

\[
Z(k) = M(k) - M(k) C^T(k) \left( C(k) M(k) C^T(k) + V(k) \right)^{-1} C(k) M(k)
\]

the connection between \( Z(k) \) and \( M(k) \):

\[
x(k) = A(k-1) x(k-1) + B(k-1) u(k-1) + B_w(k-1) w(k-1)
\Rightarrow \tilde{x}(k|k-1) = A(k-1) \tilde{x}(k-1|k-1) + B(k-1) u(k-1)
\Rightarrow \tilde{x}(k|k-1) = A(k-1) \tilde{x}(k-1|k-1) + B_w(k-1) w(k-1)
\]

thus \( M(k) = \text{Cov} [\tilde{x}(k|k-1)] \) is [using useful fact (4)]

\[
M(k) = A(k-1) Z(k-1) A^T(k-1) + B_w(k-1) W(k-1) B_w^T(k-1)
\]

with \( M(0) = E \left[ \tilde{x}(0|1) \tilde{x}(0|1)^T \right] = X_0 \)
The full set of KF equations

in a shifted index:

\[ \hat{x}(k+1|k) = \hat{x}(k+1|k) + F(k+1) [y(k+1) - C(k+1) \hat{x}(k+1|k)] \]
\[ \hat{x}(k+1|k) = A(k) \hat{x}(k|k) + B(k) u(k) \]
\[ F(k+1) = M(k+1) C^T(k+1) \left[ C(k+1) M(k+1) C^T(k+1) + V(k+1) \right]^{-1} \]
\[ M(k+1) = A(k) Z(k) A^T(k) + B_w(k) W(k) B_w^T(k) \]
\[ Z(k+1) = M(k+1) - M(k+1) C^T(k+1) \ldots \]
\[ \times \left( C(k+1) M(k+1) C^T(k+1) + V(k+1) \right)^{-1} C(k+1) M(k+1) \]

combining (9) and (10) gives the Riccati equation:

\[ M(k+1) = A(k) M(k) A^T(k) + B_w(k) W(k) B_w^T(k) \]
\[ - A(k) M(k) C^T(k) \left[ C(k) M(k) C^T(k) + V(k) \right]^{-1} C(k) M(k) A^T(k) \]

with initial conditions \( \hat{x}(0|1) = x_0 \) and \( M(0) = X_0 \).
The full set of KF equations

Several remarks

- $F(k)$, $M(k)$, and $Z(k)$ can be obtained offline first
- Kalman Filter (KF) is linear, and optimal for Gaussian. More advanced nonlinear estimation won’t improve the results here.
- KF works for time-varying systems
- the block diagram of KF is:

Steady-state KF

assumptions:

- system is time-invariant: $A$, $B$, $B_w$, and $C$ are constant;
- noise is stationary: $V > 0$ and $W > 0$ do not depend on time.

KF equations become:

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + F(k+1)[y(k+1) - C\hat{x}(k+1|k)]$$
$$= A\hat{x}(k|k) + Bu(k) + F(k+1)[y(k+1) - C\hat{x}(k+1|k)]$$

$$F(k+1) = M(k+1)C^T\left[CM(k+1)C^T + V\right]^{-1}$$

$$M(k+1) = AZ(k)A^T + B_wW^T_B; \quad M(0) = X_0$$

$$Z(k+1) = M(k+1) - M(k+1)C^T\left[CM(k+1)C^T + V\right]^{-1}CM(k+1)$$

with Riccati equation (RE):

$$M(k+1) = AM(k)A^T + B_wW^T_B - AM(k)C^T\left[CM(k)C^T + V\right]^{-1}CM(k)A^T$$
Steady-state KF

if

- $(A, C)$ is observable or detectable
- $(A, B_w)$ is controllable (disturbable) or stabilizable

then $M(k)$ in the RE converges to some steady-state value $M_s$ and KF can be implemented by

$$
\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + F_s [y(k+1) - C\hat{x}(k+1|k)]
$$

$$
\hat{x}(k+1|k) = A\hat{x}(k|k) + Bu(k)
$$

$$
F_s = M_s C^T [CM_s C^T + V]^{-1}
$$

$M_s$ is the positive definite solution of the algebraic Riccati equation:

$$
M_s = AM_s A^T + B_w W B_w^T - AM_s C^T \left[ CM_s C^T + V \right]^{-1} CM_s A^T
$$

Duality with LQ

The steady-state condition is obtained by comparing the RE in LQ and KF

discrete-time LQ:

$$
P(k) = A^T P(k+1) A - A^T P(k+1) B [R + B^T P(k+1) B]^{-1} B^T P(k+1) A + Q
$$

discrete-time KF (11):

$$
M(k+1) = AM(k) A^T - AM(k) C^T \left[ CM(k) C^T + V \right]^{-1} CM(k) A^T + B_w W B_w^T
$$

<table>
<thead>
<tr>
<th>Discrete-time LQ</th>
<th>Discrete-time KF</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>$A^T$</td>
</tr>
<tr>
<td>$B$</td>
<td>$C^T$</td>
</tr>
<tr>
<td>$C$</td>
<td>$B_w$</td>
</tr>
<tr>
<td>$R$</td>
<td>$V$</td>
</tr>
<tr>
<td>$Q = C^T C$</td>
<td>$B_w W B_w^T$</td>
</tr>
<tr>
<td>$P$</td>
<td>$M$</td>
</tr>
<tr>
<td>Backward recursion</td>
<td>Forward recursion</td>
</tr>
</tbody>
</table>
Duality with LQ

<table>
<thead>
<tr>
<th>discrete-time LQ</th>
<th>discrete-time KF</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>$A^T$</td>
</tr>
<tr>
<td>B</td>
<td>$C^T$</td>
</tr>
<tr>
<td>C</td>
<td>$B_w$</td>
</tr>
<tr>
<td>$Q = C^T C$</td>
<td>$B_w W B_w^T$</td>
</tr>
</tbody>
</table>

steady-state conditions for discrete-time LQ:
- $(A, B)$ controllable or stabilizable
- $(A, C)$ observable or detectable

steady-state conditions for discrete-time KF:
- $(A^T, C^T)$ controllable or stabilizable $\iff (A, C)$ observable or detectable
- $(A^T, B^T_w)$ observable or detectable $\iff (A, B_w)$ controllable or stabilizable

Lecture 5: Stochastic State Estimation (Kalman Filter)

Duality with LQ

- LQ: stable closed-loop “$A$” matrix is
  $$A - BK_s = A - B[R + B^T P_s B]^{-1} B^T P_s A$$

- KF: stable KF “$A$” matrix is
  $$\hat{x}(k + 1|k) = A\hat{x}(k|k) + Bu(k)$$
  $$= A\hat{x}(k|k - 1) + AF_s [y(k) - C\hat{x}(x|k - 1)] + Bu(k)$$
  $$= \left[A - AM_s C^T \left(CM_s C^T + V\right)^{-1} C\right] \hat{x}(k|k - 1) + \ldots$$
Purpose of each condition

▶ (A, C) observable or detectable: assures the existence of the steady-state Riccati solution

▶ (A, Bw) controllable or stabilizable: assures that the steady-state solution is positive definite and that the KF dynamics is stable

Remark

▶ KF: stable KF “A” matrix is

\[
\hat{x}(k + 1|k) = \left[ A - AM_s C^T \left( CM_s C^T + V \right)^{-1} C \right] \hat{x}(k|k - 1) + \ldots
\]

\[= (A - AF_s C) \hat{x}(k|k - 1) + \ldots \]

in the form of \( \hat{x}(k|k) \) dynamics:

\[
\hat{x}(k + 1|k + 1) = \hat{x}(k + 1|k) + F_s [y(k + 1) - C \hat{x}(k + 1|k)]
\]

\[= (A - F_s CA) \hat{x}(k|k) + (I - F_s C) Bu(k) + F_s y(k + 1) \]

\[= \left[ A - M_s C^T \left( CM_s C^T + V \right)^{-1} CA \right] \hat{x}(k|k) + \ldots \]

▶ can show that \( \text{eig} (A - AF_s C) = \text{eig} (A - F_s CA) \)

hint: \( \text{det} (I + MN) = \text{det} (I + NM) \Rightarrow \text{det} \left[ I - z^{-1} A(I - F_s C) \right] = \text{det} \left[ I - (I - F_s C) z^{-1} A \right] \)
Remark

intuition of guaranteed KF stability: ARE ⇒ Lyapunov equation

\[
M_s = AM_s A^T + B_w W B_w^T - AM_s C^T 
\left[ CM_s C^T + V \right]^{-1} CM_s A^T \\
= AM_s A^T + B_w W B_w^T - AM_s C^T \left[ CM_s C^T + V \right]^{-1} \left[ CM_s C^T + V \right]^{-1} CM_s A^T \\
= (A - AF_s C) M_s (A - AF_s C)^T + 2AF_s CM_s C^T F_s^T A^T \\
+ B_w W B_w^T - AF_s \left[ CM_s C^T + V \right] F_s^T A^T \\
= (A - AF_s C) M_s (A - AF_s C)^T + AF_s V F_s^T A^T + B_w W B_w^T \\
\]

\[
\iff (A - AF_s C) M_s (A - AF_s C)^T - M_s = -AF_s V F_s^T A^T - B_w W B_w^T \\
\]

which is a Lyapunov equation with the right hand side being negative semidefinite and \( M_s \succ 0 \).

Return difference equation

KF dynamics

\[
\hat{x}(k+1|k+1) = (A - F_s C A) \hat{x}(k|k) + (I - F_s C) B u(k) + F_s y(k+1) \\
= A \hat{x}(k|k) - F_s C A \hat{x}(k|k) + (I - F_s C) B u(k) + F_s y(k+1) \\
[ z I - A ] \hat{x}(k|k) = F_s y(k+1) + (I - F_s C) B u(k) - F_s C A \hat{x}(k|k) \\
\]

\[
\overset{+}{\sim} \rightarrow F_s \rightarrow (z I - A)^{-1} \hat{x}(k|k) \rightarrow \overset{-}{\sim} \rightarrow \rightarrow C A \\
\]

let \( G(z) = C(z I - A)^{-1} B_w \) 
ARE ⇒ return difference equation (RDE) (see ME232 reader)

\[
[ I + CA(z - 1 I - A)^{-1} F_s ] ( V + C M_s C^T ) [ I + CA(z - 1 I - A)^{-1} F_s ]^T = V + G(z) W G^T (z^{-1}) \\
\]
Symmetric root locus for KF

- KF eigenvalues:
  \[
  \det \left[ I + CA(zI - A)^{-1}F_s \right] = \det \left[ I + (zI - A)^{-1}F_s CA \right] \\
  = \frac{\det(zI - A + F_s CA)}{\det(zI - A)} \triangleq \beta(z) \phi(z)
  \]

- taking determinants in RDE gives
  \[
  \beta(z)\beta(z^{-1}) = \phi(z)\phi(z^{-1}) \frac{\det(V + G(z)WG^T(z^{-1}))}{\det(V + CM^C^T)}
  \]

- single-output case: KF poles come from \( \beta(z)\beta(z^{-1}) = 0 \), i.e.
  \[
  \det \left( V + G(z)WG^T(z^{-1}) \right) = V \left( 1 + G(z)\frac{W}{V}G^T(z^{-1}) \right) = 0
  \]

- \( W/V \to 0 \): KF poles \( \to \) stable poles of \( G(z) G^T(z^{-1}) \)
- \( W/V \to \infty \): KF poles \( \to \) stable zeros of \( G(z) G^T(z^{-1}) \)

Continuous-time KF

summary of solutions

system:
\[
\dot{x}(t) = Ax(t) + Bu(t) + B_w w(t) \\
y(t) = Cx(t) + v(t)
\]

assumptions: same as discrete-time KF

aim: minimize \( J = \|x(t) - \hat{x}(t)\|_2^2 \{y(\tau) : 0 \leq \tau \leq t\} \)

continuous-time KF:
\[
\frac{d\hat{x}(t|t)}{dt} = A\hat{x}(t|t) + Bu(t) + F(t) [y(t) - C\hat{x}(t|t)] , \ \hat{x}(0|0) = x_0
\]

\[
F(t) = M(t) C^T V^{-1}
\]

\[
\frac{dM(t)}{dt} = AM(t) + M(t) A^T + B_w W B_w^T - M(t) C^T V^{-1} CM(t) , \ \ M(0) = X_0
\]
Continuous-time KF: steady state

assumptions: \((A, C)\) observable or detectable; 
\((A, B_w)\) controllable or stabilizable

asymptotically stable steady-state KF:

\[
\frac{d\hat{x}(t|t)}{dt} = A\hat{x}(t|t) + B u(t) + F_s [y(t) - C\hat{x}(t|t)]
\]

\[
F_s = M_s C^T V^{-1}
\]

\[
A M_s + M_s A^T + B_w W B_w^T - M_s C^T V^{-1} C M_s = 0
\]

duality with LQ:

\[
A^T P_s + P_s A + Q - P_s B R^{-1} B^T P_s = 0
\]

\[
K = R^{-1} B^T P_s
\]

Continuous-time KF: return difference equality

analogy to LQ gives the return difference equality:

\[
\begin{bmatrix}
I + C (sI - A)^{-1} F_s
\end{bmatrix} V \begin{bmatrix}
I + F_s^T (-sI - A)^{-T} C^T
\end{bmatrix} = V + G(s) W G^T (-s)
\]

where \(G(s) = C (sI - A)^{-1} B_w\), hence:

\[
\begin{bmatrix}
I + C (j\omega I - A)^{-1} F_s
\end{bmatrix} V \begin{bmatrix}
I + C (-j\omega I - A)^{-1} F_s
\end{bmatrix}^T = V + G(j\omega) W G^T (-j\omega)
\]

observation 1: \(\frac{d\hat{x}(t|t)}{dt} = A\hat{x}(t|t) + B u(t) + F_s [y(t) - C\hat{x}(t|t)] + e_y(t)\)

\[
\begin{array}{c}
+ \\
y \\
\downarrow \downarrow \\
\rightarrow \\
F_s \\
\rightarrow \\
(sI - A)^{-1} \\
\hat{x} \\
\rightarrow \\
C \\
\rightarrow \\
e_y(t)
\end{array}
\]
Continuous-time KF: properties

observation 1:

\[
\begin{array}{c}
\vdots \\
y \\
\uparrow \\
\rightarrow \mathbf{e}_y \\
\rightarrow F_s \\
\rightarrow (sI - A)^{-1} \hat{x} \\
\rightarrow C \\
\rightarrow \\
\rightarrow \vdots \\
\end{array}
\]

- transfer function from \( y \) to \( \mathbf{e}_y \):
  \[
  \left[ I + C (j\omega - A)^{-1} F_s \right]^{-1}
  \]

- spectral density relation:
  \[
  \Phi_{\mathbf{e}_y,\mathbf{e}_y}(\omega) = \left[ I + C (j\omega - A)^{-1} F_s \right]^{-1} \Phi_{\mathbf{yy}}(\omega) \left[ I + C (-j\omega - A)^{-1} F_s \right]^{-1} \left( I + C (j\omega - A)^{-1} F_s \right)^T
  \]

Continuous-time KF: properties

observation 2:

\[
\begin{align*}
\dot{x}(t) &= Ax(t) + Bu(t) + B_w w(t) \\
y(t) &= Cx(t) + v(t)
\end{align*}
\]

\( \Rightarrow \Phi_{yy}(\omega) = G(j\omega) W G^T (-j\omega) + V \)

from observations 1 and 2:

\[
\left[ I + C (j\omega - A)^{-1} F_s \right] V \left[ I + C (-j\omega - A)^{-1} F_s \right]^T = V + G(j\omega) W G^T (-j\omega)
\]

thus says

\[
\Phi_{\mathbf{e}_y,\mathbf{e}_y}(\omega) = \left[ I + C (j\omega - A)^{-1} F_s \right]^{-1} \Phi_{\mathbf{yy}}(\omega) \left[ I + C (-j\omega - A)^{-1} F_s \right]^{-1} \left( I + C (j\omega - A)^{-1} F_s \right)^T
\]

\[= V \]

namely, the estimation error is white!
Continuous-time KF: symmetric root locus

taking determinants of RDE gives:

\[
\begin{align*}
&\det \left[ I + C (sI - A)^{-1} F_s \right] \det V \det \left[ I + C (-sI - A)^{-1} F_s \right]^T \\
&\quad = \det \left[ V + G(s) W G^T (-s) \right]
\end{align*}
\]

for single-output systems:

\[
\det \left[ I + C (sI - A)^{-1} F_s \right] \det \left[ I + C (-sI - A)^{-1} F_s \right]^T = 1 + G(s) \frac{W}{V} G^T (-s)
\]

Determines the KF eigenvalues:

\[
\begin{align*}
\det \left[ I + C (sI - A)^{-1} F_s \right] &= \det \left[ I + (sI - A)^{-1} F_s C \right] \\
&= \det \left[ (sI - A)^{-1} \right] \det [sI - A + F_s C] \\
&= \frac{\det [sI - (A - F_s C)]}{\det (sI - A)}
\end{align*}
\]

Hence looking at \( 1 + G(s) \frac{W}{V} G^T (-s) \), we have:

- \( W/V \to 0 \): KF poles \( \to \) stable poles of \( G(s) G^T (-s) \)
- \( W/V \to \infty \): KF poles \( \to \) stable zeros of \( G(s) G^T (-s) \)
Summary

1. Big picture

2. Problem statement

3. Discrete-time KF
   - Gain update
   - Covariance update
   - Steady-state KF
   - Duality with LQ

4. Continuous-time KF
   - Solution
   - Steady-state solution and conditions
   - Properties: return difference equality, symmetric root locus...
Lecture 6: Linear Quadratic Gaussian (LQG) Control

Big picture
LQ when there is Gaussian noise
LQG
Steady-state LQG

Big picture
in deterministic control design:
  ▶ state feedback: arbitrary pole placement for controllable systems
  ▶ observer provides (when system is observable) state estimation when not all states are available
  ▶ separation principle for observer state feedback control
we have now learned:
  ▶ LQ: optimal state feedback which minimizes a quadratic cost about the states
  ▶ KF: provides optimal state estimation
in stochastic control:
  ▶ the above two give the linear quadratic Gaussian (LQG) controller
Big picture

plant:
\[ x(k + 1) = Ax(k) + Bu(k) + B_w w(k) \]
\[ y(k) = Cx(k) + v(k) \]

assumptions:
- \( w(k) \) and \( v(k) \) are independent, zero mean, white Gaussian random processes, with
  \[ E[w(k)w^T(k)] = W, \ E[v(k)v^T(k)] = V \]
- \( x(0) \) is a Gaussian random vector independent of \( w(k) \) and \( v(k) \), with
  \[ E[x(0)] = x_0, \ E[(x(0) - x_0)(x(0) - x_0)^T] = X_0 \]

LQ when there is noise

Assume all states are accessible in the plant
\[ x(k + 1) = Ax(k) + Bu(k) + B_w w(k) \]

The original LQ cost
\[ 2J = x^T(N)Sx(N) + \sum_{j=0}^{N-1} \left\{ x^T(j)Qx(j) + u^T(j)Ru(j) \right\} \]

is no longer valid due to the noise term \( w(k) \).

Instead, consider a stochastic performance index:
\[
J = \mathbb{E}_{\{x(0), w(0), \ldots, w(N-1)\}} \left\{ x^T(N)Sx(N) + \sum_{j=0}^{N-1} [x^T(j)Qx(j) + u^T(j)Ru(j)] \right\}
\]

with \( S \succeq 0, \ Q \succeq 0, \ R \succ 0 \)
LQ with noise and exactly known states

solution via stochastic dynamic programming:

Define “cost to go”:

\[
J_k(x(k)) \triangleq \mathbb{E}_{W_k^+} \left\{ x^T(N)Sx(N) + \sum_{j=k}^{N-1} [x^T(j)Qx(j) + u^T(j)Ru(j)] \right\},
\]

\[
W_k^+ = \{ w(k), \ldots, w(N-1) \}
\]

We look for the optima under control \( U_k^+ = \{ u(k), \ldots, u(N-1) \} \):

\[
J_o^k(x(k)) = \min_{U_k^+} J_k(x(k))
\]

\( \triangleright \) the ultimate optimal cost is

\[
J^o = \mathbb{E}_{x(0)} \left[ \min_{U_0^+} J_0(x(0)) \right]
\]

LQ with noise and exactly known states

solution via stochastic dynamic programming:

iteration on optimal cost to go:

\[
J_o^k(x(k)) = \min_{U_k^+} \mathbb{E}_{W_k^+} \left\{ x^T(N)Sx(N) + x^T(k)Qx(k) + u^T(k)Ru(k) + \sum_{j=k+1}^{N-1} [x^T(j)Qx(j) + u^T(j)Ru(j)] \right\}
\]

\[
= \min_{U_{k+1}^+} \min_{u(k)} \mathbb{E}_{W_k^+} \left\{ x^T(k)Qx(k) + u^T(k)Ru(k) + \mathbb{E}_{W_k^+} \left[ x^T(N)Sx(N) + \sum_{j=k+1}^{N-1} [x^T(j)Qx(j) + u^T(j)Ru(j)] \right] \right\}
\]

\[
= \min_{u(k)} \left\{ x^T(k)Qx(k) + u^T(k)Ru(k) + \mathbb{E}_{W_{k+1}^+} \left[ x^T(N)Sx(N) + \sum_{j=k+1}^{N-1} [x^T(j)Qx(j) + u^T(j)Ru(j)] \right] \right\}
\]

\[
= \min_{u(k)} \left\{ \mathbb{E}_{w(k)} \left[ x^T(k)Qx(k) + u^T(k)Ru(k) + \mathbb{E}_{W_{k+1}^+} \left[ J_o^{k+1}(x(k+1)) \right] \right] \right\}
\]

\( \triangleright \) (1) to (2): \( x(k) \) does not depend on \( w(k), w(k+1), \ldots, w(N-1) \)
Lecture 6: Linear Quadratic Gaussian (LQG) Control

LQ with noise and exactly known states
solution via stochastic dynamic programming: induction

\[ J_k^0(x(k)) = \min_{u(k)} \left\{ x^T(k)Qx(k) + u^T(k)Ru(k) + \mathbb{E}_{w(k)} \left[ J_{k+1}^0(x(k+1)) \right] \right\} \]

at time \( N \):
\[ J_N^0(x(N)) = x^T(N)Sx(N) \]

assume at time \( k+1 \):
\[ J_{k+1}^0(x(k+1)) = x^T(k+1)P(k+1)x(k+1) + b(k+1) \]

cost in a standard LQ due to noise

then at time \( k \):
\[ J_k^0(x(k)) = \min_{u(k)} \left\{ x^T(k)Qx(k) + u^T(k)Ru(k) + \mathbb{E}_{w(k)} \left[ x^T(k+1)P(k+1)x(k+1) + b(k+1) \right] \right\} \]

next: use system dynamics \( x(k+1) = Ax(k) + Bu(k) + B_ww(k) \)

Lecture 6: Linear Quadratic Gaussian (LQG) Control

LQ with noise and exactly known states

after some algebra:
\[ J_k^0(x(k)) = \mathbb{E} \min_{w(k)} \{ x^T(k) \left[ Q + A^TP(k+1)A \right] x(k) \}
+ u^T(k) \left[ R + B^TP(k+1)B \right] u(k) + 2x^T(k)A^TP(k+1)Bu(k)
+ 2x^T(k)A^TP(k+1)B_ww(k) + 2u^T(k)B^TP(k+1)B_ww(k) + \mathbb{E}_{w(k)} \left[ w(k)^T B_w^TP(k+1)B_ww(k) + b(k+1) \right] \]

\( w(k) \) is white and zero mean \( \Rightarrow \):
\[ \mathbb{E}_{w(k)} \left\{ 2x^T(k)A^TP(k+1)B_ww(k) + 2u^T(k)B^TP(k+1)B_ww(k) \right\} = 0 \]

\[ \mathbb{E}_{w(k)} \left\{ w(k)^T B_w^TP(k+1)B_ww(k) \right\} \] equals
\[ \text{Tr} \left\{ \mathbb{E}_{w(k)} \left[ B_w^TP(k+1)B_ww(k)w(k)^T \right] \right\} = \text{Tr} \left[ B_w^TP(k+1)B_wW \right] \]

other terms: not random w.r.t. \( w(k) \); can be taken outside of \( \mathbb{E}_{w(k)} \)
LQ with noise and exactly known states

therefore

\[
J^o_k(x(k)) = \min_{u(k)} \{ x^T(k) \left[ Q + A^T P(k+1) A \right] x(k) \\
+ u^T(k) \left[ R + B^T P(k+1) B \right] u(k) + 2x^T(k) A^T P(k+1) Bu(k) \}
+ \text{Tr} \left[ B_w^T P(k+1) B_w W \right] + b(k+1)
\]

note: the term inside the minimization is a quadratic (actually convex) function of \( u(k) \). Optimization is easily done.

Recall: facts of quadratic functions

▶ consider

\[
f(u) = \frac{1}{2} u^T M u + p^T u + q, \quad M = M^T
\]  

▶ optimality (maximum when \( M \) is negative definite; minimum when \( M \) is positive definite) is achieved when

\[
\frac{\partial f}{\partial u^o} = M u^o + p = 0 \implies u^o = -M^{-1} p
\]

▶ and the optimal cost is

\[
f^o = f(u^o) = -\frac{1}{2} p^T M^{-1} p + q
\]
LQ with noise and exactly known states

\[ J^o_k(x(k)) = \min_{u(k)} \left\{ u^T(k) \left[ R + B^T P(k+1) B \right] u(k) + 2 x^T(k) A^T P(k+1) B u(k) + x^T(k) \left[ Q + A^T P(k+1) A \right] x(k) \right\} + \text{Tr} \left[ B_w^T P(k+1) B_w W \right] + b(k+1) \]

- optimal control law [by using (7)]:
  \[ u^o(k) = - \left[ R + B^T P(k+1) B \right]^{-1} B^T P(k+1) A x(k) \]

- optimal cost [by using (8)]:
  \[ J^o_k(x(k)) = \left\{ -x^T(k) A^T P(k+1) B \left[ R + B^T P(k+1) B \right]^{-1} B^T P(k+1) A x(k) + x^T(k) \left[ Q + A^T P(k+1) A \right] x(k) \right\} + \text{Tr} \left[ B_w^T P(k+1) B_w W \right] + b(k+1) \]

Lecture 6: Linear Quadratic Gaussian (LQG) Control

LQ with noise and exactly known states

Riccati equation:
the optimal cost

\[ J^o_k(x(k)) = \left\{ -x^T(k) A^T P(k+1) B \left[ R + B^T P(k+1) B \right]^{-1} B^T P(k+1) A x(k) + x^T(k) \left[ Q + A^T P(k+1) A \right] x(k) \right\} + \text{Tr} \left[ B_w^T P(k+1) B_w W \right] + b(k+1) \]

can be written as

\[ J^o_k(x(k)) = x^T(k) P(k) x(k) + b(k) \]

with the Riccati equation

\[ P(k) = A^T P(k+1) A - A^T P(k+1) B \left[ R + B^T P(k+1) B \right]^{-1} B^T P(k+1) A + Q \]

and

\[ b(k) = \text{Tr} \left[ B_w^T P(k+1) B_w W \right] + b(k+1) \]

boundary conditions: \( P(N) = S \) and \( b(N) = 0 \)
LQ with noise and exactly known states

- optimal control law and Riccati equation are the same as those in the regular LQ problem
- addition cost is due to $B_w w(k)$:
  \[
  b(k) = \text{Tr} \left[ B_w^T P(k+1) B_w W \right] + b(k+1), \quad b(N) = 0
  \]
- the final optimal cost is
  \[
  J^o(x(0)) = \mathbb{E}_{x(0)} \left[ x^T(0) P(0) x(0) + b(0) \right] \\
  = \mathbb{E}_{x(0)} \left[ (x_o + x(0) - x_o)^T P(0) (x_o + x(0) - x_o) + b(0) \right] \\
  = x_o^T P(0) x_o + \text{Tr} (P(0) X_o) + b(0) \tag{9}
  \]
  where
  \[
  b(0) = \sum_{j=0}^{N-1} \text{Tr} \left[ B_w^T P(j+1) B_w W \right]
  \]

LQG: LQ with noise and inexactly known states

- notice that
  - not all states may be available and there is usually output noise:
    \[
    x(k+1) = Ax(k) + Bu(k) + B_w w(k) \\
    y(k) = Cx(k) + v(k)
    \]
  - when $u$ is a function of $y$, the cost has to also consider the randomness from $V^+_k = \{v(k), \ldots, v(N-1)\}$

  \[
  J = \mathbb{E}_{x(0), W^+_0, V^+_0} \left\{ x^T(N) S x(N) + \sum_{j=0}^{N-1} [x^T(j) Q x(j) + u^T(j) R u(j)] \right\} \tag{10}
  \]
  these motivate the linear quadratic Gaussian (LQG) control problem
LQG solution

only \( y(k) \) is accessible instead of \( x(k) \), some connection has to be built to connect the cost to \( Y_k = \{ y(0), \ldots, y(k) \} \):

\[
\begin{align*}
E \left[ x^T(k) Q x(k) \right] &= E \left\{ E \left[ x^T(k) Q x(k) \mid Y_k \right] \right\} \\
&= E \left\{ E \left[ (x(k) - \hat{x}(k|k) + \hat{x}(k|k))^T Q (x(k) - \hat{x}(k|k) + \hat{x}(k|k)) \mid Y_k \right] \right\} \\
&= E \left\{ E \left[ (x(k) - \hat{x}(k|k))^T Q (x(k) - \hat{x}(k|k)) \mid Y_k + \hat{x}^T(k|k) Q \hat{x}(k|k) \right] \right\} \\
&= E \left\{ E \left[ (x(k) - \hat{x}(k|k))^T Q \hat{x}(k|k) \mid Y_k \right] \right\} + 2E \left\{ E \left[ (x(k) - \hat{x}(k|k))^T Q \hat{x}(k|k) \mid Y_k \right] \right\} \\
&= \text{(11)}
\end{align*}
\]

LQG solution

but \( E[x(k) \mid Y_k] = \hat{x}(k|k) \) and \( \hat{x}(k|k) \) is orthogonal to \( \tilde{x}(k|k) \) (property of least square estimation), so

\[
E \left\{ E \left[ (x(k) - \hat{x}(k|k))^T Q \hat{x}(k|k) \mid Y_k \right] \right\} = E \left[ (x(k) - \hat{x}(k|k))^T Q \hat{x}(k|k) \right] \\
= \text{Tr} \left[ Q \hat{x}(k|k) \tilde{x}^T(k|k) \right] = 0
\]

yielding

\[
\begin{align*}
E \left[ x^T(k) Q x(k) \right] &= E \left\{ E \left[ (x(k) - \hat{x}(k|k))^T Q (x(k) - \hat{x}(k|k)) \mid Y_k + \hat{x}^T(k|k) Q \hat{x}(k|k) \right] \right\} \\
&= E \left\{ \hat{x}^T(k|k) Q \hat{x}(k|k) \right\} \\
&\quad + E \left\{ E \left[ \text{Tr} \left\{ Q (x(k) - \hat{x}(k|k)) (x(k) - \hat{x}(k|k))^T \right\} \mid Y_k \right] \right\} \\
&= E \left[ \hat{x}^T(k|k) Q \hat{x}(k|k) \right] + \text{Tr} \left\{ Q Z(k) \right\}
\end{align*}
\]
LQG solution

the LQG cost (10) is thus

\[ J = \mathbb{E}\left\{ \hat{x}^T(N|N)S\hat{x}(N|N) + \sum_{j=0}^{N-1} [\hat{x}^T(j|j)Q\hat{x}(j|j) + u^T(j)Ru(j)] \right\} \]

\[ + \text{Tr}\{SZ(N)\} + \sum_{j=0}^{N-1} \text{Tr}\{QZ(j)\} \]

independent of the control input hence

\[ \min_{\{u(0),...,u(N-1)\}} J \iff \min_{\{u(0),...,u(N-1)\}} \hat{J} \]

LQG is equivalent to an LQ with exactly know states

consider the equivalent problem to minimize:

\[ \hat{J} = \mathbb{E}\left\{ \hat{x}^T(N|N)S\hat{x}(N|N) + \sum_{j=0}^{N-1} [\hat{x}^T(j|j)Q\hat{x}(j|j) + u^T(j)Ru(j)] \right\} \]

- \( \hat{x}(k|k) \) is fully accessible, with the dynamics:

\[ \hat{x}(k+1|k+1) = \hat{x}(k+1|k) + F(k+1) e_y(k+1) \]
\[ = A\hat{x}(k|k) + Bu(k) + F(k+1) e_y(k+1) \]

- from KF results, \( e_y(k+1) \) is white, Gaussian with covariance:

\[ V + CM(k+1)C^T \]
LQG is equivalent to LQ with exactly known states:

\[ J = E \left\{ x^T(N)Sx(N) + \sum_{j=0}^{N-1} [x^T(j)Qx(j) + u^T(j)Ru(j)] \right\} \]

\[ x(k+1) = Ax(k) + Bu(k) + B_w w(k) \]

\[ u^o(k) = - \left[ R + B^T P(k+1) B \right]^{-1} B^T P(k+1) A x(k) \]

**LQG:**

\[ \hat{J} = E \left\{ \hat{x}^T(N|N)S\hat{x}(N|N) + \sum_{j=0}^{N-1} [\hat{x}^T(j|j)Q\hat{x}(j|j) + u^T(j)Ru(j)] \right\} \]

\[ \hat{x}(k+1|k+1) = A\hat{x}(k|k) + Bu(k) + F(k+1)e_y(k+1) \]

the solution of LQG is thus:

\[ u^o(k) = - \left[ R + B^T P(k+1) B \right]^{-1} B^T P(k+1) A \hat{x}(k|k) \]

\[ P(k) = A^T P(k+1) A - A^T P(k+1) B \left[ R + B^T P(k+1) B \right]^{-1} B^T P(k+1) A + Q \]

---

**Optimal cost of LQG control**

- LQ with known states (see (9)):

\[ x(k+1) = Ax(k) + Bu(k) + B_w w(k) \]

\[ J^o = x_o^T P(0) x_o + \text{Tr}(P(0) X_o) + \sum_{j=0}^{N-1} \text{Tr} \left[ B_w^T P(j+1) B_w W \right] \]

- LQG:

\[ \hat{x}(k+1|k+1) = A\hat{x}(k|k) + Bu(k) + F(k+1)e_y(k+1) \]

\[ \hat{J}^o = x_o^T P(0) x_o + \text{Tr}[P(0) Z(0)] \]

\[ + \sum_{j=0}^{N-1} \text{Tr} \left\{ F^T(j+1)P(j+1)F(j+1)[V + CM(k+1)C^T] \right\} \]

\[ J_o = \hat{J}_o + \sum_{j=0}^{N-1} \text{Tr}\{QZ(j)\} + \text{Tr}\{SZ(N)\} \]
Separation theorem in LQG

KF: an (optimal) observer
LQ: an (optimal) state feedback control
Separation theorem in observer state feedback holds—the closed-loop dynamics contains two separated parts: LQ dynamics plus KF dynamics

Stationary LQG problem

Assumptions: system is time invariant; weighting matrices in performance index is time-invariant; noises are white, Gaussian, wide sense stationary.
Equivalent problem: minimize

$$J' = \lim_{N \to \infty} \frac{J}{N} = \lim_{N \to \infty} E \left\{ \frac{x^T(N)Sx(N)}{N} + \frac{1}{N} \sum_{j=0}^{N-1} [x^T(j)Qx(j) + u^T(j)Ru(j)] \right\}$$

$$= E \left[ x^T(k)Qx(k) + u^T(k)Ru(k) \right]$$
Solution of stationary LQG problem

\[ x(k+1) = Ax(k) + Bu(k) + B_w w(k) \]
\[ y(k) = Cx(k) + v(k) \]
\[ J' = E \left[ x^T(k) Q x(k) + u^T(k) R u(k) \right] \]

the solution is \( u = -K_s \hat{x}(k|k) \): steady-state LQ + steady-state KF

\[ K_s = \left[ R + B^T P_s B \right]^{-1} B^T P_s A \]
\[ P_s = A^T P_s A - A^T P_s B \left[ R + B^T P_s B \right]^{-1} B^T P_s A + Q \]
\[ F_s = M_s C^T \left[ C M_s C^T + V \right]^{-1} \]
\[ M_s = A M_s A^T - A M_s C^T \left[ C M_s C^T + V \right]^{-1} C M_s A^T + B_w W B_w^T \]

stability and convergence conditions of the Riccati equations:

- \( (A, B_w) \) and \( (A, B) \): controllable or stabilizable
- \( (A, C_q) \) and \( (A, C) \): observable or detectable \( (Q = C_q^T C_q) \)

Solution of stationary LQG problem

- stability conditions: guaranteed closed-loop stability and KF stability
- separation theorem: closed-loop eigenvalues come from
  - the \( n \) eigenvalues of LQ state feedback: \( A - B K_s \)
  - the \( n \) eigenvalues of KF: \( A - A F_s C \) (or equivalently \( A - F_s C A \))
- optimal cost:
  \[ J_\infty^o = \text{Tr} \left[ P_s \left( B K_s Z_s A^T + B_w W B_w^T \right) \right] \]  \hspace{1cm} (14)
- exercise: prove (14)
Continuous-time LQG

- **plant:**
  \[
  \dot{x}(t) = Ax(t) + Bu(t) + B_w w(t) \\
  y(t) = Cx(t) + v(t)
  \]

- **assumptions:** \(w(t)\) and \(v(t)\) are Gaussian and white; \(x(0)\) is Gaussian

- **cost:**
  \[
  J = \mathbb{E} \left\{ x^T(t_f) S x(t_f) + \int_{t_0}^{t_f} \left[ x^T(t) Q(t) x(t) + u^T(t) R(t) u(t) \right] dt \right\}
  \]
  where \(S \succeq 0\), \(Q(t) \succeq 0\), and \(R(t) \succ 0\) and the expectation is taken over all random quantities \(\{x(0), w(t), v(t)\}\)

Continuous-time LQG solution

- **Continuous-time LQ:**
  \[
  u(t) = -R^{-1} B^T P(t) \hat{x}(t|t) \\
  \frac{dP}{dt} = A^T P + PA - PBR^{-1} B^T P + Q, \ P(t_f) = S
  \]

- **Continuous-time KF:**
  \[
  \frac{d\hat{x}(t|t)}{dt} = A\hat{x}(t|t) + Bu(t) + F(t)(y(t) - C\hat{x}(t|t)) \]
  \[
  F(t) = M(t)C^T V^{-1}, \ \hat{x}(t_0|t_0) = x_0
  \]
  \[
  \frac{dM}{dt} = AM + MA^T - MC^T V^{-1} CM + B_w W B_w^T, \ M(t_0) = X_0
  \]
Summary

1. Big picture

2. Stochastic control with exactly known state

3. Stochastic control with inexactly known state

4. Steady-state LQG

5. Continuous-time LQG problem
Lecture 7: Principles of Feedback Design

Big picture

- we are pretty familiar with SISO feedback system design and analysis
- state-space designs (LQ, KF, LQG,...): time-domain; good mathematical formulation and solutions based on rigorous linear algebra
- frequency-domain and transfer-function analysis: builds intuition; good for properties such as stability robustness
MIMO closed-loop analysis

signals and transfer functions are vectors and matrices now:
- $r$ (reference) and $y$ (plant output): $m$-dimensional
- $G_p(s)$: $p$ by $m$ transfer function matrix

\[
E(s) = R(s) - (H(s)Y_o(s) + V(s)) = R(s) - \{H(s)G_p(s)G_c(s)E(s) + H(s)G_d(s)D(s) + V(s)\} \tag{1}
\]

MIMO closed-loop analysis

(1) gives
\[
E(s) = (I_m + G_{open}(s))^{-1} R(s)
- (I_m + G_{open}(s))^{-1} H(s)G_d(s)D(s) - (I_m + G_{open}(s))^{-1} V(s)
\]

where the loop transfer function
\[
G_{open}(s) = H(s)G_p(s)G_c(s)
\]

We want to minimize $E^*(s) \triangleq R(s) - Y(k) = E(s) + V(s)$
\[
E^*(s) = (I_m + G_{open}(s))^{-1} R(s)
- (I_m + G_{open}(s))^{-1} H(s)G_d(s)D(s) + (I_m + G_{open}(s))^{-1} G_{open}(s)V(s)
\]

Sensitivity and complementary sensitivity functions:
\[
S(s) \triangleq (I_m + G_{open}(s))^{-1}
T(s) \triangleq (I_m + G_{open}(s))^{-1} G_{open}(s)
\]
Fundamental limitations in feedback design

\[ E^*(s) = S(s)R(s) + T(s)V(s) - S(s)H(s)G_d(s)D(s) \]
\[ Y(s) = R(s) - E^*(s) = T(s)R(s) + \ldots \]

- sensitivity function \( S(s) \): explains disturbance-rejection ability
- complementary sensitivity function \( T(s) \): explains reference tracking and sensor-noise rejection abilities
- fundamental constraint of feedback design:
  \[ S(s) + T(s) = I_m \]
equivalently
\[ S(j\omega) + T(j\omega) = I_m \]
- cannot do well in all aspects: e.g., if \( S(j\omega) \approx 0 \) (good disturbance rejection), \( T(j\omega) \) will be close to identity (bad sensor-noise rejection)

Goals of SISO control design

single-input single-output (SISO) control design:

\[ S(j\omega) = \frac{1}{1 + G_{open}(j\omega)}, \quad T(j\omega) = \frac{G_{open}(j\omega)}{1 + G_{open}(j\omega)} \]

- goals:
  1. nominal stability
  2. stability robustness
  3. command following and disturbance rejection
  4. sensor-noise rejection
- feedback achieves: 1 (Nyquist theorem), 2 (sufficient (gain and phase) margins), and
  - 3: small \( S(j\omega) \) at relevant frequencies (usually low frequency)
  - 4: small \( T(j\omega) \) at relevant frequencies (usually high frequency)
- additional control design for meeting the performance goals: feedforward, predictive, preview controls, etc
SISO loop shaping

typical loop shape (magnitude response of $G_{open}$):

\[ |G_{open}(j\omega)| \]

Low freq. boundary

Crossover freq.

High freq. boundary

SISO loop shaping: stability robustness

the idea of stability margins:

\[ \frac{1}{a} = \text{Gain margin} \]

\[ \alpha = \text{Phase margin} \]

$G_{open}(j\omega)$
SISO loop shaping: stability robustness

the idea of stability margins:

\[
\begin{align*}
\text{Gain} & \quad \omega_g (\text{Gain crossover freq.}) \\
0\,\text{dB} & \quad \omega \\
\text{Phase} & \quad \omega_p (\text{Phase crossover freq.}) \\
-180^\circ & \quad \omega
\end{align*}
\]

\[
\begin{align*}
\text{Gain margin} & \\
\text{Phase margin}
\end{align*}
\]

\[
G_{\text{open}}(j\omega) \text{ should be sufficiently far away from } (-1,0) \text{ for robust stability.}
\]

Commonly there are uncertainties and the actual case is

\[
\tilde{G}_{\text{open}}(s) = G_{\text{open}}(s)[1 + \Delta(s)]
\]

e.g. ignored actuator dynamics in a positioning system:

\[
\tilde{G}_{\text{open}}(s) = G_{\text{open}}(s) \frac{1}{T_\alpha s + 1} = G_{\text{open}}(s) \left[1 - \frac{T_\alpha s}{T_\alpha s + 1}\right]
\]

\[
\Delta(j\omega) = -\frac{T_\alpha j\omega}{T_\alpha j\omega + 1}
\]
SISO loop shaping: stability robustness

if nominal stability holds, robust stability needs

\[ |\Delta(j\omega)G_{open}(j\omega)| = |\tilde{G}_{open}(j\omega) - G_{open}(j\omega)| < \frac{|d(j\omega)|}{1 + G_{open}(j\omega)} \]

\[ \Leftrightarrow \left| \Delta(j\omega) \frac{G_{open}(j\omega)}{1 + G_{open}(j\omega)} \right| < 1 \Leftrightarrow |\Delta(j\omega)T(j\omega)| < 1, \forall \omega \]

Lecture 7: Principles of Feedback Design

SISO loop shaping: stability robustness

if \( |G_{open}(j\omega)| \ll 1 \) then

\[ \left| \Delta(j\omega) \frac{G_{open}(j\omega)}{1 + G_{open}(j\omega)} \right| < 1 \]

approximately means

\[ |G_{open}(j\omega)| < \frac{1}{|\Delta(j\omega)|} \]

Lecture 7: Principles of Feedback Design
MIMO Nyquist criterion

\[ r \xrightarrow{+} e \xrightarrow{\circlearrowleft} G_{\text{open}} \xrightarrow{\circlearrowright} y \]

► assume \( G_{\text{open}} \) is \( m \times m \) and realized by
\[
\frac{dx(t)}{dt} = Ax(t) + Be(t), \quad x \in \mathbb{R}^{m \times 1}
\]
\[
y(t) = Cx(t)
\]

► the closed-loop dynamics is
\[
\begin{align*}
\frac{dx(t)}{dt} &= (A - BC)x(t) + Br(t) \\
y(t) &= Cx(t)
\end{align*}
\]

\[ (2) \]

MIMO Nyquist criterion

(2) gives the closed-loop transfer function
\[
G_{\text{closed}}(s) = C(sI - A + BC)^{-1}B
\]

► closed-loop stability depends on the eigenvalues \( \text{eig}(A - BC) \), which come from
\[
\phi_{\text{closed}}(s) = \det(sI - A + BC) = \det \left\{ (sl - A) \begin{bmatrix} I & (sl - A)^{-1} BC \end{bmatrix} \right\}
\]
\[
= \det(sI - A) \det \left( I + C(sI - A)^{-1} B \right)
\]
\[
= \left\{ \det(sI - A) \right\}_{\text{open loop}} \det (I + G_{\text{open}}(s))
\]

► hence
\[
\frac{\phi_{\text{closed}}(s)}{\phi_{\text{open}}(s)} = \det (I + G_{\text{open}}(s))
\]
MIMO Nyquist criterion

\[
\frac{\phi_{\text{closed}}(s)}{\phi_{\text{open}}(s)} = \det(I + G_{\text{open}}(s)) = \prod_{j=1}^{n_1} (s - p_{\text{cl}}) \prod_{i=1}^{n_2} (s - p_{\text{ol}})
\]

- evaluate \( \det(I + G_{\text{open}}(s)) \) along the D contour \( (R \to \infty) \)
- \( Z \) closed-loop “unstable” eigen values in \( \prod_{j=1}^{n_1} (s - p_{\text{cl}}) \) contribute to \( 2\pi Z \) net increase in phase
- \( P \) open-loop “unstable” eigen values in \( \prod_{j=1}^{n_2} (s - p_{\text{ol}}) \) contribute to \( -2\pi P \) net increase in phase
- stable eigen values do not contribute to net phase change

MIMO Nyquist criterion

the number of counter clockwise encirclements of the origin by \( \det(I + G_{\text{open}}(s)) \) is:

\[ N(0, \det(I + G_{\text{open}}(s)), D) = P - Z \]

stability condition: \( Z = 0 \)

**Theorem (Multivariable Nyquist Stability Criterion)**

the closed-loop system is asymptotically stable if and only if

\[ N(0, \det(I + G_{\text{open}}(s)), D) = P \]

i.e., the number of counterclockwise encirclements of the origin by \( \det(I + G_{\text{open}}(s)) \) along the D contour equals the number of open-loop unstable eigen values (of the A matrix).
MIMO robust stability

Given the nominal model $G_{\text{open}}$, let the actual open loop be perturbed to

$$\tilde{G}_{\text{open}}(j\omega) = G_{\text{open}}(j\omega)[1 + \Delta(j\omega)]$$

where $\Delta(j\omega)$ is the uncertainty (bounded by $\sigma(\Delta(j\omega)) \leq \bar{\sigma}$)

what properties should the nominal system possess in order to have robust stability?

MIMO robust stability

- obviously need a stable nominal system to start with:
  $$N(0, \det(I + G_{\text{open}}(s)), D) = P$$
- for robust stability, we need
  $$N(0, \det(I + G_{\text{open}}(s)(1 + \Delta(s))), D) = P$$ for all possible $\Delta$
- under nominal stability, we need the boundary condition
  $$\det(I + G_{\text{open}}(j\omega)(1 + \Delta(j\omega))) \neq 0$$

Figure: Example Nyquist plot for robust stability analysis
MIMO robust stability

- note the determinant equivalence:

\[
\det(I + G_{\text{open}}(j\omega)(1 + \Delta(j\omega))) = \det(I + G_{\text{open}}(j\omega)) \\
\times \det\left[ I + (I + G_{\text{open}}(j\omega))^{-1} G_{\text{open}}(j\omega) \Delta(j\omega) \right]
\]

- as the system is open-loop asymptotically stable, no poles are on the imaginary, i.e.,

\[
\det(I + G_{\text{open}}(j\omega)) \neq 0
\]

- hence \(\det(I + G_{\text{open}}(j\omega)(1 + \Delta(j\omega))) \neq 0 \iff \)

\[
\det\left[ I + (I + G_{\text{open}}(j\omega))^{-1} G_{\text{open}}(j\omega) \Delta(j\omega) \right] \neq 0 \quad (3)
\]

MIMO robust stability

- intuitively, (3) means \(T(j\omega)\Delta(j\omega)\) should be “smaller than” \(I\)

- mathematically, (3) will be violated if \(\exists x \neq 0\) that achieves

\[
[I + T(j\omega)\Delta(j\omega)]x = 0
\]

\[
\iff T(j\omega)\Delta(j\omega)x = -x \quad (4)
\]

which will make the singular value

\[
\sigma_{\text{max}}[T(j\omega)\Delta(j\omega)] = \max_{v \neq 0} \frac{\|T(j\omega)\Delta(j\omega)v\|_2}{\|v\|_2} \geq \frac{\|T(j\omega)\Delta(j\omega)x\|_2}{\|x\|_2}
\]

- as this cannot happen, we must have

\[
\sigma_{\text{max}}[T(j\omega)\Delta(j\omega)] < 1
\]

It turns out this is both necessary and sufficient if \(\Delta(j\omega)\) is unstructured (can ‘attack’ from any directions). Message: we can design \(G_{\text{open}}\) such that \(\sigma_{\text{max}}[\Delta(j\omega)] < \sigma_{\text{min}}[T^{-1}(j\omega)]\).
Summary

1. Big picture

2. MIMO closed-loop analysis

3. Loop shaping
   SISO case

4. MIMO stability and robust stability
   MIMO Nyquist criterion
   MIMO robust stability
Lecture 8: Discretization and Implementation of Continuous-time Design

Big picture

- Discrete-time frequency response
- Discretization of continuous-time design
- Aliasing and anti-aliasing

why are we learning this:

▶ nowadays controllers are implemented in discrete-time domain
▶ implementation media: digital signal processor, field-programmable gate array (FPGA), etc
▶ either: controller is designed in continuous-time domain and implemented digitally
▶ or: controller is designed directly in discrete-time domain
Frequency response of LTI SISO digital systems

\[ a \sin(\omega T_s k) \xrightarrow{G(z)} b \sin(\omega T_s k + \phi) \] at steady state

- sampling time: \( T_s \)
- \( \phi (e^{j\omega T_s}) \): phase difference between the output and the input
- \( M (e^{j\omega T_s}) = b/a \): magnitude difference

Continuous-time frequency response:

\[ G(j\omega) = G(s)|_{s=j\omega} = |G(j\omega)| e^{j\angle G(j\omega)} \]

Discrete-time frequency response:

\[
G(e^{j\omega T_s}) = G(z)|_{z=e^{j\omega T_s}} = \left| G(e^{j\omega T_s}) \right| e^{j\angle G(e^{j\omega T_s})} = M(e^{j\omega T_s}) e^{j\phi(e^{j\omega T_s})}
\]

Sampling

sufficient samples must be collected (i.e., fast enough sampling frequency) to recover the frequency of a continuous-time sinusoidal signal (with frequency \( \omega \) in rad/sec)

\[ \text{the sampling frequency } = \frac{2\pi}{T_s} \]

- Shannon’s sampling theorem: the Nyquist frequency (\( \frac{\Delta}{T_s} \)) must satisfy

\[ -\frac{\pi}{T_s} < \omega < \frac{\pi}{T_s} \]
Approximation of continuous-time controllers

bilinear transform

formula:

\[
\begin{align*}
    s &= \frac{2}{T_s} \frac{z - 1}{z + 1} \\
    z &= \frac{1 + \frac{T_s}{2}s}{1 - \frac{T_s}{2}s}
\end{align*}
\]  

(1)

intuition:

\[
z = e^{sT_s} = \frac{e^{sT_s/2}}{e^{-sT_s/2}} \approx \frac{1 + \frac{T_s}{2}s}{1 - \frac{T_s}{2}s}
\]

implementation: start with \(G(s)\), obtain the discrete implementation

\[
G_d(z) = G(s)\big|_{s=\frac{2}{T_s} \frac{z - 1}{z + 1}}
\]  

(2)

Bilinear transformation maps the closed left half \(s\)-plane to the closed
unit ball in \(z\)-plane

Stability reservation: \(G(s)\) stable \(\iff\) \(G_d(z)\) stable
Approximation of continuous-time controllers

frequency mismatch in bilinear transform

\[
\frac{2 \ z - 1}{T_s \ z + 1} \bigg|_{z = e^{j\omega T_s}} = 2 \ e^{j\omega T_s/2} \left( \frac{e^{j\omega T_s/2} - e^{-j\omega T_s/2}}{e^{j\omega T_s/2} + e^{-j\omega T_s/2}} \right) = j \frac{2}{T_s} \tan \left( \frac{\omega T_s}{2} \right)
\]

\(G(s)|_{s=j\omega}\) is the true frequency response at \(\omega\); yet bilinear implementation gives,

\[
G_d(e^{j\omega T_s}) = G(s)|_{s=j\omega_v} \neq G(s)|_{s=j\omega}
\]

\[
\begin{align*}
\omega & \quad \text{Tangent line at } \omega = \omega_v = 0 \\
0 & \quad \quad 45^\circ \\
\pi/T & \quad \omega_v
\end{align*}
\]

Approximation of continuous-time controllers

bilinear transform with prewarping

goal: extend bilinear transformation such that

\[
G_d(z)|_{z=e^{j\omega T_s}} = G(s)|_{s=j\omega}
\]

at a particular frequency \(\omega_p\)

solution:

\[
s = p \frac{z - 1}{z + 1}, \quad z = \frac{1 + \frac{1}{p} s}{1 - \frac{1}{p} s}, \quad p = \frac{\omega_p}{\tan \left( \frac{\omega_p T_s}{2} \right)}
\]

which gives

\[
G_d(z) = G(s)|_{s=\frac{\omega_p}{\tan \left( \frac{\omega_p T_s}{2} \right)}} \frac{z-1}{z+1}
\]

and

\[
\frac{\omega_p}{\tan \left( \frac{\omega_p T_s}{2} \right)} \frac{z-1}{z+1} \bigg|_{z=e^{j\omega_p T_s}} = j \frac{\omega_p}{\tan \left( \frac{\omega_p T_s}{2} \right)} \tan \left( \frac{\omega_p T_s}{2} \right)
\]
Approximation of continuous-time controllers
bilinear transform with prewarping

choosing a prewarping frequency $\omega_p$:
- must be below the Nyquist frequency:

$$0 < \omega_p < \frac{\pi}{T_s}$$

- standard bilinear transform corresponds to the case where $\omega_p = 0$
- the best choice of $\omega_p$ depends on the important features in control design

example choices of $\omega_p$:
- at the cross-over frequency (which helps preserve phase margin)
- at the frequency of a critical notch for compensating system resonances

Sampling and aliasing
sampling maps the continuous-time frequency

$$-\frac{\pi}{T_s} < \omega < \frac{\pi}{T_s}$$

onto the unit circle

![Diagram showing sampling and aliasing](image.png)
Sampling and aliasing

sampling also maps the continuous-time frequencies \( \frac{\pi}{T_s} < \omega < 3 \frac{\pi}{T_s} \), \( 3 \frac{\pi}{T_s} < \omega < 5 \frac{\pi}{T_s} \), etc, onto the unit circle.

---

**Example (Sampling and Aliasing)**

\( T_s = 1/60 \) sec (Nyquist frequency 30 Hz).

A continuous-time 10-Hz signal \([10 \text{ Hz} \leftrightarrow 2\pi \times 10 \text{ rad/sec} \in (-\pi/T_s, \pi/T_s)]\)

\[ y_1(t) = \sin(2\pi \times 10t) \]

is sampled to

\[ y_1(k) = \sin \left( 2\pi \times \frac{10}{60} k \right) = \sin \left( 2\pi \times \frac{1}{6} k \right) \]

A 70-Hz signal \([2\pi \times 70 \text{ rad/sec} \in (\pi/T_s, 3\pi/T_s)]\)

\[ y_2(t) = \sin(2\pi \times 70t) \]

is sampled to

\[ y_2(k) = \sin \left( 2\pi \times \frac{70}{60} k \right) = \sin \left( 2\pi \times \frac{1}{6} k \right) \equiv y_1(k)! \]
Anti-aliasing

need to avoid the negative influence of aliasing beyond the Nyquist frequencies

▶ sample faster: make $\pi / T_s$ large; the sampling frequency should be high enough for good control design

▶ anti-aliasing: perform a low-pass filter to filter out the signals $|\omega| > \pi / T_s$

Summary

1. Big picture

2. Discrete-time frequency response

3. Approximation of continuous-time controllers

4. Sampling and aliasing
Sampling example

- continuous-time signal

\[ y(t) = \begin{cases} e^{-at}, & t \geq 0 \\ 0, & t < 0 \end{cases}, \quad a > 0 \]

\[ \mathcal{L} \{y(t)\} = \frac{1}{s + a} \]

- discrete-time sampled signal

\[ y(k) = \begin{cases} e^{-aT_s k}, & k \geq 0 \\ 0, & k < 0 \end{cases} \]

\[ \mathcal{Z} \{y(k)\} = \frac{1}{1 - z^{-1}e^{-aT_s}} \]

- sampling maps the continuous-time pole \( s_i = -a \) to the discrete-time pole \( z_i = e^{-aT_s} \), via the mapping

\[ z_i = e^{s_i T_s} \]
Where are we now?

- LQ: optimal control, guaranteed robust stability under basic assumptions in stationary case
- KF: optimal state estimation, good properties from the duality between LQ and KF
- LQG: LQ+KF with separation theorem
- frequency-domain feedback design principles and implementations

Stability robustness of LQG was discussed in one of the homework problems: the nice robust stability in LQ (good gain and phase margins) is lost in LQG. LQG/LTR is one combined scheme that uses many of the concepts learned so far.
Continuous-time stationary LQG solution

\[ u(t) = -K \hat{x}(t|t) \]
\[ \frac{d \hat{x}(t|t)}{dt} = A\hat{x}(t|t) + Bu(t) + F(y(t) - C\hat{x}(t|t)) \]
\[ = (A - BK - FC)\hat{x}(t|t) + Fy(t) \]

\[ G_c(s) = K(sI - A + BK + FC)^{-1}F \quad (1) \]

Loop transfer recovery (LTR)

Theorem (Loop Transfer Recovery (LTR))

If a \( m \times m \) dimensional \( G(s) \) has only minimum phase transmission zeros, then the open-loop transfer function

\[ G(s)G_c(s) = \left[ C(sI - A)^{-1} B \right] \left[ K(sI - A + BK + FC)^{-1}F \right] \]
\[ \overset{\rho \rightarrow 0}{\longrightarrow} C(sI - A)^{-1}F \quad (2) \]

\( K \) and \( \rho \) are from the LQ \( [(A, B) \) controllable, \( (A, C) \) observable] \]

\[ J = \int_{0}^{\infty} \left( x^T(t)C^T Cx(t) + \rho u^T(t)Nu(t) \right) dt \quad (3) \]

\[ \dot{x}(t) = Ax(t) + Bu(t) \quad (4) \]
Loop transfer recovery (LTR)

\[ \begin{align*}
G_c(s) & \rightarrow G(s) = C(sI - A)^{-1} B \\
\end{align*} \]

converges, as \( \rho \to 0 \), to the target feedback loop

\[ \begin{align*}
C(sI - A)^{-1} F \\
\end{align*} \]

key concepts:
- regard LQG as an output feedback controller
- will design \( F \) such that \( C(sI - A)^{-1} F \) has a good loop shape
- not a conventional optimal control problem
- not even a stochastic control design method

Selection of \( F \) for the target feedback loop

standard KF procedure: given noise properties (\( W, V, \) etc), KF gain \( F \) comes from RE

\[ \begin{align*}
\dot{x}(t) &= Ax(t) + Lw(t), & E[w(t)w^T(t + \tau)] &= I\delta(\tau) \\
y(t) &= Cx(t) + v(t), & E[v(t)v^T(t + \tau)] &= \mu I\delta(\tau) \\
\end{align*} \]

select \( W \) and \( V \) to get a desired \( F \) (hence a fictitious KF problem):

\[ F = \frac{1}{\mu}MC^T, \quad AM + M^T A + LL^T - \frac{1}{\mu}MC^T CM = 0, \quad M \succ 0 \quad (5) \]
The target feedback loop from fictitious KF

\[ \dot{x}(t) = Ax(t) + Lw(t), \quad E[w(t)w^T(t + \tau)] = I\delta(\tau) \]
\[ y(t) = Cx(t) + v(t), \quad E[v(t)v^T(t + \tau)] = \mu I\delta(\tau) \]

Return difference equation for the fictitious KF is

\[ [I_m + G_F(s)][I_m + G_F(-s)]^T = I_m + \frac{1}{\mu} [C\Phi(s)L][C\Phi(-s)L]^T \]

where \( G_F(s) = C(sI - A)^{-1}F \) and \( \Phi(s) = (sI - A)^{-1} \). Then

\[ \sigma[I_m + G_F(j\omega)] = \sqrt{\lambda \left\{ [I_m + G_F(j\omega)][I_m + G_F(-j\omega)]^T \right\}} \]
\[ = \sqrt{1 + \frac{1}{\mu} \{\sigma[C\Phi(j\omega)L]\}^2} \geq 1 \]

The (nice) target feedback loop from fictitious KF

\[ \sigma[I_m + G_F(j\omega)] = \sqrt{\lambda \left\{ [I_m + G_F(j\omega)][I_m + G_F(-j\omega)]^T \right\}} \]
\[ = \sqrt{1 + \frac{1}{\mu} \{\sigma[C\Phi(j\omega)L]\}^2} \geq 1 \]

gives:

\[ \sigma_{\text{max}} S(j\omega) = \sigma_{\text{max}}[I + G_F(j\omega)]^{-1} \leq 1, \text{ namely} \]
\[ \text{no disturbance amplification at any frequency} \]

\[ \sigma_{\text{max}} T(j\omega) = \sigma_{\text{max}}[I - S(j\omega)] \leq 2, \text{ hence,} \]
\[ \text{guaranteed closed loop stable if } \sigma_{\text{max}}\Delta(j\omega) < 1/2 \]
Lecture 10: LQ with Frequency Shaped Cost Function (FSLQ)

Background

Parseval’s Theorem

Frequency-shaped LQ cost function

Transformation to a standard LQ

Big picture

why are we learning this:

▶ in standard LQ, \( Q \) and \( R \) are constant matrices in the cost function

\[
J = \int_{0}^{\infty} \left( x^T(t)Qx(t) + \rho u^T(t)Ru(t) \right) dt
\]  

(1)

▶ how can we introduce more design freedom for \( Q \) and \( R \)?
Theorem (Parseval’s Theorem)

For a square integrable signal $f(t)$ defined on $[0, \infty)$

$$
\int_0^\infty f^T(t)f(t)\,dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} F_T(-j\omega)F(j\omega)\,d\omega
$$

1D case:

$$
\int_0^\infty |f(t)|^2\,dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |F(j\omega)|^2\,d\omega
$$

Intuition: energy in time-domain equals energy in frequency domain

For the general case, $f(t)$ can be acausal. We have

$$
\int_{-\infty}^\infty f^T(t)f(t)\,dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} F_T(-j\omega)F(j\omega)\,d\omega
$$

Discrete-time version:

$$
\sum_{k=-\infty}^{\infty} f^T(k)f(k) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F_T(e^{-j\omega})F(e^{j\omega})\,d\omega
$$

History

Marc-Antoine Parseval (1755-1836):

- French mathematician
- published just five (but important) mathematical publications in total (source: Wikipedia.org)
Frequency-domain LQ cost function

From Parseval’s Theorem, the LQ cost in frequency domain is

\[ J = \int_0^\infty \left( x^T(t)Qx(t) + \rho u^T(t)Ru(t) \right) dt \]  

\[ = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left( X^T(-j\omega)QX(j\omega) + \rho U^T(-j\omega)RU(j\omega) \right) d\omega \]  

Frequency-shaped LQ expands \( Q \) and \( R \) to frequency-dependent functions:

\[ J = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left( X^T(-j\omega)Q(j\omega)X(j\omega) + \rho U^T(-j\omega)R(j\omega)U(j\omega) \right) d\omega \]  

Let

\[ Q(j\omega) = Q_f^T(-j\omega)Q_f(j\omega) \succeq 0, \; X_f(j\omega) = Q_f(j\omega)X(j\omega) \]
\[ R(j\omega) = R_f^T(-j\omega)R_f(j\omega) \succ 0, \; U_f(j\omega) = R_f(j\omega)U(j\omega) \]

(4) becomes

\[ J = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left( X_f^T(-j\omega)X_f(j\omega) + \rho U_f^T(-j\omega)U_f(j\omega) \right) d\omega \]

which is equivalent to (using Parseval’s Theorem again)

\[ J = \int_0^\infty \left( x_f^T(t)x_f(t) + \rho u_f^T(t)u_f(t) \right) dt \]  

(5)
Frequency-domain LQ cost function

Summarizing, we have:

- plant:
  \[
  \begin{align*}
  \dot{x}(t) &= Ax(t) + Bu(t) \\
  y(t) &= Cx(t)
  \end{align*}
  \]  
  (6)

- new cost:
  \[
  J = \int_0^\infty \left( x_f^T(t) x_f(t) + \rho u_f^T(t) u_f(t) \right) dt 
  \]  
  (7)

- filtered states and inputs:
  \[
  x(t) \rightarrow [Q_f(s)] \rightarrow x_f(t), \quad u(t) \rightarrow [R_f(s)] \rightarrow u_f(t)
  \]

We just need to translate the problem to a standard one [which we know (very well) how to solve]

Frequency-domain weighting filters

state filtering

\[
\begin{align*}
   x(t) &\rightarrow [Q_f(s)] \rightarrow x_f(t) \\
  \end{align*}
\]

- a MIMO process in general: if \( x(t) \in \mathbb{R}^n \) and \( x_f(t) \in \mathbb{R}^q \), then \( Q_f(s) \) is a \( q \times n \) transfer function matrix

- \( Q_f(s) \): state filter; designer’s choice; can be selected to meet the desired control action and the performance requirements

- write \( Q_f(s) = C_1(sI - A_1)^{-1}B_1 + D_1 \) in the general state-space realization:
  \[
  \begin{align*}
  \dot{z}_1(t) &= A_1 z_1(t) + B_1 x(t) \\
  x_f(t) &= C_1 z_1(t) + D_1 x(t)
  \end{align*}
  \]  
  (8)
Frequency-domain weighting filters
input filtering

\[ u(t) \xrightarrow{} R_f(s) \xrightarrow{} u_f(t) \]

- \( R_f(s) \): input filter; designer’s choice; can be selected to meet the robustness requirements

- write \( R_f(s) = C_2(sI - A_2)^{-1}B_2 + D_2 \) in the general state-space realization:

\[
\begin{cases}
\dot{z}_2(t) = A_2z_2(t) + B_2u(t) \\
u_f(t) = C_2z_2(t) + D_2u(t)
\end{cases}
\] \hspace{1cm} (9)

Back to time-domain design

Combining (6), (8) and (9) gives the enlarged system:

\[
\frac{d}{dt} \begin{bmatrix} x(t) \\ z_1(t) \\ z_2(t) \end{bmatrix} = \begin{bmatrix} A & 0 & 0 \\ B_1 & A_1 & 0 \\ 0 & 0 & A_2 \end{bmatrix} \begin{bmatrix} x(t) \\ z_1(t) \\ z_2(t) \end{bmatrix} + \begin{bmatrix} B \\ B_1 \\ B_2 \end{bmatrix} u(t) \]

and

\[
x_f(t) = \begin{bmatrix} D_1 & C_1 \\ & 0 \\ & & C_2 \end{bmatrix} \begin{bmatrix} x(t) \\ z_1(t) \\ z_2(t) \end{bmatrix}
\]

\[
u_f(t) = [0 \ 0 \ C_2]x_e(t) + D_2u(t)\]
Summary of solution

With the enlarged system, the cost

\[ J = \int_0^\infty \left( x_f^T(t)x_f(t) + \rho u_f^T(t)u_f(t) \right) dt \]  

(10)

translates to

\[ J = \int_0^\infty \left( x_e^T(t)Q_e x_e(t) + 2u^T(t) \begin{bmatrix} 0 & 0 & \rho D_2^T C_2 \end{bmatrix} x_e(t) + u^T(t) \begin{bmatrix} 0 & \rho D_2^T D_2 \end{bmatrix} u(t) \right) dt \]

\[ Q_e = \begin{bmatrix} D_1^T D_1 & D_1^T C_1 & 0 \\ C_1^T D_1 & C_1^T C_1 & 0 \\ 0 & 0 & \rho C_2^T C_2 \end{bmatrix} \]

▷ solution (see appendix for more details):

\[ u(t) = -R_e^{-1}(B_e^T P_e + N_e)x_e(t) = -Kx(t) - K_1 z_1(t) - K_2 z_2(t) \]

▷ algebraic Riccati equation:

\[ A_e^T P_e + P_e A_e - (B_e^T P_e + N_e)^T R_e^{-1}(B_e^T P_e + N_e) + Q_e = 0 \]

Implementation

structure of the FSLQ system:
Appendix: general LQ solution

Consider LQ problems with cost

$$J = \int_0^\infty \left( x^T(t) C^T C x(t) + 2u^T(t) N x(t) + u^T(t) R u(t) \right) dt \quad (11)$$

and system dynamics

$$\dot{x}(t) = A x(t) + B u(t)$$

- assume \((A, B)\) is controllable/stabilizable and \((A, C)\) is observable/detectable
- the solution of the problem is

$$u(t) = -R^{-1}(B^T P + N)x(t)$$

$$A^T P + PA - (B^T P + N)^T R^{-1}(B^T P + N) + Q = 0$$

Intuition: under the assumptions, we know we can stabilize the system and drive \(x(t)\) to zero. Consider Lyapunov function

$$V(t) = x^T(t) P x(t), \quad P = P^T \succ 0$$

$$\dot{V}(t) = \int_0^\infty \dot{V}(t) dt = \int_0^\infty \left( x^T(t) \left( PA + A^T P \right) x(t) + 2x^T(t) PB u(t) \right) dt$$

Adding (11) on both sides yields

$$V(\infty) - V(0) + J = \int_0^\infty \left( x^T(t) \left( Q + PA + A^T P \right) x(t) + 2x^T(t) \left( PB + N^T \right) u(t) + u^T(t) Ru(t) \right) dt \quad (12)$$

- to minimize the cost, we are going to re-organize the terms in (12) into some “squared” terms
Appendix: general LQ solution

“completing the squares”:

\[ 2x^T(t)(PB + NT)u(t) + u^T(t)Ru(t) = \left\| R^{1/2}u(t) + R^{-1/2}(B^TP + N)x(t) \right\|^2_2 
- x^T(t)(PB + NT)R^{-1}(B^TP + N)x(t) \]

hence (12) is actually

\[ V(\infty) - V(0) + J \]

\[ = \int_0^\infty \left[ x^T(t) \left( Q + PA + A^TP - (PB + NT)R^{-1}(B^TP + N) \right)x(t) 
+ \left\| R^{1/2}u(t) + R^{-1/2}(B^TP + N)x(t) \right\|^2 \right] dt \]

hence \( J_{\text{min}} = V(0) = x^T(0)Px(0) \) is achieved when

\[ Q + PA + A^TP - (PB + NT)R^{-1}(B^TP + N) = 0 \]

and \( u(t) = -R^{-1}(B^TP + N)x(t) \)
Big picture
Stable pole-zero cancellation
Phase error
Zero phase error tracking

why are we learning this:

- two basic control problems: tracking (the reference) and regulation (against disturbances)
- feedback control has performance limitations
- For tracking $r(k)$, ideally we want $G_{\text{closed}}(z^{-1}) = 1$
  - which is not attainable by feedback. We thus need feedforward control.
one naive approach: to let $y(k)$ track $y_d(k)$, we can do

$$r(k) = G^{-1}_{\text{closed}}(z^{-1}) y_d(k) = \frac{z^d A_c(z^{-1})}{B_c(z^{-1})} y_d(k) = \frac{A_c(z^{-1})}{B_c(z^{-1})} y_d(k + d) \quad (1)$$

- **causality**: (1) requires knowledge of $y_d(k)$ for at least $d$ steps ahead (usually not an issue)
- **stability**: poles of $G^{-1}_{\text{closed}}(z^{-1})$, i.e., zeros of $G_{\text{closed}}(z^{-1})$, must be all stable (usually an issue)
- **robustness**: the model $G_{\text{closed}}(z^{-1})$ needs to be accurate
The cancellable parts in \( G_{\text{closed}}(z^{-1}) \)

\[ y_d(k) \xrightarrow{\text{Feedforward}} \quad G_{\text{closed}}(z^{-1}) = \frac{z^{-d} B_c(z^{-1})}{A_c(z^{-1})} \xrightarrow{} y(k) \]

- \( G_{\text{closed}}(z^{-1}) \) is always stable \( \Rightarrow A_c(z^{-1}) \) can be fully canceled
- \( B_c(z^{-1}) \) may contain **uncancellable parts** (zeros on or outside the unit circle)
- partition \( G_{\text{closed}}(z^{-1}) \) as

\[
G_{\text{closed}}(z^{-1}) = \frac{z^{-d} B_c(z^{-1})}{A_c(z^{-1})} = \frac{z^{-d} B_c^+(z^{-1})}{A_c(z^{-1})} \frac{B_c^-(z^{-1})}{A_c(z^{-1})}
\]

(2)

Stable pole-zero cancellation

\[ y_d(k) \xrightarrow{\text{Feedforward}} \quad G_{\text{closed}}(z^{-1}) = \frac{z^{-d} B_c(z^{-1})}{A_c(z^{-1})} \xrightarrow{} y(k) \]

feedforward via stable pole-zero cancellation:

\[
G_{\text{spz}}(z^{-1}) = \frac{z^d A_c(z^{-1})}{B_c^+(z^{-1})} \frac{1}{B_c^-(1)}
\]

(3)

where \( B_c^-(1) = B_c^-(z^{-1})|_{z^{-1}=1} \)
- \( B_c^-(1) \) makes the overall DC gain from \( y_d(k) \) to \( y(k) \) to be one:

\[
G_{y_d \rightarrow y}(z^{-1}) = G_{\text{spz}}(z^{-1}) G_{\text{closed}}(z^{-1}) = \frac{B_c^-(z^{-1})}{B_c^-(1)}
\]

- example: \( B_c^-(z^{-1}) = 1 + z^{-1} \), \( B_c^-(1) = 2 \), then

\[
G_{y_d \rightarrow y}(z^{-1}) = \frac{1 + z^{-1}}{2} \text{: a moving-average low-pass filter}
\]
Stable pole-zero cancellation properties of $G_{y_d \to y}(z^{-1}) = \frac{1+z^{-1}}{2}$:

- there is always a **phase error** in tracking
- example: if $y_d(k) = \alpha k$ (a ramp signal)

$$y(k) = G_{y_d \to y}(z^{-1}) y_d(k) = \alpha k - \frac{\alpha}{2}$$

which is always delayed by a factor of $\alpha/2$

---

Zero Phase Error Tracking (ZPET)

Zero Phase Error Tracking (ZPET): extend (3) by adding a $B_c^-(z)$ part

$$G_{ZPET}(z^{-1}) = \frac{z^d A_c(z^{-1}) B_c^-(z)}{B_c^+(z^{-1}) B_c^-(1)^2}$$ (4)

where $B_c^-(z) = b_{c0}^- + b_{c1}^- z + \cdots + b_{cs}^- z^s$ if $B_c^-(z^{-1}) = b_{c0}^- + b_{c1}^- z^{-1} + \cdots + b_{cs}^- z^{-s}$

- overall dynamics between $y(k)$ and $y_d(k)$:

$$G_{y_d \to y}(z^{-1}) = G_{closed}(z^{-1}) G_{ZPET}(z^{-1}) = \frac{B_c^-(z) B_c^-(z^{-1})}{[B_c^-(1)]^2}$$ (5)

---

Lecture 11: Feedforward Control, Zero Phase Error Tracking
Zero Phase Error Tracking (ZPET)
understanding (5):
▶ the frequency response always has zero phase error:
\[ B^{-}_c(e^{j\omega}) = B^{-}_c(e^{-j\omega}) \] (a complex conjugate pair)
▶ \[ B^{-}_c(1)^2 \] normalizes \( G_{yd \rightarrow y} \) to have unity DC gain:
\[ G_{yd \rightarrow y}(e^{-j\omega}) \bigg|_{\omega = 0} = \frac{B^{-}_c(e^{j\omega}) \bigg|_{\omega = 0} B^{-}_c(e^{-j\omega}) \bigg|_{\omega = 0}}{[B^{-}_c(1)]^2} = \frac{[B^{-}_c(1)]^{-1}}{[B^{-}_c(1)]^2} \]
▶ example: \( B^{-}_c(z^{-1}) = 1 + z^{-1} \), then
\[ G_{yd \rightarrow y} (z^{-1}) = \frac{(1+z)(1+z^{-1})}{4} \]
▶ if \( y_d(k) = \alpha k \), then \( y(k) = \alpha k \! \)!
▶ fact: ZPET provides perfect tracking to step and ramp signals at steady state (see ME 233 course reader)

Zero Phase Error Tracking (ZPET)
Example: \( B^{-}_c(z^{-1}) = 1 + 2z^{-1} \)
\[ G_{yd \rightarrow y} (z^{-1}) = \frac{(1+2z)(1+2z^{-1})}{9} = \frac{2z + 5 + 2z^{-1}}{9} \]

Figure: Bode Plot of \( \frac{2z + 5 + 2z^{-1}}{9} \)

Lecture 11: Feedforward Control, Zero Phase Error Tracking
Implementation

\[
y_d(k) \xrightarrow{G_{ZPET}(z^{-1})} r(k) \xrightarrow{G_{closed}(z^{-1})} y(k)
\]

\[
G_{closed}(z^{-1}) = \frac{z^{-d}B_c^+(z^{-1})B_c^-(z^{-1})}{A_c(z^{-1})}
\]

\[
r(k) = \left[ \frac{z^d A_c(z^{-1}) B_c^-(z)}{B_c^+(z^{-1}) B_c^-(1)^2} \right] y_d(k)
\]

- \(z^d\) is not causal ⇒ do instead

\[
r(k) = \left[ \frac{A_c(z^{-1}) B_c^-(z)}{B_c^+(z^{-1}) B_c^-(1)^2} \right] y_d(k + d)
\]

- \(B_c^-(z) = b_{c0}^- + b_{c1}^- z + \cdots + b_{cs}^- z^s\) is also not causal ⇒ do instead

\[
r(k) = \left[ \frac{A_c(z^{-1}) z^{-s} B_c^-(z)}{B_c^+(z^{-1}) B_c^-(1)^2} \right] y_d(k + d + s)
\]

- at time \(k\), requires \(y_d(k + d + s)\): \(d + s\) steps preview of the desired output

Example:

\[
G_{closed}(z^{-1}) = \frac{z^{-1}(1 + 2z^{-1})}{3}
\]

- without feedforward control:
Implementation

Example:

\[ G_{\text{closed}}(z^{-1}) = \frac{z^{-1}(1+2z^{-1})}{3} \]

▶ with ZPET feedforward:

ZPET extensions:

▶ standard form:

\[ G_{\text{ZPET}}(z^{-1}) = \frac{z^d A_c(z^{-1})}{B_c^+(z^{-1})} \frac{B_c^-(z)}{B_c^-(1)^2} \]


\[ G_{\text{ZPET}}(z^{-1}) = \frac{z^d A_c(z^{-1})}{B_c^+(z^{-1})} \frac{B_c^-(z)}{B_c^-(1)^2} \frac{(z^{-1} - b)(z - b)}{(1 - b)^2}, \quad 0 < b < 1 \]

▶ remark: \( \frac{(z^{-1} - b)(z - b)}{(1 - b)^2}, \quad 0 < b < 1 \) is a high-pass filter with unity DC gain
Lecture 12: Preview Control

Review: optimal tracking
We consider controlling the system

\[ x(k+1) = Ax(k) + Bu(k) \]  \hspace{1cm} (1)
\[ y(k) = Cx(k) \]

where

\[ x \in \mathbb{R}^n, \ u \in \mathbb{R}^m, \ y \in \mathbb{R}^r \]

Optimal tracking with full reference information (homework 1):

\[
\min_{U_0} J := \frac{1}{2} \left[ y_d(N) - y(N) \right]^T S \left[ y_d(N) - y(N) \right] \\
+ \frac{1}{2} \sum_{k=0}^{N-1} \left( \left[ y_d(k) - y(k) \right]^T Q_y [y_d(k) - y(k)] + u(k)^T Ru(k) \right) \hspace{1cm} (2)
\]

\[
u^o(k) = - \left[ R + B^T P(k+1)B \right]^{-1} B^T \left[ P(k+1)Ax(k) + b^T(k+1) \right] \hspace{1cm} (3)
\]

\[ J_k^o(x(k)) = \frac{1}{2} x^T(k)P(k)x(k) + b(k)x(k) + c(k) \hspace{1cm} (4)\]
Overview of preview control

Preview control considers the same cost-function structure, with:

- **a $N_p$-step preview window**: the desired output signals in this window are known

- **post preview window**: after the preview window we assume we no longer know the desired output (due to, e.g., limited vision in the example of vehicle driving), but we assume the reference is generated from some models.

- e.g. (deterministic model)

$$y_d(k + N_p + l) = y_d(k + N_p), \quad l > 0$$  \hfill (5)

- or (stochastic model):

$$x_d(k + 1) = A_d x_d(k) + B_d w_d(k)$$

$$y_d(k) = C_d x_d(k)$$

where $w_d(k)$ is white and Gaussian distributed. Note: if $A_d = I$, $B_d = 0$, $C_d = I$, $x_d(k + N_p) = y_d(k + N_p)$, then (6)$\iff$(5).

Structuring the future knowledge

Knowledge of the future trajectory can be built into

$$
\begin{bmatrix}
    y_d(k + 1) \\
    y_d(k + 2) \\
    \vdots \\
    y_d(k + N_p) \\
    x_d(k + N_p + 1)
\end{bmatrix}
= 
\begin{bmatrix}
    0 & I & 0 & \ldots & 0 \\
    0 & 0 & I & \ddots & \vdots \\
    \vdots & \ddots & \ddots & \ddots & 0 \\
    0 & \ldots & 0 & 0 & C_d \\
    0 & \ldots & 0 & 0 & A_d
\end{bmatrix}
\begin{bmatrix}
    y_d(k) \\
    y_d(k + 1) \\
    \vdots \\
    y_d(k + N_p - 1) \\
    x_d(k + N_p)
\end{bmatrix}
+ 
\begin{bmatrix}
    0 \\
    0 \\
    \vdots \\
    0 \\
    \overline{w_d(k)}
\end{bmatrix}
\begin{bmatrix}
    0 \\
    0 \\
    \vdots \\
    0 \\
    B_d
\end{bmatrix}
$$

\hfill (7)
The cost function

At time $k$

$$J_k = \frac{1}{1+N} \mathbb{E} \left\{ (y(N+k) - y_d(N+k))^T S_y (y(N+k) - y_d(N+k)) \right. $$

$$ + \sum_{j=0}^{N-1} \left[ (y(j+k) - y_d(j+k))^T Q_y (y(j+k) - y_d(j+k)) \right. $$

$$ \left. \left. + u(j+k)^T R u(j+k) \right\} \right\} \quad (8)$$

- a moving horizon cost
- only $u(k)$ is applied to the plant after we find a solution to minimize $J_k$.
- in deterministic formulation, we remove the expectation sign. In stochastic formulation, expectation is taken with respect to

$$\{w_d(k+N_p), w_d(k+N_p+1), \ldots, w_d(k+N-1)\}$$

for the minimization of $J_k$.

Augmenting the system

Augmenting the plant with the reference model yields

$$\begin{bmatrix} x(k+1) \\ Y_d(k+1) \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & A_{Y_d} \end{bmatrix} \begin{bmatrix} x(k) \\ Y_d(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} 0 \\ B_{Y_d} \end{bmatrix} \bar{w}_d(k)$$

$$A_e \quad x_e(k) \quad B_e \quad B_{w,e}$$

(9)

and

$$y(j+k) - y_d(j+k) = C x(k+j) - [I, 0, \ldots, 0] Y_d(k+j)$$

$$= [C, -I, 0, \ldots, 0] x_e(k+j)$$

$$C_e$$
Translation to a standard LQ

\[ y(j + k) - y_d(j + k) = [C, -I, 0, \ldots, 0] x_e(k + j) \]

Hence

\[
J_k = \frac{1}{1 + N} \mathbb{E} \left\{ (y(N + k) - y_d(N + k))^T S_y (y(N + k) - y_d(N + k)) + \sum_{j=0}^{N-1} [(y(j + k) - y_d(j + k))^T Q_y (y(j + k) - y_d(j + k)) + u(j + k)^T R_u(j + k)] \right\}
\]

is nothing but

\[
J_k = \frac{1}{1 + N} \mathbb{E} \left\{ x_e(N + k)^T C_e^T S_y C_e x_e(N + k) + \sum_{j=0}^{N-1} [x_e(j + k)^T C_e^T Q_y C_e x_e(j + k) + u(j + k)^T R_u(j + k)] \right\}
\]

(10)

Solution of the preview control problem

The equivalent formulation

\[ x_e(k + 1) = A_e x_e(k) + B_e u(k) + B_{w,e} \tilde{w}_d(k) \]

\[
J_k = \frac{1}{1 + N} \mathbb{E} \left\{ x_e(N + k)^T C_e^T S_y C_e x_e(N + k) + \sum_{j=0}^{N-1} [x_e(j + k)^T C_e^T Q_y C_e x_e(j + k) + u(j + k)^T R_u(j + k)] \right\}
\]

is a standard LQ (deterministic formulation) or a standard LQG problem with exactly known state (stochastic formulation). Hence

\[
u^o(k) = - \left[ B_e^T P(k + 1) B_e + R \right]^{-1} B_e^T P(k + 1) A_e x_e(k)
\]

\[
P(k) = - A_e^T P(k + 1) B_e \left[ B_e^T P(k + 1) B_e + R \right]^{-1} B_e^T P(k + 1) A_e + A_e^T P(k + 1) A_e + C_e^T Q_y C_e
\]

where \( P(k + N) = C_e^T S_y C_e \)
Remark

Let \( u^o(k) = K_e x_e(k) = [ K_{e1}(k) \ K_{e2}(k) ] x_e(k) \), the closed-loop matrix is

\[
A_e - B_e K_e(k) = \begin{bmatrix} A & 0 \\ 0 & A_{Y_d} \end{bmatrix} - \begin{bmatrix} B \\ 0 \end{bmatrix} [ K_{e1}(k) \ K_{e2}(k) ]
\]

\[
= \begin{bmatrix} A - B K_{e1}(k) & -B K_{e2}(k) \\ 0 & A_{Y_d} \end{bmatrix}
\]

- the closed-loop eigenvalue from \( A_{Y_d} \) will not be changed.
- The Riccati equation may look ill conditioned if \( A_{Y_d} \) contains marginally stable eigenvalues. This, however, does not cause a problem. For additional details, see the course reader or come to the instructor’s office hour.

Summary

1. Big picture
2. Formulation of the optimal control problem
3. Translation to a standard LQ
Big picture
review of integral control in PID design
example:

\[ E(s) \xrightarrow{C(s)} P(s) \xrightarrow{} Y(s) \]

where

\[ P(s) = \frac{1}{ms + b}, \quad C(s) = k_p + k_i \frac{1}{s} + k_ds, \quad k_p, k_i, k_d > 0 \]

- the integral action in PID control perfectly rejects (asymptotically) constant disturbances \( D(s) = d_o/s \):

\[ E(s) = \frac{-P(s)}{1 + P(s)C(s)} D(s) = \frac{-d_o}{(m + k_d)s^2 + (k_p + b)s + k_i} \]

\[ \Rightarrow e(t) \to 0 \]
Big picture
review of integral control in PID design

the “structure” of the reference/disturbance is built into the integral controller:

- controller:

\[
C(s) = k_p + k_i \frac{1}{s} + k_d s = \frac{1}{s} (k_d s^2 + k_p s + k_i)
\]

- constant disturbance:

\[
d(t) = d_o \iff \mathcal{L} \{d(t)\} = \frac{1}{s} d_o
\]

General case: internal model principle (IMP)

Theorem (Internal Model Principle)

Assume \( B_p(s) = 0 \) and \( A_d(s) = 0 \) do not have common roots. If the closed loop is asymptotically stable, and \( A_c(s) \) can be factorized as \( A_c(s) = A_d(s) A'_c(s) \), then the disturbance is asymptotically rejected.
Internal model principle

discrete-time case:

**Theorem (Discrete-time IMP)**

Assume \( B_p(z^{-1}) = 0 \) and \( A_d(z^{-1}) = 0 \) do not have common zeros. If the closed loop is asymptotically stable, and \( A_c(z^{-1}) \) can be factorized as \( A_c(z^{-1}) = A_d(z^{-1}) A'_c(z^{-1}) \), then the disturbance is asymptotically rejected.

**Proof:** analogous to the continuous-time case.
Internal model principle
the disturbance structure:

\[
R(z^{-1}) + E(z^{-1}) \rightarrow B_c(z^{-1}) \quad \frac{B_c(z^{-1})}{A_c(z^{-1})} \\
\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
\sum \quad \sum \quad \sum \quad \sum \\
\rightarrow Y(z^{-1})
\]

\[
D(z^{-1}) = \frac{B_d(z^{-1})}{A_d(z^{-1})}
\]

example disturbance structures:

<table>
<thead>
<tr>
<th>( d(k) )</th>
<th>( A_d(z^{-1}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>constant ( d_o )</td>
<td>( \frac{1}{1-z^{-1}} )</td>
</tr>
<tr>
<td>( \cos(\omega_0 k) ) and ( \sin(\omega_0 k) )</td>
<td>( 1 - 2z^{-1}\cos(\omega_0) + z^{-2} )</td>
</tr>
<tr>
<td>shifted ramp signal ( d(k) = \alpha k + \beta )</td>
<td>( 1 - 2z^{-1} + z^{-2} )</td>
</tr>
<tr>
<td>periodic: ( d(k) = d(k - N) )</td>
<td>( 1 - z^{-N} )</td>
</tr>
</tbody>
</table>

observations:

- the controller contains a “counter disturbance” generator
- high-gain control: the open-loop frequency response

\[
P(e^{-j\omega}) C(e^{-j\omega}) = \frac{e^{-d_j\omega} B_p(e^{-j\omega}) B_c(e^{-j\omega})}{A_p(e^{-j\omega}) A_c'(e^{-j\omega}) A_d(e^{-j\omega})}
\]

is large at frequencies where \( A_d(e^{-j\omega}) = 0 \)

- \( D(z^{-1}) = \frac{B_d(z^{-1})}{A_d(z^{-1})} \) means \( d(k) \) is the impulse response of \( \frac{B_d(z^{-1})}{A_d(z^{-1})} \):

\[
A_d(z^{-1}) d(k) = B_d(z^{-1}) \delta(k)
\]
Repetitive control

Repetitive control focus on attenuating periodic disturbances with

\[ A_d(z^{-1}) = 1 - z^{-N} \]

Control structure:

\[
\begin{align*}
R(z^{-1}) + E(z^{-1}) &\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \ quad \end{align*}
\]

It remains to design \( B_c(z^{-1}) \) and \( A'_c(z^{-1}) \). We discuss two methods:

- **pole placement**
- (partial) cancellation of plant dynamics: *prototype repetitive control*
1, Pole placement: prerequisite

**Theorem**

Consider \( G(z) = \frac{\beta(z)}{\alpha(z)} = \frac{\beta_1 z^{n-1} + \beta_2 z^{n-2} + \cdots + \beta_n}{z^n + \alpha_1 z^{n-1} + \cdots + \alpha_n} \). \( \alpha(z) \) and \( \beta(z) \) are coprime (no common roots) iff \( S \) (Sylvester matrix) is nonsingular:

\[
S = \begin{bmatrix}
1 & 0 & \cdots & 0 & \beta_1 & 0 & \cdots & 0 \\
\alpha_1 & 1 & \cdots & \beta_2 & \beta_1 & \cdots & \cdots \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \ddots & \cdots \\
\vdots & \vdots & \cdots & 0 & \cdots & \cdots & \cdots & 0 \\
\vdots & \vdots & \cdots & \alpha_1 & 1 & \beta_{n-1} & \cdots & 0 \\
\alpha_{n-1} & \alpha_1 & \beta_n & \cdots & \cdots & \cdots & \cdots & \beta_1 \\
\alpha_n & \cdots & 0 & \beta_n & \cdots & \cdots & \cdots & \cdots \\
0 & \alpha_n & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\vdots & \vdots & \cdots & \alpha_{n-1} & \cdots & \cdots & \cdots & \beta_n \\
0 & \cdots & 0 & \alpha_n & 0 & \cdots & \cdots & 0 & \beta_n \\
\end{bmatrix}_{(2n-1) \times (2n-1)}
\]

Example:

\[
G(z) = \frac{\beta_1 z^{n-1} + \beta_2 z^{n-2} + \cdots + \beta_n}{z^n + \alpha_1 z^{n-1} + \cdots + \alpha_n} = \frac{z^{n-1} + \alpha_1 z^{n-2} + \cdots + \alpha_{n-1}}{z^n + \alpha_1 z^{n-1} + \cdots + \alpha_{n-1} z + 0}
\]

i.e.

\[
\begin{align*}
\beta_1 &= 1 \\
\beta_i &= \alpha_{i-1} \quad \forall i \geq 2 \\
\alpha_n &= 0
\end{align*}
\]

\( \alpha(z) \) and \( \beta(z) \) are not coprime, and \( S \) is clearly singular.
1, Pole placement: big picture

\[
R(z^{-1}) + E(z^{-1}) = \frac{B_c(z^{-1})}{A_c(z^{-1})A_d(z^{-1})} + \frac{z^{-d}B_p(z^{-1})}{A_p(z^{-1})} Y(z^{-1})
\]

Disturbance model: \( A_d(z^{-1}) = 1 - z^{-N} \)
Pole placement assigns the closed-loop characteristic equation:

\[
z^{-d}B_p(z^{-1})B_c(z^{-1}) + A_p(z^{-1})A_c'(z^{-1})A_d(z^{-1}) = 1 + \eta_1z^{-1} + \eta_2z^{-2} + \cdots + \eta_qz^{-q}
\]

which is in the structure of a Diophantine equation.

Design procedure: specify the desired closed-loop dynamics \( \eta(z^{-1}) \); match coefficients of \( z^{-i} \) on both sides to get \( B_c(z^{-1}) \) and \( A_c'(z^{-1}) \).

Questions:

- what are the constraints for choosing \( \eta(z^{-1}) \)?
- how to guarantee unique solution in Diophantine equation?
Design and analysis procedure

General procedure of control design:

- Problem definition
- Control design for solution (current stage)
- Prove stability
- Prove stability robustness
- Case study or implementation results

1. Pole placement: details

Theorem (Diophantine equation)

Given
\[ \eta(z^{-1}) = 1 + \eta_1 z^{-1} + \eta_2 z^{-2} + \cdots + \eta_q z^{-q} \]
\[ \alpha(z^{-1}) = 1 + \alpha_1 z^{-1} + \cdots + \alpha_n z^{-n} \]
\[ \beta(z^{-1}) = \beta_1 z^{-1} + \beta_2 z^{-2} + \cdots + \beta_n z^{-n} \]

The Diophantine equation
\[ \alpha(z^{-1}) \sigma(z^{-1}) + \beta(z^{-1}) \gamma(z^{-1}) = \eta(z^{-1}) \]

can be solved uniquely for \( \sigma(z^{-1}) \) and \( \gamma(z^{-1}) \)
\[ \sigma(z^{-1}) = 1 + \sigma_1 z^{-1} + \cdots + \sigma_{n-1} z^{-(n-1)} \]
\[ \gamma(z^{-1}) = \gamma_0 + \gamma_1 z^{-1} + \cdots + \gamma_{n-1} z^{-(n-1)} \]

if the numerators of \( \alpha(z^{-1}) \) and \( \beta(z^{-1}) \) are coprime and
\( \deg(\eta(z^{-1})) = q \leq 2n - 1 \)
1, Pole placement: details

Proof of Diophantine equation Theorem (key ideas):

\[ \alpha (z^{-1}) \sigma (z^{-1}) + \beta (z^{-1}) \gamma (z^{-1}) = \eta (z^{-1}) \]

Matching the coefficients of \( z^{-i} \) gives (see one numerical example in course reader)

\[
S = \begin{bmatrix}
\sigma_1 & \alpha_1 \\
\sigma_2 & \alpha_2 \\
\vdots & \vdots \\
\sigma_{n-1} & \alpha_n \\
\gamma_0 & 0 \\
\vdots & \vdots \\
\gamma_{n-1} & 0
\end{bmatrix} + \begin{bmatrix}
\eta_1 \\
\eta_2 \\
\vdots \\
\eta_{n-1} \\
\eta_n \\
\vdots \\
\eta_{2n-1}
\end{bmatrix}
\]

The coprime condition assures \( S \) is invertible. \( \deg \eta (z^{-1}) \leq 2n - 1 \) assures the proper dimension on the right hand side of the equality.

2, Prototype repetitive control: simple case

If all poles and zeros of the plant are stable, then prototype repetitive control uses

\[ C (z^{-1}) = \frac{k_r z^{-N+d} A_p (z^{-1})}{(1 - z^{-N}) B_p (z^{-1})} \]

Theorem (Prototype repetitive control)

Under the assumptions above, the closed-loop system is asymptotically stable for \( 0 < k_r < 2 \)
2, Prototype repetitive control: stability

Proof of Theorem on prototype repetitive control:

From
\[ 1 + \frac{k_r z^{-N+d} A_p(z^{-1})}{(1-z^{-N}) B_p(z^{-1})} \cdot \frac{z^{-d} B_p(z^{-1})}{A_p(z^{-1})} = 0 \]

the closed-loop characteristic equation is
\[ A_p(z^{-1}) B_p(z^{-1}) \left[ 1 - (1-k_r) z^{-N} \right] = 0 \]

- roots of \( A_p(z^{-1}) B_p(z^{-1}) = 0 \) are all stable
- roots of \( 1 - (1-k_r) z^{-N} = 0 \) are
\[ |1-k_r|^{\frac{1}{N}} e^{\frac{2\pi i}{N}}, \ i = 0, \pm 1, \ldots, \ \text{for} \ 0 < k_r \leq 1 \]
\[ |1-k_r|^{\frac{1}{N}} e^{i \left( \frac{2\pi i}{N} + \frac{\pi}{N} \right)}, \ i = 0, \pm 1, \ldots, \ \text{for} \ 1 < k_r \]

which are all inside the unit circle

Consider the case with plant uncertainty
\[ \forall k_r > 0, \ \text{the closed-loop system is now unstable!} \]
To make the controller robust to plant uncertainties, do instead

\[ C(z^{-1}) = \frac{k_r q(z, z^{-1}) z^{-N+d} A_p(z^{-1})}{(1 - q(z, z^{-1}) z^{-N}) B_p(z^{-1})} \]  

\( q(z, z^{-1}) \): low-pass filter. e.g. zero-phase low pass \( \frac{a_1 z^{-1} + a_0 + a_1 z}{a_0 + 2a_1} \)

which shifts the marginary stable open-loop poles to be inside the unit circle:

\[ A_p(z^{-1}) B_p(z^{-1}) \left[ 1 - (1 - k_r) q(z, z^{-1}) z^{-N} \right] = 0 \]

If poles of the plant are stable but **NOT all zeros are stable**, let

\[ B_p(z^{-1}) = B_p^{-}(z^{-1}) B_p^{+}(z^{-1}) \quad [B_p^{-}(z^{-1})—the uncancellable part] \]

and

\[ C(z^{-1}) = \frac{k_r z^{-N+\mu} A_p(z^{-1}) B_p^{-}(z) z^{-\mu}}{(1 - z^{-N}) B_p^{+}(z^{-1}) z^{-d} b} , \quad b > \max_{\omega \in [0,\pi]} |B_p^{-}(e^{i\omega})|^2 \]

Similar as before, can show that the closed-loop system is stable (in-class exercise).
2, Prototype repetitive control: extension

Exercise: analyze the stability of

\[ D(z^{-1}) = \frac{B_d(z^{-1})}{A_d(z^{-1})} \]

\[ R(z^{-1}) + E(z^{-1}) \rightarrow C(z^{-1}) \]

\[ Y(z^{-1}) = \frac{z^{-d} B_p(z^{-1})}{A_p(z^{-1})} \]

\[ C(z^{-1}) = \frac{k_r z^{-N+\mu} A_p(z^{-1}) B_p^-(z) z^{-\mu}}{(1-z^{-N}) B_p^+(z^{-1}) z^{-d} b}, \quad b > \max_{\omega \in [0,\pi]} |B_p^-(e^{j\omega})|^2 \quad (3) \]

Key steps:

\[ \left| \frac{B_p^-(e^{j\omega}) B_p^-(e^{-j\omega})}{b} \right| < 1; \quad \left| \frac{k_r B_p^-(e^{j\omega}) B_p^-(e^{-j\omega})}{b} - 1 \right| < 1; \quad \text{all roots from} \]

\[ z^{-N} \left[ \frac{k_r B_p^-(z) B_p^-(z^{-1})}{b} - 1 \right] + 1 = 0 \]

are inside the unit circle.

Robust version in the presence of plant uncertainties:

\[ C(z^{-1}) = \frac{k_r z^{-N+\mu} q(z, z^{-1}) A_p(z^{-1}) B_p^-(z) z^{-\mu}}{(1-q(z, z^{-1}) z^{-N}) B_p^+(z^{-1}) z^{-d} b} \quad (4) \]

where

\[ q(z, z^{-1}) : \text{low-pass filter. e.g. zero-phase low pass} \quad \frac{\alpha_1 z^{-1} + \alpha_0 + \alpha_1 z}{\alpha_0 + 2\alpha_1} \]

and \( \mu \) is the order of \( B_p^-(z) \)
Example

\[ r(k) \]  
\[ \downarrow \]  
\[ C(z^{-1}) \]  
\[ \downarrow \]  
\[ \text{Plant} \]  
\[ \downarrow \]  
\[ y(k) \]

Disturbance period: \( N = 10 \); nominal plant:

\[
\frac{z^{-d} B_p(z^{-1})}{A_p(z^{-1})} = \frac{z^{-1}}{(1-0.8z^{-1})(1-0.7z^{-1})}
\]

\[
C(z^{-1}) = k_r \frac{(1-0.8z^{-1})(1-0.7z^{-1})q(z,z^{-1})z^{-10}}{z^{-1}(1-q(z,z^{-1})z^{-10})}
\]

Additional reading

- ME233 course reader
1. **Big Picture**
   review of integral control in PID design

2. **Internal Model Principle**
   theorems
   typical disturbance structures

3. **Repetitive Control**
   use of internal model principle
   design by pole placement
   design by stable pole-zero cancellation
Lecture 14: Disturbance Observer

Big picture

Disturbance and uncertainties in mechanical systems:

- system models are important in design: e.g., in ZPET, observer, and preview controls
- inevitable to have uncertainty in actual mechanical systems
- system is also subjected to disturbances

Related control design:

- robust control
- adaptive control

Disturbance observer is one example of robust control.
Disturbance observer (DOB)

- introduced by Ohnishi (1987) and refined by Umeno and Hori (1991)

System:

\[ V(s) = G_{uv}(s) [U(s) + D(s)] \]

Assumptions: 
- \( u(t) \)–input; \( d(t) \)–disturbance; \( v(t) \)–output;
- \( G_{uv}(s) \)–actual plant dynamics between \( u \) and \( v \);
- \( G^n_{nv}(s) \)–nominal model

if \( Q(s) = 1 \), then

\[ U(s) = U^*(s) - \left[ \frac{G_{uv}(s)}{G^n_{uv}(s)} (U(s) + D(s)) + \frac{1}{G^n_{uv}(s)} \Xi(s) - U(s) \right] \]

\[ \Rightarrow U(s) = \frac{G^n_{uv}(s)}{G_{uv}(s)} U^*(s) - D(s) - \frac{1}{G_{uv}(s)} \Xi(s) \]

\[ V(s) = G^n_{uv}(s) U^*(s) - \Xi(s) \]

i.e., dynamics between \( U^*(s) \) and \( V(s) \) follows the nominal model; and disturbance is rejected
DOB intuition

if \( Q(s) = 1 \), then

\[
\hat{D}(s) = \left( \frac{G_{uv}(s)}{G_{uv}^n(s)} - 1 \right) U(s) + \frac{1}{G_{uv}^n(s)} \Xi(s) + \frac{G_{uv}(s)}{G_{uv}^n(s)} D(s)
\]

\[
\approx \frac{1}{G_{uv}(s)} \Xi(s) + D(s) \text{ if } G_{uv}(s) = G_{uv}^n(s)
\]

i.e., disturbance \( D(s) \) is estimated by \( \hat{D}(s) \).

DOB details: causality

It is impractical to have \( Q(s) = 1 \).

e.g., if \( G_{uv}(s) = 1/s^2 \), then \( 1/G_{uv}^n(s) = s^2 \) (not realizable)

\( Q(s) \) should be designed such that \( Q(s)/G_{nv}^n(s) \) is causal. e.g. (low-pass filter)

\[
Q(s) = \frac{1 + \sum_{k=1}^{N-r} a_k (\tau s)^k}{1 + \sum_{k=1}^{N} a_k (\tau s)^k}, \quad Q(s) = \frac{3\tau s + 1}{(\tau s + 1)^3}, \quad Q(s) = \frac{6(\tau s)^2 + 4\tau s + 1}{(\tau s + 1)^4}
\]

where \( \tau \) determines the filter bandwidth
DOB details: nominal model following

\[
\begin{align*}
V(s) &= G_{uv}^o(s) U^*(s) + G_{dv}^o(s) D(s) + G_{\xi v}^o(s) \Xi(s) \\
\end{align*}
\]

where

\[
\begin{align*}
G_{uv}^o &= \frac{G_{uv} G_{uv}^n}{G_{uv}^n + (G_{uv} - G_{uv}^n) Q}, \\
G_{dv}^o &= \frac{G_{uv} G_{uv}^n (1 - Q)}{G_{uv}^n + (G_{uv} - G_{uv}^n) Q}, \\
G_{\xi v}^o &= - \frac{G_{uv} Q}{G_{uv}^n + (G_{uv} - G_{uv}^n) Q}
\end{align*}
\]

if \( Q(s) \approx 1 \), we have disturbance rejection and nominal model following:

\[
G_{uv}^o \approx G_{uv}^n, \quad G_{dv}^o \approx 0, \quad G_{\xi v}^o = -1
\]

if \( Q(s) \approx 0 \), DOB is cut off:

\[
G_{uv}^o \approx G_{uv}, \quad G_{dv}^o \approx G_{uv}, \quad G_{\xi v}^o \approx 0
\]
DOB details: stability robustness

\[
\begin{align*}
D(s) & \quad U^*(s) \quad U(s) \quad G_{uv}(s) \quad V(s) \\
\hat{D}(s) & \quad - \quad Q(s) \quad 1/G^n_{uv}(s) \quad + \quad \Xi(s) \\
\end{align*}
\]

\[
G_{uv} = \frac{G_{uv} G'_{uv}}{G_{uv} + (G_{uv} - G'_{uv}) Q}, \quad G_{dv} = \frac{G_{uv} G^n_{uv} (1 - Q)}{G_{uv} + (G_{uv} - G^n_{uv}) Q}, \quad G_{\xi_v} = -\frac{G_{uv} Q}{G_{uv} + (G_{uv} - G^n_{uv}) Q}
\]

closed-loop characteristic equation:

\[
G^n_{uv}(s) + (G_{uv}(s) - G^n_{uv}(s)) Q(s) = 0
\]

\[
\Leftrightarrow G^n_{uv}(s) (1 + \Delta(s) Q(s)) = 0, \text{ if } G_{uv}(s) = G^n_{uv}(s) (1 + \Delta(s))
\]

**robust stability condition**: stable zeros for \(G^n_{nv}(s)\), plus

\[
||\Delta(j\omega) Q(j\omega)|| < 1, \quad \forall \omega
\]

Application example
Discrete-time case

\[ u^*(k) \uparrow \quad u(k) + \quad d(k) \downarrow \quad P(z^{-1}) \rightarrow \rightarrow \quad y(k) \]

\[ \begin{align*}
&\downarrow \quad \downarrow \\
&z^{-m} \quad + \quad 1/P_n(z^{-1}) \\
&Q(z^{-1})
\end{align*} \]

where \( P(z^{-1}) \approx z^{-m}P_n(z^{-1}) \)

Big picture

We have been assuming knowledge of the plant in controller design. In practice, plant models come from:

- modeling by physics: Newton’s law, conservation of energy, etc
- (input-output) data-based system identification

The need for system identification and adaptive control come from

- unknown plants
- time-varying plants
- known disturbance structure but unknown disturbance parameters
System modeling

Consider the input-output relationship of a plant:

\[ u(k) \rightarrow G_p(z^{-1}) = \frac{z^{-1}B(z^{-1})}{A(z^{-1})} \rightarrow y(k) \]

or equivalently

\[ u(k) \rightarrow \frac{B(z^{-1})}{A(z^{-1})} \rightarrow y(k + 1) \quad (1) \]

where

\[ B(z^{-1}) = b_0 + b_1z^{-1} + \cdots + b_mz^{-m}; \quad A(z^{-1}) = 1 + a_1z^{-1} + \cdots + a_nz^{-n} \]

- \( y(k + 1) \) is a linear combination of \( y(k), \ldots, y(k + 1 - n) \) and \( u(k), \ldots, u(k - m) \):

\[ y(k + 1) = -\sum_{i=1}^{n} a_i y(k + 1 - i) + \sum_{i=0}^{m} b_i u(k - i) \quad (2) \]

Define parameter vector \( \theta \) and regressor vector \( \phi(k) \):

\[ \theta \triangleq [a_1, a_2, \cdots a_n, b_0, b_1, \cdots, b_m]^T \]

\[ \phi(k) \triangleq [-y(k), \cdots, -y(k + 1 - n), u(k), u(k - 1), \cdots, u(k - m)]^T \]

- (2) can be simply written as:

\[ y(k + 1) = \theta^T \phi(k) \quad (3) \]

- \( \phi(k) \) and \( y(k + 1) \) are known or measured
- **goal**: estimate the unknown \( \theta \)
Parameter estimation

Suppose we have an estimate of the parameter vector:
\[ \hat{\theta} \triangleq [\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_n, \hat{b}_0, \hat{b}_1, \ldots, \hat{b}_m]^T \]

At time \( k \), we can do estimation:
\[ \hat{y}(k+1) = \hat{\theta}^T(k) \phi(k) \]  
(4)

where \( \hat{\theta}(k) \triangleq [\hat{a}_1(k), \hat{a}_2(k), \ldots, \hat{a}_n(k), \hat{b}_0(k), \hat{b}_1(k), \ldots, \hat{b}_m(k)]^T \)

Parameter identification by least squares (LS)

At time \( k \), the least squares (LS) estimate of \( \theta \) minimizes:
\[ J_k = \sum_{i=1}^{k} \left[ y(i) - \hat{\theta}^T(k) \phi(i-1) \right]^2 \]  
(5)

Solution:
\[ J_k = \sum_{i=1}^{k} \left( y(i)^2 + \hat{\theta}^T(k) \phi(i-1) \phi^T(i-1) \hat{\theta}(k) - 2y(i) \phi^T(i-1) \hat{\theta}(k) \right) \]

Letting \( \partial J_k / \partial \hat{\theta}(k) = 0 \) yields
\[ \hat{\theta}(k) = \left[ \sum_{i=1}^{k} \phi(i-1) \phi^T(i-1) \right]^{-1} \sum_{i=1}^{k} \phi(i-1) y(i) \]  
(6)
Recursive least squares (RLS)

At time $k + 1$, we know $u(k + 1)$ and have one more measurement $y(k + 1)$. Instead of (5), we can do better by minimizing

$$J_{k+1} = \sum_{i=1}^{k+1} [y(i) - \hat{\theta}^T(k+1)\phi(i-1)]^2$$

whose solution is

$$\hat{\theta}(k+1) = \left[\sum_{i=1}^{k+1} \phi(i-1)\phi^T(i-1)\right]^{-1} \sum_{i=1}^{k+1} \phi(i-1)y(i) \quad (7)$$

**recursive least squares** (RLS): no need to do the matrix inversion in (7), $\hat{\theta}(k+1)$ can be obtained by

$$\hat{\theta}(k+1) = \hat{\theta}(k) + \text{[correction term]} \quad (8)$$

**RLS parameter adaptation**

Goal: to obtain

$$\hat{\theta}(k + 1) = \hat{\theta}(k) + \text{[correction term]} \quad (9)$$

Derivations:

$$F(k+1)^{-1} = \sum_{i=1}^{k+1} \phi(i-1)\phi^T(i-1) = F(k)^{-1} + \phi(k)\phi^T(k)$$

$$\hat{\theta}(k + 1) = F(k+1) \sum_{i=1}^{k+1} \phi(i-1)y(i)$$

$$= F(k+1) \left[ \sum_{i=1}^{k} \phi(i-1)y(i) + \phi(k)y(k+1) \right]$$

$$= F(k+1) \left[ F(k)^{-1} \hat{\theta}(k) + \phi(k)y(k+1) \right]$$

$$= F(k+1) \left[ \left( F(k+1)^{-1} - \phi(k)\phi^T(k) \right) \hat{\theta}(k) + \phi(k)y(k+1) \right]$$

$$= \hat{\theta}(k) + F(k+1)\phi(k) \left[ y(k+1) - \hat{\theta}^T(k)\phi(k) \right] \quad (10)$$
RLS parameter adaptation

Define

\[ \hat{y}^o(k + 1) = \hat{\theta}^T(k)\phi(k) \]
\[ \varepsilon^o(k + 1) = y(k + 1) - \hat{y}^o(k + 1) \]

(10) is equivalent to

\[ \hat{\theta}(k + 1) = \hat{\theta}(k) + F(k + 1)\phi(k)\varepsilon^o(k + 1) \]  

(11)

RLS adaptation gain recursion

\( F(k + 1) \) is called the adaptation gain, and can be updated by

\[ F(k + 1) = F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{1 + \phi^T(k)F(k)\phi(k)} \]  

(12)

Proof:

- matrix inversion lemma: if \( A \) is nonsingular, \( B \) and \( C \) have compatible dimensions, then
  \[ (A + BC)^{-1} = A^{-1} - A^{-1}B(CA^{-1}B + I)^{-1}CA^{-1} \]
- note the algebra
  \[ F(k + 1) = \left[ \sum_{i=1}^{k+1} \phi(i - 1)\phi^T(i - 1) \right]^{-1} = [F(k)^{-1} + \phi(k)\phi^T(k)]^{-1} \]
  \[ = F(k) - F(k)\phi(k)\left(\phi^T(k)F(k)\phi(k) + 1\right)^{-1}\phi^T(k)F(k) \]
  which gives (12)
RLS parameter adaptation

An alternative representation of adaptation law (11):

\[
(12) \Rightarrow F(k+1)\phi(k) = F(k)\phi(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{1 + \phi^T(k)F(k)\phi(k)} \phi(k)
\]

\[
= \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}
\]

Hence we have the parameter adaptation algorithm (PAA):

\[
\hat{\theta}(k+1) = \hat{\theta}(k) + F(k+1)\phi(k)\epsilon^o(k+1)
\]

\[
= \hat{\theta}(k) + \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}\epsilon^o(k+1)
\]

\[
F(k+1) = F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{1 + \phi^T(k)F(k)\phi(k)}
\]

PAA implementation

- \( \hat{\theta} (0) \): initial guess of parameter vector

\[
\hat{\theta}(k+1) = \hat{\theta}(k) + \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}\epsilon^o(k+1)
\]

- \( F (0) = \sigma I \): \( \sigma \) is a large number, as \( F (k) \) is always none-increasing

\[
F(k+1) = F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{1 + \phi^T(k)F(k)\phi(k)}
\]
RLS parameter adaptation

Up till now we have been using the *a priori* prediction and *a priori* prediction error:

\[
\hat{y}^o(k + 1) = \hat{\theta}^T(k)\phi(k) \quad \text{after measurement of } y(k)
\]
\[
\varepsilon^o(k + 1) = y(k + 1) - \hat{y}^o(k + 1)
\]

Further analysis (e.g., convergence of \(\hat{\theta}(k)\)) requires the new definitions of *a posteriori* prediction and *a posteriori* prediction error:

\[
\hat{y}(k + 1) = \hat{\theta}^T(k + 1)\phi(k) \quad \text{after measurement of } y(k + 1)
\]
\[
\varepsilon(k + 1) = y(k + 1) - \hat{y}(k + 1)
\]

**Relationship between \(\varepsilon(k + 1)\) and \(\varepsilon^o(k + 1)\)**

Note that

\[
\hat{\theta}(k + 1) = \hat{\theta}(k) + \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}\varepsilon^o(k + 1)
\]

\[
\Rightarrow \phi^T(k)\hat{\theta}(k + 1) = \phi^T(k)\hat{\theta}(k) + \frac{\phi^T(k)F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}\varepsilon^o(k + 1)
\]

\[
\Rightarrow y(k + 1) - \hat{y}(k + 1) = y(k + 1) - \hat{y}^o(k + 1) - \frac{\phi^T(k)F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}\varepsilon^o(k + 1)
\]

Hence

\[
\varepsilon(k + 1) = \frac{\varepsilon^o(k + 1)}{1 + \phi^T(k)F(k)\phi(k)}
\]

▶ note: \(|\varepsilon(k + 1)| \leq |\varepsilon^o(k + 1)| \) \((\hat{y}(k + 1)\) is more accurate than \(\hat{y}^o(k + 1)\))
A posteriori RLS parameter adaptation

With (13), we can write the PAA in the a posteriori form

\[
\hat{\theta}(k + 1) = \hat{\theta}(k) + F(k)\phi(k)\varepsilon(k + 1)
\]  

(14)

which is not implementable but is needed for stability analysis.

Forgetting factor

motivation

- previous discussions assume the actual parameter vector \( \theta \) is constant
- adaptation gain \( F(k) \) keeps decreasing, as

\[
F^{-1}(k + 1) = F^{-1}(k) + \phi(k)\phi^T(k)
\]

- this means adaptation becomes weaker and weaker
- for time-varying parameters, we need a mechanism to “forget” the “old’ data
Forgetting factor

Consider a new cost

\[ J_k = \sum_{i=1}^{k} \lambda^{k-i} \left[ y(i) - \hat{\theta}^T(k)\phi(i-1) \right]^2, \quad 0 < \lambda \leq 1 \]

▷ past errors are less weighted:

\[ J_k = \left[ y(k) - \hat{\theta}^T(k)\phi(k-1) \right]^2 + \lambda \left[ y(k-1) - \hat{\theta}^T(k)\phi(k-2) \right]^2 \]
\[ \quad + \lambda^2 \left[ y(k-2) - \hat{\theta}^T(k)\phi(k-3) \right]^2 + \ldots \]

▷ the solution is:

\[ \hat{\theta}(k) = \left[ \sum_{i=1}^{k} \lambda^{k-i} \phi(i-1)\phi^T(i-1) \right]^{-1} \sum_{i=1}^{k} \lambda^{k-i} \phi(i-1)y(i) \quad (15) \]

Forgetting factor

▷ in (15), the recursion of the adaptation gain is:

\[ F(k+1)^{-1} = \lambda F(k)^{-1} + \phi(k)\phi(k)^T \]

or, equivalently

\[ F(k+1) = \frac{1}{\lambda} \left[ F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda + \phi^T(k)F(k)\phi(k)} \right] \quad (16) \]
Forgetting factor

The weighting can be made more flexible:

\[
F(k + 1) = \frac{1}{\lambda_1(k)} \left[ F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda_1(k) + \phi^T(k)F(k)\phi(k)} \right]
\]

which corresponds to the cost function

\[
J_k = \left[ y(k) - \hat{\theta}^T(k)\phi(k-1) \right]^2 + \lambda_1(k-1)\left[ y(k-1) - \hat{\theta}^T(k)\phi(k-2) \right]^2 + \lambda_1(k-1)\lambda_1(k-2)\left[ y(k-2) - \hat{\theta}^T(k)\phi(k-3) \right]^2 + \ldots
\]

i.e. (assuming \( \prod_{j=k}^{k-1} \lambda_1(j) = 1 \))

\[
J_k = \sum_{i=1}^{k} \left\{ \left( \prod_{j=i}^{k-1} \lambda_1(j) \right) \left[ y(i) - \hat{\theta}^T(k)\phi(i-1) \right]^2 \right\}
\]

The general form of the adaptation gain is:

\[
F(k + 1) = \frac{1}{\lambda_1(k)} \left[ F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda_1(k) + \phi^T(k)F(k)\phi(k)} \right]
\] (17)

which comes from:

\[
F(k + 1)^{-1} = \lambda_1(k)F(k)^{-1} + \lambda_2(k)\phi(k)\phi^T(k)
\]

with \( 0 < \lambda_1(k) \leq 1 \) and \( 0 \leq \lambda_2(k) \leq 2 \) (for stability requirements, will come back to this soon).

<table>
<thead>
<tr>
<th>( \lambda_1(k) )</th>
<th>( \lambda_2(k) )</th>
<th>PAA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>constant adaptation gain</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>least square gain</td>
</tr>
<tr>
<td>(&lt; 1)</td>
<td>1</td>
<td>least square gain with forgetting factor</td>
</tr>
</tbody>
</table>
*Influence of the initial conditions*

If we initialize $F(k)$ and $\hat{\theta}(k)$ at $F_0$ and $\theta_0$, we are actually minimizing

$$J_k = (\hat{\theta}(k) - \theta_0)^T F_0^{-1} (\hat{\theta}(k) - \theta_0) + \sum_{i=1}^{k} \alpha_i \left[ y(i) - \hat{\theta}^T(k) \phi(i-1) \right]^2$$

where $\alpha_i$ is the weighting for the error at time $i$. The least square parameter estimate is

$$\hat{\theta}(k) = \left[ F_0^{-1} + \sum_{i=1}^{k} \alpha_i \phi(i-1) \phi^T(i-1) \right]^{-1} \left[ F_0^{-1} \theta_0 + \sum_{i=1}^{k} \alpha_i \phi(i-1) y(i) \right]$$

We see the relative importance of the initial values decays with time.

If it is possible to wait a few samples before the adaptation, proper initial values can be obtained if the recursion is started at time $k_0$ with

$$F(k_0) = \left[ \sum_{i=1}^{k_0} \alpha_i \phi(i-1) \phi^T(i-1) \right]^{-1}$$

$$\hat{\theta}(k_0) = F(k_0) \sum_{i=1}^{k_0} \alpha_i \phi(i-1) y(i)$$
Lecture 16: Stability of Parameter Adaptation Algorithms

Big picture

▶ For
\[ \hat{\theta}(k+1) = \hat{\theta}(k) + \text{[correction term]} \]
we haven't talked about whether \( \hat{\theta}(k) \) will converge to the true value \( \theta \) if \( k \to \infty \). We haven't even talked about whether \( \hat{\theta}(k) \) will stay bounded or not!

▶ Tools of stability evaluation: Lyapunov-based analysis, or hyperstability theory (topic of this lecture)
Outline

1. Big picture

2. Hyperstability theory
   - Passivity
   - Main results
   - Positive real and strictly positive real
   - Understanding the hyperstability theorem

3. Procedure of PAA stability analysis by hyperstability theory

4. Appendix
   - Strictly positive real implies strict passivity

Hyperstability theory

history

Vasile M. Popov:
- born in 1928, Romania
- retired from University of Florida in 1993
- developed hyperstability theory independently from Lyapnov theory
Hyperstability theory
Consider a closed-loop system in Fig. 1

\[ \begin{align*}
\dot{x}(t) &= Ax(t) + Bu(t) \\
v(t) &= Cx(t) + Du(t)
\end{align*} \]

Hyperstability discusses conditions for “nice” behaviors in \( x \).

Passive systems

**Definition (Passive system).**
The system \( v \rightarrow \text{System} \rightarrow w \) is called passive if

\[ \int_0^{t_1} w^T(t)v(t)\,dt \geq -\gamma^2, \quad \forall t_1 \geq 0 \quad \text{or} \quad \sum_{k=0}^{k_1} w^T(k)v(k) \geq -\gamma^2, \quad \forall k_1 \geq 0 \]

where \( \delta \) and \( \gamma \) depends on the initial conditions.

- intuition: \( \int_0^{t_1} w^T(t)v(t)\,dt \) is the work/supply done to the system. By conservation of energy,

\[ E(t_1) \leq E(0) + \int_0^{t_1} w^T(t)v(t)\,dt \]
Strictly passive systems

If the equality is *strict* in the passivity definition, with

\[
\int_0^{t_1} w^T(t) v(t) \, dt \geq -\gamma^2
\]

\[
+ \delta \int_0^{t_1} v^T(t) v(t) \, dt + \varepsilon \int_0^{t_1} w^T(t) w(t) \, dt, \quad \forall t_1 \geq 0
\]

or in the discrete-time case

\[
\sum_{k=0}^{k_1} w^T(k) v(k) \geq -\gamma^2
\]

\[
+ \delta \sum_{k=0}^{k_1} v^T(k) v(k) + \varepsilon \sum_{k=0}^{k_1} w^T(k) w(k), \quad \forall k_1 \geq 0
\]

where \( \delta \geq 0, \ \varepsilon \geq 0 \), but not both zero, the system is *strictly passive*.

Passivity of combined systems

**Fact (Passivity of connected systems).**

*If two systems \( S_1 \) and \( S_2 \) are both passive, then the following parallel and feedback combination of \( S_1 \) and \( S_2 \) are also passive.*
Definition (Hyperstability).
The feedback system in Fig. 1 is hyperstable if and only if there exist positive constants $\delta > 0$ and $\gamma > 0$ such that
\[
\|x(t)\| < \delta [\|x(0)\| + \gamma], \quad \forall t > 0 \quad \text{or} \quad \|x(k)\| < \delta [\|x(0)\| + \gamma], \quad \forall k > 0
\]
for all feedback blocks that satisfy the Popov inequality
\[
\int_{0}^{t_1} w^T(t) v(t) dt \geq -\gamma^2, \quad \forall t_1 \geq 0 \quad \text{or} \quad \sum_{k=0}^{k_1} w^T(k) v(k) \geq -\gamma^2, \quad \forall k_1 \geq 0
\]
In other words, the LTI block is bounded in states for any initial conditions for any passive nonlinear blocks.

Definition (Asymptotic hyperstability).
The feedback system below is asymptotically hyperstable if and only if it is hyperstable and for all bounded $w$ satisfying the Popov inequality we have
\[
\lim_{k \to \infty} x(k) = 0
\]
Hyperstability theory

Theorem (Hyperstability).

The feedback system in Fig. 1 is hyperstable if and only if the nonlinear block satisfies Popov inequality (i.e., it is passive) and the LTI transfer function is positive real.

Theorem (Asymptotical hyperstability).

The feedback system in Fig. 1 is asymptotically hyperstable if and only if the nonlinear block satisfies Popov inequality and the LTI transfer function is strictly positive real.

intuition: a strictly passive system in feedback connection with a passive system gives an asymptotically stable closed loop.

Positive real and strictly positive real

Positive real transfer function (continuous-time case): a SISO transfer function $G(s)$ is called positive real (PR) if

- $G(s)$ is real for real values of $s$
- $\text{Re}\{G(s)\} > 0$ for $\text{Re}\{s\} > 0$

The above is intuitive but not practical to evaluate. Equivalently, $G(s)$ is PR if

1. $G(s)$ does not possess any pole in $\text{Re}\{s\} > 0$ (no unstable poles)
2. any pole on the imaginary axis $j\omega_0$ does not repeat and the associated residue (i.e., the coefficient appearing in the partial fraction expansion) $\lim_{s \to j\omega_0} (s - j\omega_0) G(s)$ is non-negative
3. $\forall \omega \in \mathbb{R}$ where $s = j\omega$ is not a pole of $G(s)$, $G(j\omega) + G(-j\omega) = 2\text{Re}\{G(j\omega)\} \geq 0$
Positive real and strictly positive real

Strictly positive real transfer function (continuous-time case): a SISO transfer function $G(s)$ is strictly positive real (SPR) if

1. $G(s)$ does not possess any pole in $\text{Re}\{s\} \geq 0$

2. $\forall \omega \in \mathbb{R}, \ G(j\omega) + G(-j\omega) = 2\text{Re}\{G(j\omega)\} > 0$

![Nyquist plot example](image.png)

**Figure:** example Nyquist plot of a SPR transfer function

Positive real and strictly positive real

discrete-time case

A SISO discrete-time transfer function $G(z)$ is positive real (PR) if:

1. it does not possess any pole outside of the unit circle

2. any pole on the unit circle does not repeat and the associated residue is non-negative

3. $\forall |\omega| \leq \pi$ where $z = e^{j\omega}$ is not a pole of $G(z)$,

\[ G(e^{-j\omega}) + G(e^{j\omega}) = 2\text{Re}\{G(e^{j\omega})\} \geq 0 \]

$G(z)$ is strictly positive real (SPR) if:

1. $G(z)$ does not possess any pole outside of or on the unit circle on $z$-plane

2. $\forall |\omega| < \pi, \ G(e^{-j\omega}) + G(e^{j\omega}) = 2\text{Re}\{G(e^{j\omega})\} > 0$
Examples of PR and SPR transfer functions

- \( G(z) = c \) is SPR if \( c > 0 \)
- \( G(z) = \frac{1}{z-a}, \ |a| < 1 \) is asymptotically stable but not PR:

\[
2 \Re \{ \mathcal{F}(e^{j\omega}) \} = \frac{1}{e^{j\omega} - a} + \frac{1}{e^{-j\omega} - a} \frac{\cos \omega - a}{1 + a^2 - 2a \cos \omega}
\]

- \( G(z) = \frac{z}{z-a}, \ |a| < 1 \) is asymptotically stable and SPR

Strictly positive real implies strict passivity

It turns out [see Appendix (to prove on board at the end of class)]:

**Lemma**: the LTI system \( G(s) = C(sI - A)^{-1}B + D \) (in minimal realization)

\[
\dot{x} = Ax + Bu \\
y = Cx + Du
\]

is

- passive if \( G(s) \) is positive real
- strictly passive if \( G(s) \) is strictly positive real

Analogous results hold for discrete-time systems.
Understanding the hyperstability theorem

Example: consider a mass-spring-damper system

\[ m\ddot{x} + b\dot{x} + kx = u \Rightarrow \]

\[ G_{u \rightarrow x}(s) = \frac{1}{ms^2 + bs + k} \]

\[ G_{u \rightarrow v}(s) = \frac{s}{ms^2 + bs + k} \]

with a general nonlinear feedback control law

\[ \int_0^{t_1} u(t) v(t) \, dt \text{ is the total energy supplied to the system} \]
Understanding the hyperstability theorem

- if the nonlinear block satisfies the Popov inequality
  \[ \int_{t_1}^{t_0} w(t) v(t) \, dt \geq -\gamma_0^2, \quad \forall t_1 \geq 0 \]

  then from \( u(t) = -w(t) \), the energy supplied to the system is bounded by
  \[ \int_{t_1}^{t_0} u(t) v(t) \, dt \leq \gamma_0^2 \]

- energy conservation (assuming \( v(0) = v_0 \) and \( x(0) = x_0 \)):
  \[ \frac{1}{2} m v^2 + \frac{1}{2} k x^2 - \frac{1}{2} m v_0^2 - \frac{1}{2} k x_0^2 = \int_{t_1}^{t_0} u(t) v(t) \, dt \leq \gamma_0^2 \]

- define state vector \( x = [x_1, x_2]^T \), \( x_1 = \sqrt{\frac{k}{2}} x \), \( x_2 = \sqrt{\frac{m}{2}} v \), then
  \[ \|x(t)\|^2 \leq \|x(0)\|^2 + \gamma_0^2 \leq (\|x(0)\|^2 + \gamma_0)^2 \]

which is a special case in the hyperstability definition

intuition from the example:
The nonlinear block satisfying Popov inequality assures bounded supply to the LTI system. Based on energy conservation, the energy of the LTI system is bounded. If the energy function is positive definite with respect to the states, then the states will be bounded.

more intuition:
If the LTI system is strictly passive, it consumes energy. The bounded supply will eventually be all consumed, hence the convergence to zero for the states.
A remark about hyperstability

An example of a system that is asymptotically hyperstable and stable:

\[ v \xrightarrow{+} \frac{1}{s+1} \xrightarrow{\text{static gain } k (>0)} w \]

Stable systems may however be not hyperstable: for instance

\[ v \xrightarrow{+} \frac{1}{s-1} \xrightarrow{\text{static gain } k (>1)} w \]

is stable but not hyperstable \((\frac{1}{s-1} \text{ is unstable and hence not SPR})\)

Outline

1. Big picture

2. Hyperstability theory
   - Passivity
   - Main results
   - Positive real and strictly positive real
   - Understanding the hyperstability theorem

3. Procedure of PAA stability analysis by hyperstability theory

4. Appendix
   - Strictly positive real implies strict passivity
PAA stability analysis by hyperstability theory

- step 1: translate the adaptation algorithm to a feedback combination of a LTI block and a nonlinear block, as shown in Fig. 1
- step 2: verify that the feedback block satisfies the Popov inequality
- step 3: check that the LTI block is strictly positive real
- step 4: show that the output of the feedback block is bounded. Then from the definition of asymptotic hyperstability, we conclude that the state $x$ converges to zero

Example: hyperstability of RLS with constant adaptation gain

Recall PAA with recursive least squares:

- a priori parameter update

$$
\hat{\theta}(k+1) = \hat{\theta}(k) + \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}\varepsilon^o(k+1)
$$

- a posteriori parameter update

$$
\hat{\theta}(k+1) = \hat{\theta}(k) + F(k)\phi(k)\varepsilon(k+1)
$$

We use the a posteriori form to prove that the RLS with $F(k) = F \succ 0$ is always asymptotically hyperstable.
Example cont’d

step 1: transformation to a feedback structure

\[
\hat{\theta}(k+1) = \hat{\theta}(k) + F\phi(k)\epsilon(k+1)
\]

*parameter estimation error (vector)* \(\tilde{\theta}(k) = \hat{\theta}(k) - \theta:

\[
\tilde{\theta}(k+1) = \tilde{\theta}(k) + F\phi(k)\epsilon(k+1)
\]

*a posteriori prediction error* \(\epsilon(k+1) = y(k+1) - \hat{\theta}^T(k+1)\phi(k):

\[
\epsilon(k+1) = \theta^T\phi(k) - \hat{\theta}^T(k+1)\phi(k)
= -\tilde{\theta}^T(k+1)\phi(k)
\]
Example cont’d
step 2: Popov inequality

for the feedback nonlinear block, need to prove

\[
\sum_{k=0}^{k_1} \tilde{\theta}^T(k+1) \phi(k) \varepsilon(k+1) \geq -\gamma_0^2, \ \forall k_1 \geq 0
\]

\[
\tilde{\theta}(k+1) - \tilde{\theta}(k) = F\phi(k)\varepsilon(k+1)\text{ gives}
\]

\[
\sum_{k=0}^{k_1} \tilde{\theta}^T(k+1) \phi(k) \varepsilon(k+1) = \sum_{k=0}^{k_1} \left( \tilde{\theta}^T(k+1) F^{-1} \tilde{\theta}(k+1) - \tilde{\theta}^T(k+1) F^{-1} \tilde{\theta}(k) \right)
\]

Example cont’d
step 2: Popov inequality

“adding and substracting terms” gives

\[
\sum_{k=0}^{k_1} \tilde{\theta}^T(k+1) \phi(k) \varepsilon(k+1)
\]

\[
= \sum_{k=0}^{k_1} \left( \tilde{\theta}^T(k+1) F^{-1} \tilde{\theta}(k+1) - \tilde{\theta}^T(k+1) F^{-1} \tilde{\theta}(k) \right)
\]

\[
= \sum_{k=0}^{k_1} \left( \tilde{\theta}^T(k+1) F^{-1} \tilde{\theta}(k+1) \pm \tilde{\theta}^T(k) F^{-1} \tilde{\theta}(k) - \tilde{\theta}^T(k+1) F^{-1} \tilde{\theta}(k) \right)
\]
Example cont’d
step 2: Popov inequality
Combining terms yields
\[
\sum_{k=0}^{k_1} \left( \tilde{\theta}^T (k + 1) F^{-1} \tilde{\theta} (k + 1) \pm \tilde{\theta}^T (k) F^{-1} \tilde{\theta} (k) - \tilde{\theta}^T (k + 1) F^{-1} \tilde{\theta} (k) \right)
\]
\[
= \sum_{k=0}^{k_1} \frac{1}{2} \left( \tilde{\theta}^T (k + 1) F^{-1} \tilde{\theta} (k + 1) - \tilde{\theta}^T (k) F^{-1} \tilde{\theta} (k) \right)
\]
\[
+ \sum_{k=0}^{k_1} \frac{1}{2} \left( \tilde{\theta}^T (k + 1) F^{-1} \tilde{\theta} (k + 1) - 2 \tilde{\theta}^T (k + 1) F^{-1} \tilde{\theta} (k) + \tilde{\theta}^T (k) F^{-1} \tilde{\theta} (k) \right)
\]
\[
= \sum_{k=0}^{k_1} \frac{1}{2} \left( \tilde{\theta}^T (k + 1) F^{-1} \tilde{\theta} (k + 1) - \tilde{\theta}^T (k) F^{-1} \tilde{\theta} (k) \right)
\]
\[
\geq 0
\]

▶ [★] is equivalent to
\[
\left( F^{-1/2} \tilde{\theta} (k + 1) - F^{-1/2} \tilde{\theta} (k) \right)^T \left( F^{-1/2} \tilde{\theta} (k + 1) - F^{-1/2} \tilde{\theta} (k) \right) \geq 0
\]

Example cont’d
step 2: Popov inequality

▶ the underlined term is also lower bounded:
\[
\sum_{k=0}^{k_1} \frac{1}{2} \left( \tilde{\theta}^T (k + 1) F^{-1} \tilde{\theta} (k + 1) - \tilde{\theta}^T (k) F^{-1} \tilde{\theta} (k) \right)
\]
\[
= \frac{1}{2} \tilde{\theta}^T (k_1 + 1) F^{-1} \tilde{\theta} (k_1 + 1) - \frac{1}{2} \tilde{\theta}^T (0) F^{-1} \tilde{\theta} (0)
\]
\[
\geq - \frac{1}{2} \tilde{\theta}^T (0) F^{-1} \tilde{\theta} (0)
\]
hence
\[
\sum_{k=0}^{k_1} \tilde{\theta}^T (k + 1) \phi (k) \varepsilon (k + 1) \geq - \frac{1}{2} \tilde{\theta}^T (0) F^{-1} \tilde{\theta} (0)
\]
Example cont’d

step 3: SPR condition

\[
0 \xrightarrow{+} \text{Nonlinear Block} \xrightarrow{w(k)} 1 \xrightarrow{\epsilon(k+1)}
\]

the identity block \( G(z^{-1}) = 1 \) is always SPR

- from steps 1-3, we conclude the adaptation system is asymptotically hyperstable
- this means \( \epsilon(k+1) \) will be bounded, and if \( w(k) \) is further shown to be bounded, \( \epsilon(k+1) \) converge to zero

Example cont’d

step 4: boundedness of the signal

\[
0 \xrightarrow{+} \text{Nonlinear Block} \xrightarrow{w(k)} 1 \xrightarrow{\epsilon(k+1)}
\]

\( \epsilon(k+1) = -w(k) \), so \( w(k) \) is bounded if \( \epsilon(k+1) \) is bounded

thus hyperstability theorem gives that \( \epsilon(k+1) \) converges to zero
For this simple case, we can intuitively see why $\varepsilon(k+1) \to 0$: Popov inequality gives

$$\sum_{k=0}^{k_1} \varepsilon(k+1) w(k) \geq -\gamma_0^2; \quad \text{as} \ w(k) = -\varepsilon(k+1), \text{so}$$

$$\sum_{k=0}^{k_1} \varepsilon^2(k+1) \leq \gamma_0^2$$

Let $k_1 \to \infty$. $\varepsilon(k+1)$ must converge to 0 to ensure the boundedness.

Recall

$$\varepsilon(k+1) = \frac{\varepsilon^o(k+1)}{1 + \phi^T(k) F \phi(k)}$$

- $\varepsilon(k+1) \to 0$ does not necessarily mean $\varepsilon^o(k+1) \to 0$
- need to show $\phi(k)$ is bounded: for instance, the plant needs to be input-output stable for $y(k)$ to be bounded
- see details in ME 233 course reader

There are different PAAs with different stability and convergence requirements
Summary

1. Big picture

2. Hyperstability theory
   - Passivity
   - Main results
     - Positive real and strictly positive real
     - Understanding the hyperstability theorem

3. Procedure of PAA stability analysis by hyperstability theory

4. Appendix
   - Strictly positive real implies strict passivity

Exercise

In the following block diagrams, \( u \) and \( y \) are respectively the input and output of the overall system; \( h(\cdot) \) is a sector nonlinearity satisfying

\[
2|x| < |h(x)| < 5|x|
\]

Check whether they satisfy the Popov inequality.
Kalman Yakubovich Popov (KYP) Lemma connects frequency-domain SPR conditions and time-domain system matrices: 

**Lemma**: Consider \( G(s) = C (sI - A)^{-1} B + D \) where \((A, B)\) is controllable and \((A, C)\) is observable. \( G(s) \) is **strictly positive real** if and only if there exist matrices \( P = P^T > 0 \), \( L \), and \( W \), and a positive constant \( \varepsilon \) such that:

\[
PA + A^T P = -L^T L - \varepsilon P
\]
\[
PB = C^T - L^T W
\]
\[
W^T W = D + D^T
\]

Proof: see H. Khalil, “Nonlinear Systems”, Prentice Hall

Discrete-time version of KYP lemma: replace \( s \) with \( z \) and replace the matrix equalities with

\[
A^T PA - P = -L^T L - \varepsilon P
\]
\[
B^T PA - C = -K^T L
\]
\[
D + D^T - B^T PB = K^T K
\]
*Strictly positive real implies strict passivity

From KYP lemma, the following result can be shown:

**Lemma:** the LTI system $G(s) = C(sI - A)^{-1}B + D$ (in minimal realization)

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

is

- passive if $G(s)$ is positive real
- strictly passive if $G(s)$ is strictly positive real

Analogous results hold for discrete-time systems.

---

**Proof:** Consider $V = \frac{1}{2}x^TPx$:

$$V(x(T)) - V(x(0)) = \int_0^T \dot{V} dt = \int_0^T \left[ \frac{1}{2}x^T(A^TP + PA)x + u^TB^TPx \right] dt$$

Let $u$ and $y$ be the input and the output of $G(s)$. KYP lemma gives

$$V(x(T)) - V(x(0)) = \int_0^T \left[ -\frac{1}{2}x^T(L^TL + \varepsilon P)x + u^TB^TPx \right] dt$$

$$\int_0^T u^T y dt = \int_0^T u^T (Cx + Du) dt = \int_0^T \left[ u^T(B^TP + W^TL)x + u^TDu \right] dt$$

$$= \int_0^T \left[ u^T(B^TP + W^TL)x + \frac{1}{2}u^T(D + D^T)u \right] dt$$

$$= \int_0^T \left[ u^T(B^TP + W^TL)x + \frac{1}{2}u^TW^TWu \right] dt$$
**Strictly positive real implies strict passivity**

hence

\[ \int_0^T u^T y dt - V(x(T)) + V(x(0)) \]

\[ = \int_0^T \left[ u^T (B^T P + W^T L)x + \frac{1}{2} u^T W W^T u + \frac{1}{2} x^T (L^T L + \varepsilon P)x - u^T B^T P x \right] dt \]

\[ = \frac{1}{2} \int_0^T (Lx + Wu)^T (Lx + Wu) dt + \frac{1}{2} \varepsilon x^T P x \geq \frac{1}{2} \varepsilon x^T P x > 0 \]
Lecture 17: PAA with Parallel Predictors

Big picture: we know now...

\[
\begin{align*}
  u(k) & \rightarrow \frac{B(z^{-1})}{A(z^{-1})} \rightarrow y(k+1) \\
  y(k+1) & = B(z^{-1}) u(k) - (A(z^{-1}) - 1) y(k+1) \\
  & = \theta^T \phi(k)
\end{align*}
\]

In RLS:

\[
\hat{y}^o(k+1) = \hat{\theta}^T(k) \phi(k) = \hat{B}(z^{-1}, k) u(k) - \left(\hat{A}(z^{-1}, k) - 1\right) y(k+1)
\]

Understanding the notation: if \( B(z^{-1}) = b_0 + b_1 z^{-1} + \cdots + b_m z^{-m} \), then \( \hat{B}(z^{-1}, k) = \hat{b}_0(k) + \hat{b}_1(k) z^{-1} + \cdots + \hat{b}_m(k) z^{-m} \)

Remark: \( z^{-1} \)–shift operator; some references use \( q^{-1} \) instead
RLS is a series-parallel adjustable system

RLS in *a posteriori* form:

\[ \hat{y}(k+1) = \hat{B}(z^{-1}, k+1) u(k) - \left( \hat{A}(z^{-1}, k+1) - 1 \right) y(k+1) \]

prediction error:

\[ \varepsilon(k+1) = y(k+1) - \hat{y}(k+1) = \hat{A}(z^{-1}, k+1) y(k+1) - \hat{B}(z^{-1}, k+1) u(k) \]

A series-parallel structure: \( \hat{A}(z^{-1}, k+1) \)–in series with plant; \( \hat{B}(z^{-1}, k+1) \)–in parallel with the plant

**Observation**

If hyperstability holds such that \( \varepsilon(k+1) \to 0, \hat{y}(k+1) \to y(k+1) \), it seems fine to do instead:

\[ \hat{y}(k+1) = \hat{B}(z^{-1}, k+1) u(k) - \left( \hat{A}(z^{-1}, k+1) - 1 \right) \hat{y}(k+1) \quad (1) \]

i.e.

\[ u(k) \xrightarrow{\hat{B}(z^{-1}, k+1)} \frac{\hat{B}(z^{-1}, k+1)}{\hat{A}(z^{-1}, k+1)} \xrightarrow{\hat{A}(z^{-1}, k+1)} \hat{y}(k+1) \]

then we have a parallel structure

\[ u(k) \xrightarrow{\frac{\hat{B}(z^{-1}, k+1)}{\hat{A}(z^{-1}, k+1)}} \frac{\hat{B}(z^{-1}, k+1)}{\hat{A}(z^{-1}, k+1)} \xrightarrow{\hat{A}(z^{-1}, k+1)} \hat{y}(k+1) \]

\[ \varepsilon(k+1) \]

▶ it turns out this brings certain advantages
Other names

\[ u(k) \rightarrow \hat{B}(z^{-1}, k+1) \rightarrow \hat{A}(z^{-1}, k+1) \rightarrow \varepsilon(k+1) \]

is also called an output-error method

\[ u(k) \rightarrow \hat{B}(z^{-1}, k+1) \rightarrow \hat{B}(z^{-1}, k+1) \rightarrow \hat{A}(z^{-1}, k+1) \rightarrow \varepsilon(k+1) \]

is also called an equation-error method

Benefits of parallel algorithms

Intuition: when there is noise,

\[ u(k) \rightarrow \hat{B}(z^{-1}, k+1) \rightarrow \hat{B}(z^{-1}, k+1) \rightarrow \hat{A}(z^{-1}, k+1) \rightarrow \varepsilon(k+1) \]

provides better convergence of \( \hat{\theta} \) than

\[ u(k) \rightarrow \hat{B}(z^{-1}, k+1) \rightarrow \hat{B}(z^{-1}, k+1) \rightarrow \hat{A}(z^{-1}, k+1) \rightarrow \varepsilon(k+1) \]

We will talk about the PAA convergence in a few more lectures.
1. Big picture
   Series-parallel adjustable system (equation-error method)
   Parallel adjustable system (output-error method)

2. RLS-based parallel PAA
   Formulas
   Stability requirement for PAAs with fixed adaptation gain
   Stability requirement for PAAs with time-varying adaptation gain

3. Parallel PAAs with relaxed SPR requirements

4. PAAs with time-varying adaptation gains (revisit)

RLS based parallel PAA

\[
\begin{align*}
\hat{B}(z^{-1}, k+1) \\
\hat{A}(z^{-1}, k+1)
\end{align*}
\]

\[
\begin{align*}
&b_0 + b_1 z^{-1} + \cdots + b_m z^{-m} \\
&1 + a_1 z^{-1} + \cdots + a_n z^{-n}
\end{align*}
\]

PAA summary:

\begin{itemize}
  \item a priori
    \[
    \hat{\theta}(k+1) = \hat{\theta}(k) + \frac{F(k) \phi(k)}{1 + \phi^T(k) F(k) \phi(k)} \varepsilon^o(k+1)
    \]
  \item a posteriori
    \[
    \hat{\theta}(k+1) = \hat{\theta}(k) + F(k) \phi(k) \varepsilon(k+1)
    \]
    \[
    F^{-1}(k+1) = \lambda_1(k) F^{-1}(k) + \lambda_2(k) \phi(k) \phi^T(k)
    \]
\end{itemize}

\[
\phi^T(k) = [-\hat{y}(k), -\hat{y}(k-1), \ldots, -\hat{y}(k+1-n), u(k), \ldots, u(k-m)]
\]
Stability of RLS based parallel PAA

step 1: transformation to a feedback structure

**parameter estimation error:**

\[
\tilde{\theta}(k+1) = \tilde{\theta}(k) + F(k)\phi(k)\varepsilon(k+1)
\]

**a posteriori prediction error:**

\[
y(k+1) = B(z^{-1})u(k)\]

\[
\hat{y}(k+1) = A(z^{-1})y(k+1)
\]

\[
\hat{B}(z^{-1}, k+1) u(k) = \hat{A}(z^{-1}, k+1) \hat{y}(k+1)
\]

hence

\[
A(z^{-1})y(k+1) - \hat{A}(z^{-1}, k+1) \hat{y}(k+1) \pm A(z^{-1})\hat{y}(k+1) = B(z^{-1})u(k) - \hat{B}(z^{-1}, k+1) u(k)
\]

i.e.

\[
A(z^{-1})\varepsilon(k+1) = [B(z^{-1}) - \hat{B}(z^{-1}, k+1)]u(k)
\]

\[-[A(z^{-1}) - \hat{A}(z^{-1}, k+1)]\hat{y}(k+1)
\]

Look at [★]:

\[
B(z^{-1}) = b_0 + b_1z^{-1} + \cdots + b_mz^{-m}
\] gives

\[
\begin{bmatrix}
B(z^{-1}) - \hat{B}(z^{-1}, k+1)
\end{bmatrix} u(k)
\]

\[
-\begin{bmatrix}
A(z^{-1}) - \hat{A}(z^{-1}, k+1)
\end{bmatrix} \hat{y}(k+1)
\]

\[
\begin{bmatrix}
\begin{bmatrix}
\begin{bmatrix}
b_0 - \hat{b}_0(k+1)
b_1 - \hat{b}_1(k+1)
\vdots
b_m - \hat{b}_m(k+1)
\end{bmatrix}^T
\begin{bmatrix}
1
z^{-1}
\vdots
z^{-m}
\end{bmatrix}
\end{bmatrix}
u(k)
\end{bmatrix}
\]
Stability of RLS based parallel PAA

step 1: transformation to a feedback structure

Similarly, for \( A(z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_n z^{-n} \)

\[
\left[ \hat{A}(z^{-1}, k+1) - A(z^{-1}) \right] \hat{y}(k+1) = \begin{bmatrix} a_1 - \hat{a}_1(k+1) \\ a_2 - \hat{a}_2(k+1) \\ \vdots \\ a_n - \hat{a}_n(k+1) \end{bmatrix}^T \begin{bmatrix} -\hat{y}(k) \\ -\hat{y}(k-1) \\ \vdots \\ -\hat{y}(k+1-n) \end{bmatrix}
\]

Recall: \( \theta^T = [a_1, a_2, \ldots a_n, b_0, b_1, \ldots, b_m]^T \)

\( \phi(k) = [-\hat{y}(k), -\hat{y}(k-1), \ldots, -\hat{y}(k+1-n), u(k), \ldots, u(k-m)] \)

hence

\[
A(z^{-1}) \varepsilon(k+1) = \left[ B(z^{-1}) - \hat{B}(z^{-1}, k+1) \right] u(k) \\
- \left[ A(z^{-1}) - \hat{A}(z^{-1}, k+1) \right] \hat{y}(k+1) = -\tilde{\theta}^T(k+1) \phi(k)
\]
Stability of RLS based parallel PAA

step 2: Popov inequality

We will consider a simplified case with \( F(k) = F \succ 0 \):

\[
\sum_{k=0}^{k_1} \tilde{\theta}^T(k+1) \phi(k) \epsilon(k+1) \geq -\frac{1}{2} \tilde{\theta}^T(0) F^{-1} \tilde{\theta}(0)
\]

The nonlinear block is exactly the same as that in RLS, hence satisfying Popov inequality:

If \( G(z^{-1}) = \frac{1}{A(z^{-1})} \) is SPR, then the PAA is asymptotically hyperstable.

Remarks:

- RLS has an identity block: \( G(z^{-1}) = 1 \) which is independent of the plant
- \( 1/A(z^{-1}) \) depends on the plant (usually not SPR)
- several other PAAs are developed to relax the SPR condition
Stability of RLS based parallel PAA: extension

For the case of a time-varying $F(k)$ with

$$F^{-1}(k+1) = \lambda_1(k) F^{-1}(k) + \lambda_2(k) \phi(k) \phi^T(k)$$

the nonlinear block is more involved; we'll prove later, that it requires

$$\frac{1}{A(z^{-1})} - \frac{1}{2} \lambda,$$

where $\lambda = \max_k \lambda_2(k) < 2$, to be SPR

Outline

1. Big picture
   - Series-parallel adjustable system (equation-error method)
   - Parallel adjustable system (output-error method)

2. RLS-based parallel PAA
   - Formulas
     - Stability requirement for PAAs with fixed adaptation gain
     - Stability requirement for PAAs with time-varying adaptation gain

3. Parallel PAAs with relaxed SPR requirements

4. PAAs with time-varying adaptation gains (revisit)
Parallel algorithm with a fixed compensator

Instead of:

\[
\hat{\theta}(k+1) = \hat{\theta}(k) + \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}\varepsilon_o(k+1)
\]

\[
F^{-1}(k+1) = \lambda_1(k)F^{-1}(k) + \lambda_2(k)\phi(k)\phi^T(k)
\]

\[
\phi^T(k) = [-\hat{y}(k), -\hat{y}(k-1), \ldots, -\hat{y}(k+1-n), u(k), \ldots, u(k-m)]
\]

do:

\[
\hat{\theta}(k+1) = \hat{\theta}(k) + \frac{F(k)\phi(k)}{1 + \phi^T(k)F(k)\phi(k)}V^o(k+1)
\]

where

\[
V(k+1) = C(z^{-1})\varepsilon(k+1) = (c_0 + c_1z^{-1} + \ldots c_nz^{-n})\varepsilon(k+1)
\]

\[
V^o(k+1) = c_0\varepsilon^o(k+1) + c_1\varepsilon(k) + \ldots c_n\varepsilon(k-n+1)
\]

Parallel algorithm with a fixed compensator

The SPR requirement becomes

\[
\frac{C(z^{-1})}{A(z^{-1})} - \lambda \leq 2, \quad \lambda = \max_k \lambda_2(k) < 2
\]

should be SPR.

Remark:

- if \( c_i \)'s are close to \( a_i \)'s, (2) approximates \( 1 - \lambda/2 > 0 \), and hence is likely to be SPR
- problem: \( A(z^{-1}) \) is unknown \textit{a priori} for the assigning of \( C(z^{-1}) \)
- solution: make \( C(z^{-1}) \) to be adjustable as well
Parallel algorithm with an adjustable compensator

If \( A(z^{-1}) = 1 + a_1z^{-1} + \cdots + a_nz^{-n} \), let \( \hat{C}(z^{-1}) = 1 + \hat{c}_1z^{-1} + \cdots + \hat{c}_nz^{-n} \) and

\[
\nu(k+1) = \hat{C}(z^{-1}, k+1) \epsilon(k+1)
\]

\[
\nu^o(k+1) = \epsilon^o(k+1) + \sum_{i=1}^{n} \hat{c}_i(k) \epsilon(k+1-i)
\]

do

\[
\hat{\theta}_e(k+1) = \hat{\theta}_e(k) + \frac{F_e(k) \phi_e(k)}{1 + \phi^T_e(k) F_e(k) \phi_e(k)} \nu^o(k+1)
\]

\[
\hat{\theta}_e^T(k) = \begin{bmatrix} \hat{T}(k), \hat{c}_1(k), \ldots, \hat{c}_n(k) \end{bmatrix}
\]

\[
\phi^T_e(k) = \begin{bmatrix} \phi^T(k), -\epsilon(k), \ldots, -\epsilon(k+1-n) \end{bmatrix}
\]

\[
F_e^{-1}(k+1) = \lambda_1(k) F_e^{-1}(k) + \lambda_2(k) \phi_e(k) \phi^T_e(k)
\]

which has guaranteed **asymptotical stability**.

---

Lecture 17: PAA with Parallel Predictors

---

General PAA block diagram

---

<table>
<thead>
<tr>
<th>( H(z^{-1}) )</th>
<th>PAA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RLS/parallel predictor with adjustable compensator</td>
</tr>
<tr>
<td>( 1/A(z^{-1}) )</td>
<td>parallel predictor</td>
</tr>
<tr>
<td>( C(z^{-1})/A(z^{-1}) )</td>
<td>parallel predictor with fixed compensator</td>
</tr>
</tbody>
</table>
General PAA block diagram

- if \( F(k) = F \), \( H(z^{-1}) \) being SPR is sufficient for asymptotic stability
- if \( F(k) \) is time-varying, we will show next: \( H(z^{-1}) - \frac{1}{2} \lambda \) being SPR is sufficient for asymptotic stability

Outline

1. Big picture
   Series-parallel adjustable system (equation-error method)
   Parallel adjustable system (output-error method)

2. RLS-based parallel PAA
   Formulas
   Stability requirement for PAAs with fixed adaptation gain
   Stability requirement for PAAs with time-varying adaptation gain

3. Parallel PAAs with relaxed SPR requirements

4. PAAs with time-varying adaptation gains (revisit)
PAA with time-varying adaptation gains

\[ \tilde{\theta}^T(k+1) = \tilde{\theta}(k) + F(k)\phi(k) \varepsilon(k+1) \]

where \( F^{-1}(k+1) = \lambda_1(k)F^{-1}(k) + \lambda_2(k)\phi^T(k)\phi(k) \)

- unfortunately, the nonlinear block does not satisfy Popov inequality (not passive)

PAA with time-varying adaptation gains

a modification can re-gain the passivity of the feedback block
PAA with time-varying adaptation gains

a modification can re-gain the passivity of the feedback block

step 1: show that the following is passive

\[
\begin{align*}
\lambda_2(k)/2 \\
\lambda_2(k)/2 \\
\lambda_2(k)/2 \\
PAA \tilde{\theta}(k) \text{ Eq}
\end{align*}
\]

\[
\begin{align*}
w(k) = \tilde{\theta}^T(k+1)\phi(k) \\
\lambda_2(k)/2 \\
PAA \tilde{\theta}(k) \text{ Eq}
\end{align*}
\]

\[
\begin{align*}
\epsilon(k+1) \\
H(z^{-1}) \\
\lambda/2 \\
\lambda/2 \\
\lambda_2(k)/2 \\
\lambda_2(k)/2 \\
PAA \tilde{\theta}(k) \text{ Eq}
\end{align*}
\]

\[
\begin{align*}
\lambda_2(k)/2 \\
\lambda_2(k)/2 \\
PAA \tilde{\theta}(k) \text{ Eq}
\end{align*}
\]

step 2: the following is then passive

\[
\begin{align*}
\frac{1}{2} (\lambda - \lambda_2(k)) \\
\lambda_2(k)/2 \\
PAA \tilde{\theta}(k) \text{ Eq}
\end{align*}
\]

\[
\begin{align*}
w(k) \\
\frac{1}{2} (\lambda - \lambda_2(k)) \\
\lambda_2(k)/2 \\
PAA \tilde{\theta}(k) \text{ Eq}
\end{align*}
\]

\[
\begin{align*}
\epsilon(k+1) \geq 0 \\
H(z^{-1}) \\
\lambda/2 \\
\lambda/2 \\
\lambda_2(k)/2 \\
\lambda_2(k)/2 \\
PAA \tilde{\theta}(k) \text{ Eq}
\end{align*}
\]

note that it is a feedback connection of a passive block with

\[
\frac{1}{2} (\lambda - \lambda_2(k)) \geq 0
\]

step 3: SPR condition for the linear block

\[
H(z^{-1}) - \frac{\lambda}{2}
\]
Passivity of the sub nonlinear block

Consider:

\[ \lambda_2(k)/2 \]
\[ \text{PAA } \tilde{\theta}(k) \text{ Eq} \]
\[ \varepsilon(k+1) \]
\[ w(k) = \tilde{\theta}^T(k+1)\phi(k) \]
\[ s(k) \]

\[ s(k) = \varepsilon(k+1) + \frac{\lambda_2(k)}{2} \tilde{\theta}^T(k+1)\phi(k) \]
gives

\[ \sum_{k=0}^{k_1} w(k)s(k) \]
\[ = \sum_{k=0}^{k_1} \tilde{\theta}^T(k+1)\phi(k) \left[ \varepsilon(k+1) + \frac{\lambda_2(k)}{2} \tilde{\theta}^T(k+1)\phi(k) \right] \]
\[ \downarrow \text{note that } F^{-1}(k+1) = \lambda_1(k)F^{-1}(k) + \lambda_2(k)\phi(k)\phi^T(k) \]
\[ = \sum_{k=0}^{k_1} \tilde{\theta}^T(k+1)\phi(k)\varepsilon(k+1) + \frac{1}{2} \tilde{\theta}^T(k+1) \left[ F^{-1}(k+1) - \lambda_1(k)F^{-1}(k) \right] \tilde{\theta}(k+1) \]

which is no less than \(-\frac{1}{2} \tilde{\theta}^T(0)F^{-1}(0)\tilde{\theta}(0)\) as shown next.

Proof of passivity of the sub nonlinear block

\[ \tilde{\theta}(k+1) = \tilde{\theta}(k) + F(k)\phi(k)\varepsilon(k+1) \]

hence

\[ \sum_{k=0}^{k_1} \tilde{\theta}^T(k+1)\phi(k)\varepsilon(k+1) = \sum_{k=0}^{k_1} \tilde{\theta}^T(k+1)F^{-1}(k) \left( \tilde{\theta}(k+1) - \tilde{\theta}(k) \right) \]

Combining terms and after some algebra (see appendix), we get

\[ \sum_{k=0}^{k_1} w(k)s(k) = \sum_{k=0}^{k_1} \frac{1}{2} \tilde{\theta}^T(k+1)(1 - \lambda_1(k))F^{-1}(k)\tilde{\theta}(k+1) \]
\[ + \sum_{k=0}^{k_1} \frac{1}{2} \left[ \tilde{\theta}(k+1) - \tilde{\theta}(k) \right]^T F^{-1}(k) \left[ \tilde{\theta}(k+1) - \tilde{\theta}(k) \right] \]
\[ + \sum_{k=0}^{k_1} \frac{1}{2} \left[ \tilde{\theta}^T(k+1)F^{-1}(k)\tilde{\theta}(k+1) - \tilde{\theta}^T(k)F^{-1}(k)\tilde{\theta}(k) \right] \]
\[ = \left[ \frac{1}{2} \tilde{\theta}^T(k_1+1)F^{-1}(k_1)\tilde{\theta}(k_1+1) - \frac{1}{2} \tilde{\theta}^T(0)F^{-1}(0)\tilde{\theta}(0) \right] \]
Summary

In summary, the NL block indeed satisfies Popov inequality. For stability of PAA, it is sufficient that

\[ H(z^{-1}) - \frac{\lambda}{2} \text{ is SPR} \]

Appendix: derivation of (3)

\[
\sum_{k=0}^{k_1} \tilde{\theta}^T (k + 1) F^{-1} (k) \tilde{\theta} (k + 1) - \tilde{\theta}^T (k + 1) F^{-1} (k + 1) \tilde{\theta} (k + 1) \]

\[
= \sum_{k=0}^{k_1} \tilde{\theta}^T (k + 1) F^{-1} (k) \tilde{\theta} (k + 1) - \tilde{\theta}^T (k + 1) F^{-1} (k + 1) \tilde{\theta} (k + 1) + \frac{1}{2} \tilde{\theta}^T (k + 1) \left[ F^{-1} (k + 1) - \lambda_1 (k) F^{-1} (k) \right] \tilde{\theta} (k + 1)
\]

\[
= \sum_{k=0}^{k_1} \tilde{\theta}^T (k + 1) F^{-1} (k) \tilde{\theta} (k + 1) - \tilde{\theta}^T (k + 1) F^{-1} (k + 1) \tilde{\theta} (k + 1) + \frac{1}{2} \tilde{\theta}^T (k + 1) \left[ F^{-1} (k + 1) - \lambda_1 (k) F^{-1} (k) \right] \tilde{\theta} (k + 1)
\]

\[
+ \frac{1}{2} \tilde{\theta}^T (k + 1) F^{-1} (k + 1) \tilde{\theta} (k + 1)
\]

(4)

The term \( \frac{1}{2} \tilde{\theta}^T (k + 1) (1 - \lambda_1 (k)) F^{-1} (k) \tilde{\theta} (k + 1) \) is always none-negative if \( 1 - \lambda_1 (k) \geq 0 \), which is the assumption in the forgetting factor definition. We only need to worry about

\[
\sum_{k=0}^{k_1} \frac{1}{2} \tilde{\theta}^T (k + 1) F^{-1} (k) \tilde{\theta} (k + 1) - \tilde{\theta}^T (k + 1) F^{-1} (k) \tilde{\theta} (k) + \frac{1}{2} \tilde{\theta}^T (k + 1) F^{-1} (k + 1) \tilde{\theta} (k + 1)
\]

(5)
Appendix: derivation of (3)

The underlined terms are already available in (5). Adding and substracting terms in (5) gives

\[ \sum_{k=0}^{k_1} \frac{1}{2} \tilde{\theta}^T (k+1) F^{-1} (k) \tilde{\theta}(k+1) - \frac{1}{2} \tilde{\theta}^T (k) F^{-1} (k) \tilde{\theta}(k) \]

\[ + \frac{1}{2} \tilde{\theta}^T (k+1) F^{-1} (k+1) \tilde{\theta} (k+1) - \tilde{\theta}^T (k+1) F^{-1} (k) \tilde{\theta}(k) + \frac{1}{2} \tilde{\theta}^T (k) F^{-1} (k) \tilde{\theta}(k) \]

\[ = \sum_{k=0}^{k_1} \frac{1}{2} \tilde{\theta}^T (k+1) F^{-1} (k) \tilde{\theta}(k+1) - \frac{1}{2} \tilde{\theta}^T (k) F^{-1} (k) \tilde{\theta}(k) \]

\[ + \frac{1}{2} \left[ \tilde{\theta}(k+1) - \tilde{\theta}(k) \right]^T F^{-1} (k) \left[ \tilde{\theta}(k+1) - \tilde{\theta}(k) \right] \geq 0 \]

Summarizing, we get

\[ \sum_{k=0}^{k_1} w(k) s(k) = \sum_{k=0}^{k_1} \frac{1}{2} \tilde{\theta}^T (k+1) (1 - \lambda_1 (k)) F^{-1} (k) \tilde{\theta}(k+1) \]

\[ + \sum_{k=0}^{k_1} \frac{1}{2} \left[ \tilde{\theta}(k+1) - \tilde{\theta}(k) \right]^T F^{-1} (k) \left[ \tilde{\theta}(k+1) - \tilde{\theta}(k) \right] \]

\[ + \sum_{k=0}^{k_1} \frac{1}{2} \left[ \tilde{\theta}^T (k+1) F^{-1} (k) \tilde{\theta}(k+1) - \tilde{\theta}^T (k) F^{-1} (k) \tilde{\theta}(k) \right] \]

\[ \geq 0 \]

\[ \frac{1}{2} \tilde{\theta}^T (k_1+1) F^{-1} (k_1) \tilde{\theta}(k_1+1) - \frac{1}{2} \tilde{\theta}^T (0) F^{-1} (0) \tilde{\theta}(0) \]

hence

\[ \sum_{k=0}^{k_1} w(k) s(k) \geq - \frac{1}{2} \tilde{\theta}^T (0) F^{-1} (0) \tilde{\theta}(0) \]
Summary

1. Big picture
   Series-parallel adjustable system (equation-error method)
   Parallel adjustable system (output-error method)

2. RLS-based parallel PAA
   Formulas
   Stability requirement for PAAs with fixed adaptation gain
   Stability requirement for PAAs with time-varying adaptation gain

3. Parallel PAAs with relaxed SPR requirements

4. PAAs with time-varying adaptation gains (revisit)
Lecture 18: Parameter Convergence in PAAs

Big picture

why are we learning this:
Consider a series-parallel PAA

\[
\begin{align*}
B(z^{-1},k+1) & \quad u(k) \\
B(z^{-1}) & \quad y(k+1) \\
\hat{A}(z^{-1},k+1) & \quad \epsilon(k+1)
\end{align*}
\]

where the plant is stable.

(Hyper)stability of PAA gives

\[
\lim_{k \to \infty} \epsilon(k) = \lim_{k \to \infty} \left\{ -\tilde{\theta}^T(k) \phi(k-1) \right\} = 0
\]

But this does not guarantee

\[
\lim_{k \to \infty} \hat{\theta}(k) = 0 \iff \lim_{k \to \infty} \hat{\theta}(k) = \theta
\]
Consider a new polynomial $\alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_m z^{-m-n}$

\[
\left[ \tilde{A}(z^{-1}) B(z^{-1}) - A(z^{-1}) \tilde{B}(z^{-1}) \right] u(k) \rightarrow 0
\]

where $\tilde{A}(z^{-1}) = \hat{A}(z^{-1}) - A(z^{-1})$.

Two questions we are going to discuss for assuring $\tilde{\theta} = 0$:

- is $\alpha_i = 0$ true iff $\tilde{\alpha}_i = 0$, $\tilde{b}_i = 0$ (i.e., $\{\alpha_i\} = 0 \iff \tilde{\theta} = 0$)?

- if $\alpha_i \neq 0$, can $\left[ \alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_m z^{-m-n} \right] u(k) = 0$?
Parameter convergence condition

Qs 1: $\alpha_i = 0 \iff \tilde{a}_i = 0, \tilde{b}_i = 0$? Ans: yes if $B(z^{-1})$ and $A(z^{-1})$ are coprime

$$\alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_{m+n} z^{-m-n} = \tilde{A}(z^{-1}) B(z^{-1}) - A(z^{-1}) \tilde{B}(z^{-1})$$

- the right hand side is composed of terms of $\tilde{a}_i b_j$ and $a_p \tilde{b}_q$
- comparing coefficients of $z^{-k}$ gives

$$\begin{bmatrix}
\alpha_0 \\
\alpha_1 \\
\vdots \\
\alpha_{m+n} \\
\end{bmatrix} = S 
\begin{bmatrix}
\tilde{b}_0 \\
\tilde{b}_1 \\
\vdots \\
\tilde{b}_m \\
\end{bmatrix}, \ S: \text{a square matrix composed of } \{a_i, b_j\}$$

- turns out $S$ is non-singular if and only if $B(z^{-1})$ and $A(z^{-1})$ are coprime (recall the theorem discussed in repetitive control)

Parameter convergence condition

Qs 2: if $\alpha_i \neq 0$, can $[\alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_{m+n} z^{-m-n}] u(k) = 0$?

Simple example with $n + m = 2$, $u(k) = \cos(\omega k) = \Re\{e^{j\omega k}\}$:

$$[\alpha_0 + \alpha_1 z^{-1} + \alpha_2 z^{-2}] u(k) \rightarrow 0$$

$$\iff [\alpha_0 + \alpha_1 z^{-1} + \alpha_2 z^{-2}] e^{j\omega k} \rightarrow 0$$

which can be achieved either by $\alpha_0 = \alpha_1 = \alpha_2 = 0$ (the desired case) or by

$$\left(1 - e^{-j\omega z^{-1}}\right) \left(1 - e^{j\omega z^{-1}}\right) e^{j\omega k} = \left[1 - 2\cos(\omega) z^{-1} + z^{-2}\right] e^{j\omega k} \rightarrow 0$$
Parameter convergence condition

Qs 2: if $\alpha_i \neq 0$, can $[\alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_{m+n} z^{-m-n}] u(k) = 0$?

If, however,

$$u(k) = c_1 \cos(\omega_1 k) + c_2 \cos(\omega_2 k) = \Re \left\{ c_1 e^{j\omega_1 k} + c_2 e^{j\omega_2 k} \right\}$$

then

$$[\alpha_0 + \alpha_1 z^{-1} + \alpha_2 z^{-2}] u(k) \to 0$$

can only be achieved by $\alpha_0 = \alpha_1 = \alpha_2 = 0$ (the desired case).

Observations:

- complex roots of $\alpha_0 + \alpha_1 z^{-1} + \alpha_2 z^{-2}$ always come as pairs
- impossible for $\alpha_0 + \alpha_1 z^{-1} + \alpha_2 z^{-2}$ to have four roots at $e^{\pm j\omega_1}$ and $e^{\pm j\omega_2}$
- if the total number of parameters $n + m = 3$, $u(k)$ should contain at least $2 \left( \frac{n+m+1}{2} \right)$ frequency components

Parameter convergence condition

general case:

$$\alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_{m+n} z^{-m-n} = 0$$

- number of the pairs of roots = $(m+n)/2$, if $m+n$ is even
- number of the pairs of roots = $(m+n-1)/2$ if $m+n$ is odd

Theorem (Persistant of excitation for PAA convergence)

For PAAs with a series-parallel predictor, the convergence

$$\lim_{k \to \infty} \hat{\theta}_i(k) = \theta_i(k)$$

is assured if

1. the plant transfer function is irreducible
2. the input signal contains at least $1 + (m+n)/2$ (for $n + m$ even) or $(m+n+1)/2$ (for $m + n$ odd) independent frequency components.
Effect of noise on parameter identification

Noise modeling:

\[ u(k) \xrightarrow{B(z^{-1})} y(k+1) + w(k+1) \]

i.e.

\[ A(z^{-1}) y(k+1) = B(z^{-1}) u(k) + w(k+1) \]
\[ y(k+1) = \theta^T \phi(k) + w(k+1) \]

or

\[ u(k) \xrightarrow{\frac{B(z^{-1})}{A(z^{-1})}} y(k+1) + n(k+1) \]

i.e.

\[ y(k+1) = \theta^T \phi(k) + A(z^{-1}) n(k+1) \]

which is equivalent to \( w(k+1) = A(z^{-1}) n(k+1) \) in the first case.
Effect of noise on parameter identification

plant output: \( y(k+1) = \theta^T \phi(k) + w(k+1) \)

predictor output: \( \hat{y}(k+1) = \hat{\theta}^T (k+1) \phi(k) \)

\( a \ posteriori \) prediction error:
\[
\varepsilon(k+1) = y(k+1) - \hat{y}(k+1) = -\hat{\theta}^T (k+1) \phi(k) + w(k+1)
\]

PAA:
\[
\hat{\theta}(k+1) = \hat{\theta}(k) + F(k) \phi(k) \varepsilon(k+1) = \hat{\theta}(k) + F(k) \phi(k) \varepsilon(k+1) + F(k) \phi(k) w(k+1)
\]
Effect of noise on parameter identification

plant output: \( y(k+1) = \theta^T \phi(k) + w(k+1) \)

predictor output: \( \hat{y}(k+1) = \hat{\theta}^T (k+1) \phi(k) \)

\( a \ posteriori \) prediction error:
\[
\varepsilon(k+1) = y(k+1) - \hat{y}(k+1) = -\hat{\theta}^T (k+1) \phi(k) + w(k+1)
\]

PAA:
\[
\hat{\theta}(k+1) = \hat{\theta}(k) + F(k) \phi(k) \varepsilon(k+1)
\]
\[
= \hat{\theta}(k) + F(k) \phi(k) \varepsilon(k+1) + F(k) \phi(k) w(k+1)
\]

- \( F(k) \phi(k) w(k+1) \) is integrated by PAA
- need:
  \[ E[\phi(k) w(k+1)] = 0 \]

and a vanishing adaptation gain \( F(k) \):
\[
F^{-1}(k+1) = \lambda_1(k) F^{-1}(k) + \lambda_2(k) \phi(k) \phi^T(k), \quad \lambda_1(k) \xrightarrow{k \to \infty} 1 \quad \text{and} \quad 0 < \lambda_2(k) < 2
\]

Series-parallel PAA convergence condition

\[
\hat{\theta}(k+1) = \hat{\theta}(k) + F(k) \phi(k) \varepsilon(k+1) + F(k) \phi(k) w(k+1)
\]

In series-parallel PAA:
\[
\phi(k) = [-y(k), -y(k-1), \ldots, -y(k-n+1), \]
\[
u(k), u(k-1), \ldots, u(k-m)]^T
\]

\[ E[\phi(k) w(k+1)] = 0 \]

is achieved if
- \( w(k+1) \) is white, and
- \( u(k) \) and \( w(k+1) \) are independent
Series-parallel PAA convergence condition

$w(k+1) \downarrow \downarrow u(k) \rightarrow \rightarrow B(z^{-1}) + \rightarrow \rightarrow \frac{1}{A(z^{-1})} y(k+1) \rightarrow \rightarrow$

Issues: $w(k+1)$ is rarely white, e.g.,

$u(k) \rightarrow \rightarrow B(z^{-1}) + n(k+1) \rightarrow \rightarrow \frac{1}{A(z^{-1})} y(k+1) \rightarrow \rightarrow$

where the output measurement noise $n(k+1)$ is usually white but

$y(k+1) = \theta^T \phi(k) + A(z^{-1}) n(k+1)$

so $w(k+1)$ is not white.

Parallel PAA convergence condition

In parallel PAA:

$\phi(k) = [-\hat{y}(k), -\hat{y}(k-1), \ldots, -\hat{y}(k-n+1), u(k), u(k-1), \ldots, u(k-m)]^T$

$E[\phi(k)w(k+1)] = 0$ does not require $w(k+1)$ to be white as $\hat{y}(k)$ does not depend on $w(k+1)$ by design
Summary

Theorem (Series-parallel PAA convergence condition)
When the predictor is of series-parallel type, the PAA with a vanishing adaptation gain has unbiased convergence when
i. $u(k)$ is rich in frequency (persistent excitation) and is independent from the noise $w(k+1)$
ii. $w(k+1)$ is white

Theorem (Parallel PAA convergence condition)
When the predictor is of parallel type, the PAA with vanishing adaptation gain has unbiased convergence when
i. $u(k)$ satisfies the persistent excitation condition
ii. $u(k)$ is independent from $w(k+1)$

Note: parallel predictors have more strict stability requirements

Outline

1. Big picture
2. Parameter convergence conditions
3. Effect of noise on parameter identification
4. Convergence improvement in the presence of stochastic noises
5. Effect of deterministic disturbances
Convergence improvement when there is noise

**Extended least squares**

If the effect of noise can be expressed as

\[
\begin{align*}
\hat{y}(k+1) &= \hat{\theta}_e^T (k+1) \phi_e(k) \\
\phi_e^T(k) &= \left[ \phi^T(k), n(k), \ldots, n(k-nC+1) \right]^T \\
\end{align*}
\]

but \(n(k), \ldots, n(k-nC+1)\) are not measurable. However, if \(\hat{\theta}_e\) is close to \(\theta_e\), then

\[
\epsilon(k+1) = y(k+1) - \hat{y}(k+1) \approx n(k+1)
\]

**Extended least squares uses**

\[
\begin{align*}
\hat{y}(k+1) &= \hat{\theta}_e^T (k+1) \phi_e^*(k) \\
\phi_e^*(k) &= \left[ \phi^T(k), \epsilon(k), \ldots, \epsilon(k-nC+1) \right]^T \\
\end{align*}
\]

where \(\epsilon(k) = y(k) - \hat{y}(k)\)
Convergence improvement when there is noise
output error method with adjustable compensator

\[
\hat{\theta}_e(k+1) = \hat{\theta}_e(k) + \frac{F_e(k) \phi_e(k)}{1 + \phi_e^T(k) F_e(k) \phi_e(k)} v^o(k+1)
\]

\[
\hat{\theta}_e^T(k) = \begin{bmatrix} \hat{\theta}^T(k), \hat{c}_1(k), \ldots, \hat{c}_n(k) \end{bmatrix}
\]

\[
\phi_e^T(k) = \begin{bmatrix} \phi^T(k), -\epsilon(k), \ldots, -\epsilon(k+1-n) \end{bmatrix}
\]

\[
F_e^{-1}(k+1) = \lambda_1(k) F_e^{-1}(k) + \lambda_2(k) \phi_e(k) \phi_e^T(k)
\]

Stability condition:

\[
1 - \frac{\lambda}{2} \text{ is SPR; } \lambda = \max_k \lambda_2(k) < 2
\]

Convergence condition: depend on properties of the disturbance and \(A(z^{-1})\); see details in ME233 reader
Different recursive identification algorithms

▶ there are more PAAs for improved convergence

▶ each algorithm suits for a certain model of plant + disturbance

Outline

1. Big picture

2. Parameter convergence conditions

3. Effect of noise on parameter identification

4. Convergence improvement in the presence of stochastic noises

5. Effect of deterministic disturbances
Effect of deterministic disturbances

Intuition: if the disturbance structure is known, it can be included in PAA for improved performance.

Example (constant disturbance):

\[
\begin{align*}
y(k+1) &= -\sum_{i=1}^{n} a_i y(k+1-i) + \sum_{i=0}^{m} b_i u(k-i) + d = \theta^T \phi(k) + d
\end{align*}
\]

Approach 1: enlarge the model as

\[
y(k+1) = \begin{bmatrix} \theta^T, d \end{bmatrix} \begin{bmatrix} \phi(k) \\ 1 \end{bmatrix} = \theta_e^T \phi_e(k)
\]

and construct PAA on \( \theta_e \).

Effect of deterministic disturbances

\[
\begin{align*}
y(k+1) &= -\sum_{i=1}^{n} a_i y(k+1-i) + \sum_{i=0}^{m} b_i u(k-i) + d = \theta^T \phi(k) + d
\end{align*}
\]

Approach 2: notice that \((1-z^{-1})d = 0\), we can do

\[
y(k+1) \rightarrow 1-z^{-1} \rightarrow y_f(k+1); \quad u(k+1) \rightarrow 1-z^{-1} \rightarrow u_f(k+1);
\]

and have a new “disturbance-free” model for PAA:

\[
y_f(k+1) = -\sum_{i=1}^{n} a_i y_f(k+1-i) + \sum_{i=0}^{m} b_i u_f(k-i)
\]
Effect of deterministic disturbances

\[ u(k) \rightarrow B(z^{-1}) \rightarrow y(k) + d \rightarrow y(k+1) \]

Similar considerations can be applied to the cases when \( d \) is sinusoidal, repetitive, etc.
Lecture 19: Adaptive Control based on Pole Assignment

Big picture

reasons for adaptive control:

▶ unknown or time-varying plants
▶ unknown or time-varying disturbance (with known structure but unknown coefficients)

two main steps:

▶ decide the controller structure
▶ design PAA to adjust the controller parameters
two ways of adaptation process:

▶ indirect adaptive control: adapt the plant parameters and use them in the updated controller
▶ direct adaptive control: directly adapt the controller parameters
**RST control structure**

Plant:

\[ G(z^{-1}) = \frac{z^{-d}B(z^{-1})}{A(z^{-1})} \quad B(z^{-1}) = b_0 + b_1z^{-1} + \cdots + b_mz^{-m}, \quad b_0 \neq 0 \]

\[ A(z^{-1}) = 1 + a_1z^{-1} + \cdots + a_nz^{-n} \]

Consider RST type controller:

\[ r^*(k) \xrightarrow{T(z^{-1})} r(k) \xrightarrow{1/S(z^{-1})} u(k) \xrightarrow{z^{-d}B(z^{-1})/A(z^{-1})} y(k) \]

Closed-loop transfer function:

\[ \frac{Y(z^{-1})}{R(z^{-1})} = \frac{z^{-d}B(z^{-1})}{A(z^{-1})S(z^{-1}) + z^{-d}B(z^{-1})R(z^{-1})} \]

**Pole placement**

Closed-loop pole assignment via:

\[ z^{-d}B(z^{-1})R(z^{-1}) + S(z^{-1})A(z^{-1}) = D(z^{-1}) \]

- this is a polynomial (Diophantine) equation
- design \( D(z^{-1}) \), find \( S(z^{-1}) \) and \( R(z^{-1}) \) by coefficient matching
Pole placement for plants with stable zeros

If zeros of plant are all stable, they can be cancelled. We can do

\[ S(z^{-1}) = S'(z^{-1})B(z^{-1}) \]
\[ D(z^{-1}) = D'(z^{-1})B(z^{-1}) \]

yielding

\[ z^{-d} R(z^{-1}) + S'(z^{-1})A(z^{-1}) = D'(z^{-1}) \]

where the polynomials should match order:

\[ S'(z^{-1}) = 1 + s'_1 z^{-1} + \cdots + s'_{d-1} z^{-(d-1)} \]
\[ R(z^{-1}) = r_0 + r_1 z^{-1} + \cdots + r_{n-1} z^{-(n-1)} \]

The transfer function from \( r(k) \) to \( y(k) \) is thus

\[
G_{r \rightarrow y}(z^{-1}) = \frac{z^{-d} B(z^{-1})}{S(z^{-1})A(z^{-1}) + z^{-d} B(z^{-1}) R(z^{-1})} = \frac{z^{-d}}{D'(z^{-1})}
\]

Hence we can let

\[ T(z^{-1}) = D'(z^{-1}), \quad r^*(k) = y_d(k + d) \]

which means

\[ D'(z^{-1}) [y(k + d) - y_d(k + d)] = 0 \]

- this is the desired control goal, you can compare it with the goal in system identification: \( y(k + 1) - \hat{y}(k + 1) = 0 \)
- next we express \( D'(z^{-1}) y(k + d) \) and \( D'(z^{-1}) y_d(k + d) \) in forms similar to “\( \theta^T \phi(k) \)”
Pole placement for plants with stable zeros

the \( D'(z^{-1}) y(k+d) \) term

For a tuned pole placement with known plant model:

- \( z^{-d} R(z^{-1}) + S'(z^{-1}) A(z^{-1}) = D'(z^{-1}) \) yields

\[
A(z^{-1}) S'(z^{-1}) y(k+d) = D'(z^{-1}) y(k+d) - z^{-d} R(z^{-1}) y(k+d)
\]

- and the plant model

\[
\begin{align*}
\begin{array}{c|c}
  u(k) & \multicolumn{1}{c}{z^{-d} B(z^{-1})} \\
  \hline
  \multicolumn{1}{c}{A(z^{-1})} & y(k)
\end{array}
\end{align*}
\]

gives

\[
A(z^{-1}) y(k+d) = B(z^{-1}) u(k)
\]

Combining the two gives

\[
D'(z^{-1}) y(k+d) = B(z^{-1}) S'(z^{-1}) u(k) + R(z^{-1}) y(k) \quad (2)
\]

Pole placement for plants with stable zeros

the \( D'(z^{-1}) y(k+d) \) term

We will now simplify (2). Note first:

\[
S(z^{-1}) = B(z^{-1}) S'(z^{-1}) = s_0 + s_1 z^{-1} + \cdots + s_{d+m-1} z^{-(d+m-1)}
\]

hence

\[
D'(z^{-1}) y(k+d) = \underbrace{B(z^{-1}) S'(z^{-1}) u(k) + R(z^{-1}) y(k)}_{\theta_c^T \phi(k)}
\]

where

\[
\theta_c^T = [s_0, s_1, \ldots, s_{d+m-1}, r_0, \ldots, r_{n-1}]
\]
\[
\phi(k) = [u(k), u(k-1), \ldots, u(k-d-m+1), y(k), \ldots, y(k-n+1)]^T
\]
Pole placement for plants with stable zeros

For the actual adaptive $S(z^{-1})$ and $R(z^{-1})$, the control law is

$$y_d(k + d) \rightarrow D'(z^{-1}) r(k) \rightarrow \frac{1}{\hat{S}(z^{-1})} u(k)$$

i.e.

$$u(k) = \frac{1}{\hat{S}(z^{-1})} \left[ D'(z^{-1}) y_d(k + d) - \hat{R}(z^{-1}) y(k) \right]$$

yielding

$$D'(z^{-1}) y_d(k + d) = \hat{S}(z^{-1}) u(k) + \hat{R}(z^{-1}) y(k) = \hat{\theta}_c^T \phi(k) \quad (3)$$

This is a direct adaptive control: no explicit $B(z^{-1})$ and $A(z^{-1})$ in $\hat{\theta}_c$

Hence we can define

$$\varepsilon(k + d) = D'(z^{-1}) y(k + d) - \hat{\theta}_c^T (k + d) \phi(k)$$

or equivalently

a posteriori: $$\varepsilon(k) = D'(z^{-1}) y(k) - \hat{\theta}_c^T (k) \phi(k - d)$$

a priori: $$\varepsilon^o(k) = D'(z^{-1}) y(k) - \hat{\theta}_c^T (k - 1) \phi(k - d)$$

and apply parameter adaptation for $\theta_c$, e.g., using series-parallel predictors

$$\hat{\theta}_c(k) = \hat{\theta}_c(k - 1) + \frac{F(k - 1) \phi(k - d)}{1 + \phi(k - d)^T F(k - 1) \phi(k - d)} \varepsilon^o(k)$$

$$F^{-1}(k) = \lambda_1(k) F^{-1}(k - 1) + \lambda_2(k) \phi(k - d) \phi^T(k - d)$$
Comparison with system identification

Comparison:
standard system identification:

\[ y(k + 1) = \theta^T \phi(k) \]
\[ \hat{\theta}(k + 1) = \hat{\theta}(k) + F(k) \phi(k) \varepsilon(k + 1) \]
\[ \varepsilon(k + 1) = \frac{\varepsilon^o(k + 1)}{1 + \phi^T(k) F(k) \phi(k)} \]

adaptive pole placement:

\[ D'(z^{-1}) y(k) = \theta_c^T \phi(k - d) \]
\[ \hat{\theta}_c(k) = \hat{\theta}_c(k - 1) + F(k - 1) \phi(k - d) \varepsilon(k) \]
\[ \varepsilon(k) = \frac{\varepsilon^o(k)}{1 + \phi^T(k - d) F(k - 1) \phi(k - d)} \]

Pole placement for plants with stable zeros

PAA Stability

First obtain the \textit{a posteriori} dynamics of the parameter error:

\[ \hat{\theta}_c(k) = \hat{\theta}_c(k - 1) + F(k - 1) \phi(k - d) \varepsilon(k) \]
\[ \Rightarrow \tilde{\theta}_c(k) = \tilde{\theta}_c(k - 1) + F(k - 1) \phi(k - d) \varepsilon(k) \]

In the mean time

\[ \varepsilon(k) = D'(z^{-1}) y(k) - \hat{\theta}_c^T(k) \phi(k - d) \]
\[ \downarrow \text{recall} \ D'(z^{-1}) y(k + d) = \theta_c^T \phi(k) \]
\[ = \theta_c^T \phi(k - d) - \hat{\theta}_c^T(k) \phi(k - d) \]
\[ = -\tilde{\theta}_c(k)^T \phi(k - d) \]
Pole placement for plants with stable zeros

PAA Stability

\[ \varepsilon(k) = -\tilde{\theta}_c(k)^T \phi(k - d) \]

\[ \tilde{\theta}_c(k) = \tilde{\theta}_c(k - 1) + F(k - 1) \phi(k - d) \varepsilon(k) \]

The PAA thus is in a standard series-parallel structure with the LTI block being 1. Hyperstability easily follows, which gives

\[ \lim_{k \to \infty} \varepsilon(k) = \frac{D'(z^{-1})y(k) - \hat{\theta}_c^T(k - 1) \phi(k - d)}{1 + \phi^T(k - d) F(k - 1) \phi(k - d)} \to 0 \]

Similar as before, to prove \( \varepsilon^o(k) = D'(z^{-1})(y(k) - y_d(k)) \to 0 \), we need to show that \( \phi(k - d) \) is bounded, which can be shown to be true (see ME233 reader).

Design procedure:

Step 1: choose desired \( D'(z^{-1}) \) (deg \( D'(z^{-1}) \leq n + d - 1 \)). The overall closed-loop characteristic polynomial is \( D'(z^{-1})B(z^{-1}) \).

Step 2: determine orders in the Diophantine equation \( S'(z^{-1}) \) (deg \( S'(z^{-1}) = d - 1 \)) and \( R(z^{-1}) \) (deg \( R(z^{-1}) = n - 1 \)).

Step 3: at each time index, do the following:

- apply an appropriate PAA to estimate the coefficients of \( S(z^{-1}) = S'(z^{-1})B(z^{-1}) \) and \( R(z^{-1}) \), based on the reparameterized plant model

\[ D'(z^{-1})y(k) = \theta_c^T \phi(k - d) \]

- use the estimated parameter vector, \( \hat{\theta}_c(k) \), to compute the control signal \( u(k) \) according to

\[ u(k) = \frac{1}{\hat{S}(z^{-1})} \left[ D'(z^{-1})y_d(k + d) - \hat{R}(z^{-1})y(k) \right] \]
Example
Consider a plant (discrete-time model of \(1/(ms + b)\) with an extra delay)
\[ G_p(z^{-1}) = \frac{z^{-2}b_0}{1 + a_1z^{-1}} \]
We have \(B(z^{-1}) = b_0\) (\(m = 0\) here); \(A(z^{-1}) = 1 + a_1z^{-1}\) (\(n = 1\) here); \(d = 2\). The pole placement equation is
\[
(1 + a_1z^{-1})(1 + s_1'z^{-1}) + z^{-2}r_0 = 1 + d_1'z^{-1} + d_2z^{-2}
\Rightarrow s_1' = d_1' - a_1, r_0 = d_2' - a_1(d_1' - a_1)
\]
and \(S(z^{-1}) = S'(z^{-1})B(z^{-1}) = s_0 + s_1z^{-1}; R(z^{-1}) = r_o\)
\[ u(k) = \frac{1}{\hat{S}(z^{-1})} \left[ D'(z^{-1})y_d(k + d) - \hat{R}(z^{-1})y(k) \right] \]
\[ = \frac{1}{\hat{s}_0(k)} \left[ D'(z^{-1})y_d(k + 2) - \hat{r}_0(k)y(k) - \hat{s}_1(k)u(k - 1) \right] \]

Remark
Parameter convergence is achieved if the excitation \(y_d\) is rich in frequency (which may not be assured in practice). Yet the performance goal of making \(D'(z^{-1})[y(k) - y_d(k)]\) small can still be achieved even if \(y_d\) is not rich in frequency.
If the disturbance structure is known, we can estimate its parameters for disturbance cancellation. Consider, e.g.,

\[ y(k) = \frac{z^{-d}B(z^{-1})}{A(z^{-1})} [u(k) + d(k)] \]

where \( B(z^{-1}) \) is cancellable and the disturbance satisfies

\[ W(z^{-1}) d(k) = (1 - z^{-1}) d(k) = 0 \]

the deterministic control law should be:

\[ u(k) = \frac{1}{S(z^{-1})} \left[ -R(z^{-1}) y(k) + D'(z^{-1}) y_d(k + d) \right] - d \]

Disturbance cancellation

\[ u(k) = \frac{1}{S(z^{-1})} \left[ -R(z^{-1}) y(k) + D'(z^{-1}) y_d(k + d) \right] - d \]

can be equivalently represented as

\[ D'(z^{-1}) y_d(k + d) = \theta_c^T \phi(k) + d^*, \quad d^* = S(z^{-1}) d \]

\[ = \theta_{ce}^T \phi_e(k), \quad \theta_{ce} = \left[ \theta_c^T, d^* \right]^T, \quad \phi_e(k) = \left[ \phi^T(k), 1 \right] \]

In the adaptive case:

\[ D'(z^{-1}) y_d(k + d) = \hat{\theta}_{ce}^T (k + d) \phi_e(k) \]

where \( \hat{\theta}_{ce}(k) \) is updated via a PAA, e.g.

\[ \hat{\theta}_{ce}(k) = \hat{\theta}_{ce}(k - 1) + \frac{F(k - 1) \phi_e(k - d) \left[ D'(z^{-1}) y(k) - \hat{\theta}_{ce}^T (k - 1) \phi_e(k - d) \right]}{1 + \phi_e^T (k - d) F(k - 1) \phi_e(k - d)} \]
1. Big picture

2. Adaptive pole placement
   - Cancellable $B(z^{-1})$
     - Remark

3. Extension: adaptive pole placement with disturbance cancellation

4. Pole placement with no cancellation of $B(z^{-1})$

5. Indirect adaptive pole placement

**Uncancellable $B(z^{-1})$**

\[ Y(z^{-1}) = \frac{R(z^{-1})}{R(z^{-1})} = \frac{z^{-d}B(z^{-1})}{A(z^{-1})S(z^{-1}) + z^{-d}B(z^{-1})R(z^{-1})} = \frac{z^{-d}B(z^{-1})}{D(z^{-1})} \]

If $B(z^{-1})$ contains unstable roots or if we don’t want to cancel it, we can do

\[ r^*(k) = y_d(k+d) \rightarrow T(z^{-1}) = \frac{D(z^{-1})}{B(1)} \rightarrow \frac{z^{-d}B(z^{-1})}{D(z^{-1})} \rightarrow y(k) \]

\[ \Rightarrow D(z^{-1}) \left[ y(k+d) - \frac{B(z^{-1})}{B(1)}y_d(k+d) \right] = 0 \]
Uncancellable $B\left(z^{-1}\right)$

or

$$r^*(k) = y_d(k+d) \rightarrow T\left(z^{-1}\right) = \frac{D(z^{-1})B(z)}{[B(1)]^2} r(k) \rightarrow \frac{z^{-d}B(z^{-1})}{D(z^{-1})} y(k)$$

$$\Rightarrow D\left(z^{-1}\right) \left[ y(k+d) - \frac{B\left(z^{-1}\right)B(z)}{[B(1)]^2} y_d(k+d) \right] = 0$$

which gives zero phase error tracking.

Remark: can also partially cancel the stable parts of $B\left(z^{-1}\right)$

Note: now we explicitly need $B(1)$ and/or $B(z)$ in $T\left(z^{-1}\right) \Rightarrow$ need adaptation to find the plant parameters $\Rightarrow$ indirect adaptive control

---

**Indirect adaptive pole placement: big picture**

Consider the plant $z^{-d}B\left(z^{-1}\right)/A\left(z^{-1}\right)$.

Pole placement with known plant parameters:

$$A\left(z^{-1}\right)S\left(z^{-1}\right) + z^{-d}B\left(z^{-1}\right)R\left(z^{-1}\right) = D\left(z^{-1}\right)$$

**Assumptions:**

- we know $n$, $m$, and $d$;
- the plant is irreducible.
Indirect adaptive pole placement: big picture

\[ r^*(k) \rightarrow \hat{T}(k, z^{-1}) \rightarrow u(k) \rightarrow y(k) \]

At time \( k \), identify \( \hat{B}(k, z^{-1}) \) and \( \hat{A}(k, z^{-1}) \) (using a suitable PAA); design \( \hat{T}(k, z^{-1}) \) based on methods previously discussed.

Solve Diophantine equation

\[
\hat{A}(k, z^{-1}) \hat{S}(k, z^{-1}) + z^{-d} \hat{B}(k, z^{-1}) \hat{R}(k, z^{-1}) = D(z^{-1})
\]

for \( \hat{S}(k, z^{-1}) \) and \( \hat{R}(k, z^{-1}) \).

Indirect adaptive pole placement: details

Controller order:

\[
\begin{align*}
\hat{A}(k, z^{-1}) &\text{ order: } n \\
\hat{S}(k, z^{-1}) &\text{ order: } d+m-1 \\
\hat{B}(k, z^{-1}) &\text{ order: } d+1 \\
\hat{R}(k, z^{-1}) &\text{ order: } n-1 \\
D(z^{-1}) &\text{ order } \leq n+m+d-1
\end{align*}
\]

Controller parameters:

\[
\begin{align*}
\hat{S}(k, z^{-1}) &= \hat{s}_0(k) + \hat{s}_1(k) z^{-1} + \cdots + \hat{s}_{r-1}(k) z^{-d-m+1} \\
\hat{R}(k, z^{-1}) &= \hat{r}_0(k) + \hat{r}_1(k) z^{-1} + \cdots + \hat{r}_{r-1}(k) z^{-n+1}
\end{align*}
\]

Solvability of the Diophantine equation: \( \hat{A}(k, z^{-1}) \) and \( \hat{B}(k, z^{-1}) \) need to be coprime. If not, use the previous estimation.

Control law:

\[
u(k) = \frac{1}{\hat{S}(k, z^{-1})} \left[ \hat{T}(k, z^{-1}) r^*(k) - \hat{R}(k, z^{-1}) y(k) \right] \]
Indirect adaptive pole placement: extension

Consider the plant $z^{-1} B (z^{-1}) / A (z^{-1})$ with the general feedback design

\[ y^* (k) \xrightarrow{\hat{R}(k,z^{-1})} \frac{z^{-1} B (z^{-1}) A (z^{-1})}{\hat{S}(k,z^{-1})} \xrightarrow{\hat{R}(k,z^{-1})} y(k) \]

Similar as before, but assume we know only the order of the plant: $r = \max(n, m + 1)$.

Pole placement with known plant parameters:

\[ A (z^{-1}) S (z^{-1}) + z^{-1} B (z^{-1}) R (z^{-1}) = D (z^{-1}) \]

▶ Can write $B (z^{-1}) = b_0 + b_1 z^{-1} + \cdots + b_{r-1} z^{-r+1}$ and $A (z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_r z^{-r}$

▶ At time $k$, identify $\hat{B}(k,z^{-1})$ and $\hat{A}(k,z^{-1})$

▶ Solve Diophantine equation

\[ \hat{A}(k,z^{-1}) \hat{S}(k,z^{-1}) + z^{-1} \hat{B}(k,z^{-1}) \hat{R}(k,z^{-1}) = D (z^{-1}) \]

for $\hat{S}(k,z^{-1})$ and $\hat{R}(k,z^{-1})$
Indirect adaptive pole placement: extension

Controller order:

\[
\hat{A}(k, z^{-1}) \hat{S}(k, z^{-1}) + z^{-1} \hat{B}(k, z^{-1}) \hat{R}(k, z^{-1}) = D(z^{-1})
\]

\[
\text{order: } r \quad \text{order: } r-1 \quad \text{order: } r \quad \text{order: } r-1 \quad \text{order: } \leq 2r-1
\]

Controller parameters:

\[
\hat{S}(k, z^{-1}) = \hat{s}_0(k) + \hat{s}_1(k) z^{-1} + \cdots + \hat{s}_{r-1}(k) z^{-r+1}
\]

\[
\hat{R}(k, z^{-1}) = \hat{r}_0(k) + \hat{r}_1(k) z^{-1} + \cdots + \hat{r}_{r-1}(k) z^{-r+1}
\]

Control law:

\[
u(k) = \frac{\hat{R}(k, z^{-1})}{\hat{S}(k, z^{-1})} [y^*(k) - y(k)] \\
= \frac{1}{\hat{s}_0(k)} \left\{-\hat{s}_1(k) u(k-1) - \cdots - \hat{s}_{r-1} u(k-r+1) + \hat{r}_0(k) [y^*(k) - y(k)] + \cdots + \hat{r}_{r-1}(k) [y^*(k-r+1) - y(k-r+1)] \right\}
\]

Summary

1. Big picture

2. Adaptive pole placement
   Cancellable \( B(z^{-1}) \)
   Remark

3. Extension: adaptive pole placement with disturbance cancellation

4. Pole placement with no cancellation of \( B(z^{-1}) \)

5. Indirect adaptive pole placement
We mentioned that direct adaptive control requires no identification of the plant. In direct adaptive pole placement, the closed loop characteristic polynomial is

\[ D(z^{-1}) = D'(z^{-1})B(z^{-1}) \]

which depends on \( B(z^{-1}) \). So the closed-loop design directly depends on the plant zeros. Why is it still direct adaptive control?