Robust and optimal control

Clément Roos – Thierry Miquel – Daniel Alazard

ONERA THE FRENCH AEROSPACE LAB

ENAC

ISAE Institut Supérieur de l'Aéronautique et de l'Espace

Course overview

Objectives

- Learn the basics in optimal control
- Learn the basics in robust control and robustness analysis
- Implement and use robust analysis and control design tools in Matlab/Simulink

Contact

- Clément ROOS
  ONERA The French Aerospace Lab
  clement.roos@onera.fr

Program at a glance

- 35h between 04/12/2023 and 13/02/2024
- 22h of lectures and exercises
- 11h of Matlab workshops
- Evaluation by workshop report and written exams (2h)

Team

• Daniel ALAZARD	daniel.alazard@isae.fr	optimal control
• Thierry MIQUEL	thierry.miquel@enac.fr	optimal control
• Jean-Marc BIANNIC	jean-marc.biannic@onera.fr	robust control
• Clément ROOS	clement.roos@onera.fr	robust control

ONERA THE FRENCH AEROSPACE LAB

ENAC

isae Institut Supérieur de l'Aéronautique et de l'Espace

Course overview

Part 1 – Optimal control

Lectures: 10 hours (04/12, 18/12, 09/01)

- Bellman Principle, Hamilton-Jacobi-Bellman equation, Pontryagin's minimum principle
- LQ, LQG and LQG/LTR control

Workshop: 3 hours (12/12)

- Optimal rendez-vous trajectory

Part 2 – Robust control

Lectures: 12 hours (10/01, 12/01, 16/01, 24/01, 30/01)

- H_∞ and H_2 control
- μ-analysis

Workshops: 8 hours (18/01, 02/02)

- H_∞ control for an unstable aerospace vehicle
- Robustness analysis of a spark ignition engine

Evaluation

- January: Two short intermediate written exams (individually)
- 02/02/2024: Robustness analysis workshop report (in pairs) to be uploaded on LMS
- 13/02/2024: Final written exam (individually)
- Grades will be weighted as follows: final written exam 50%, intermediate written exams 25%, workshop report 25%

Part 1 - Optimal control

Optimal Control & Guidance: from Dynamic Programming to Pontryagin's Minimum Principle

D. Alazard[1]

[1] Institut Supérieur de l'Aréonautique et de l'Espace.

see also: http://www.llibre.fr/michel/copie_cert/DCSD-2009-008-NOT-010-1.0.pdf

Outline

1 Dynamic Programming (D.P.)
- Problem statement
- Resolution with Bellman's principle
- Linear system and Quadratic index (LQ) particular case

2 Pontryagin's Minimum Principle (P.M.P.)
- Problem statement
- From D.P. to P.M.P.: Hamilton-Jacobi-Bellman equation
- Hamilton-Jacobi-Bellman (H.J.B.) equation resolution
- Pontryagin's Minimum Principle (P.M.P.)
- Linear system and Quadratic index (LQ) particular case
- Ex: bang-bang control

Dynamic Programming (D.P.)

Dynamic programming is a method to solve optimization problems, based on the **Bellman's optimality principle**. It consists in simplifying a complicated problem by breaking it down into simpler sub-problems in a recursive manner. If the optimality condition of a problem can be expressed as a function of the optimum of a simpler sub-problem then D.P. can be used.

D.P. can be also used in computer science, mathematics, management science, economics , ... We focus here on the D.P. applied to optimal control of dynamic systems.

P.D.: (simplified) problem statement

Let us consider a non-linear, discrete-time, dynamic system described by a state-space representation:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)\ , \quad \mathbf{x}_k \in \mathbb{R}^n\ , \quad \mathbf{u}_k \in \mathbb{R}^m\ , \quad \mathbf{x}_0 \text{ is known.} \tag{1}$$

Within a given discrete-time horizon N, one wishes to impose the final state $\mathbf{x}_N$. Without loss of generality, one can choose:

$$\mathbf{x}_N = \mathbf{0}\ , \quad (N \text{ is given})\ , \tag{2}$$

while minimizing a performance index :

$$J = \sum_{k=0}^{N-1} r(\mathbf{x}_k, \mathbf{u}_k)\ . \tag{3}$$

Example: $x_{k+1} = x_k + u_k$ (single accumulator) with $J = \sum_{k=0}^{N-1} u_k^2$ (minimal energy control) or with $J = \sum_{k=0}^{N-1} 1 = N$ (minimal time).

Resolution with Bellman's principle

Let us define the **cost-to-go function** R_i at the time i as the performance index value for a trajectory starting at time i and ending at time N and satisfying the constraints (1) and (2):

$$R_i(\mathbf{x}_i, \mathbf{u}_i, \mathbf{u}_{i+1}, \dots, \mathbf{u}_{N-1}) = \sum_{k=i}^{N-1} r(\mathbf{x}_k, \mathbf{u}_k)\ . \tag{4}$$

$$\begin{aligned} R_i(.) &= r(\mathbf{x}_i, \mathbf{u}_i) + R_{i+1}(\mathbf{x}_{i+1}, \mathbf{u}_{i+1}, \dots, \mathbf{u}_{N-1}) \\ &= r(\mathbf{x}_i, \mathbf{u}_i) + R_{i+1}(\mathbf{f}(\mathbf{x}_i, \mathbf{u}_i), \mathbf{u}_{i+1}, \dots, \mathbf{u}_{N-1}) \end{aligned}$$

The optimal cost-to-go function $\widehat{R_i}(\mathbf{x}_i)$ computed on the optimal trajectory starting from $\mathbf{x}_i$ at time i depends only on $\mathbf{x}_i$

$$\widehat{R_i}(\mathbf{x}_i) = \min_{\mathbf{u}_i, \mathbf{u}_{i+1}, \dots, \mathbf{u}_{N-1}} R_i(\mathbf{x}_i, \mathbf{u}_i, \mathbf{u}_{i+1}, \dots, \mathbf{u}_{N-1})$$

Bellman optimality principle

$$\widehat{R_i}(\mathbf{x}_i) = \min_{\mathbf{u}_i} \Big(r(\mathbf{x}_i, \mathbf{u}_i) + \widehat{R}_{i+1}(\mathbf{f}(\mathbf{x}_i, \mathbf{u}_i)) \Big) \tag{5}$$

Eq. (5) is a backward recursive equation to be solved numerically starting at time $N-1$:

- step # 1 compute $\widehat{\mathbf{u}}_{N-1}(\mathbf{x}_{N-1})$ to meet the constraints (1) and (2), i.e.: $\mathbf{x}_N = \mathbf{f}(\mathbf{x}_{N-1}, \mathbf{u}_{N-1}) = \mathbf{0}$,
- step # 2 compute $\widehat{R}_{N-1}(\mathbf{x}_{N-1}) = r(\mathbf{x}_{N-1}, \widehat{\mathbf{u}}_{N-1}(\mathbf{x}_{N-1}))$,
- step # 3 initialize $i = N-1$: $\Rightarrow$ $\mathbf{x}_i = \mathbf{x}_{N-1}$, $\widehat{\mathbf{u}}_i = \widehat{\mathbf{u}}_{N-1}$ and $\widehat{R}_i = \widehat{R}_{N-1}$,
- step # 4 compute: $R_{i-1}(\mathbf{x}_{i-1}, \mathbf{u}_{i-1}) = r(\mathbf{x}_{i-1}, \mathbf{u}_{i-1}) + \widehat{R}_i(\mathbf{f}(\mathbf{x}_{i-1}, \mathbf{u}_{i-1}))$,
- step # 5 optimize: $\widehat{\mathbf{u}}_{i-1}(\mathbf{x}_{i-1}) = \arg\min_{\mathbf{u}_{i-1}} R_{i-1}(\mathbf{x}_{i-1}, \mathbf{u}_{i-1})$ and compute: $\widehat{R}_{i-1}(\mathbf{x}_{i-1}) = R_{i-1}(\mathbf{x}_{i-1}, \widehat{\mathbf{u}}_{i-1}(\mathbf{x}_{i-1}))$,
- step # 6 if $i > 0$, then $i = i-1$ go to step # 3,
- step # 7 from $\mathbf{x}_0$ and $\widehat{\mathbf{u}}_i(\mathbf{x}_i)$, $i = 0, \dots, N-1$ one can compute the optimal state trajectory $\widehat{\mathbf{x}}_i$ by the direct integration of (1).

Remark: the D.P. provides directly a **closed-loop state feedback** :

$$\widehat{\mathbf{u}}_i(\mathbf{x}_i), \quad i = 0, \dots, N-1\ .$$

Example: $x_{k+1} = x_k + u_k$, $J = \sum_{k=0}^{N-1} u_k^2$.

In a more general case: the optimization problem may consider additional constraints to constraints (1) and (2):

- instantaneous constraint: $\boldsymbol{\gamma}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \ \forall \ k$. Ex: saturation of the j-th control signal: $|\mathbf{u}_k(j)| - \bar{\mathbf{u}}(j) \leq 0, \ j = 1, \ldots, m$ or operating condition constraint on the j-th state component: $\mathbf{x}_k(j) - \bar{\mathbf{x}}(j) \leq 0$, $\underline{\mathbf{x}}(j) - \mathbf{x}_k(j) \leq 0, \ \forall \ k$,
- global constraint: $\mathbf{g}(\mathbf{x}_0, \ldots, \mathbf{x}_N, \mathbf{u}_0, \ldots, \mathbf{x}_{N-1}) \leq 0$. Ex. total on-board energy limitation: $\sum_{k=0}^{N-1} \mathbf{u}_k^T \mathbf{u}_k - E_{tot} \leq 0$ or $\sum_{k=1}^{N-1} |\mathbf{u}_k - \mathbf{u}_{k-1}| - E_{tot} \leq 0$,
- constraint on the terminal state: $\mathbf{l}(\mathbf{x}_N) = 0$ (instead of $\mathbf{x}_N = 0$), associated with a penalty on the terminal state in the performance index: $J = \sum_{k=0}^{N-1} r(\mathbf{x}_k, \mathbf{u}_k) + r_N(\mathbf{x}_N)$,
- constraint on the initial state: $\mathbf{h}(\mathbf{x}_0) = 0$.

The P.D. can also solve non-stationary problems: i.e. the functions $\mathbf{f}$, $\boldsymbol{\gamma}$ and r could depends in time k and could change with time k:

$$\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \rightarrow \mathbf{f}_k(\mathbf{x}_k, \mathbf{u}_k, k), \ \boldsymbol{\gamma}(\mathbf{x}_k, \mathbf{u}_k) \rightarrow \boldsymbol{\gamma}_k(\mathbf{x}_k, \mathbf{u}_k, k), \ r(\mathbf{x}_k, \mathbf{u}_k) \rightarrow r_k(\mathbf{x}_k, \mathbf{u}_k, k)$$

Linear system and Quadratic index (LQ) particular case

Same problem but $f(.)$ is Linear and J is Quadratic (in $\mathbf{x}_k$ and $\mathbf{u}_k$):

LQ case

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \quad \mathbf{x}_k \in \mathbb{R}^n, \ \mathbf{u}_k \in \mathbb{R}^m, \ \mathbf{x}_N = 0; \ \mathbf{x}_0, N \text{ are known.} \tag{6}$$

$$J = \sum_{k=0}^{N-1} \mathbf{x}_k^T \mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^T \mathbf{R}\mathbf{u}_k \quad \text{with: } \mathbf{Q} \succeq 0, \ \mathbf{R} \succ 0 \,. \tag{7}$$

In that case, one can find an analytical solution. Indeed:

At time $N-1$: $\mathbf{A}\mathbf{x}_{N-1} + \mathbf{B}\mathbf{u}_{N-1} = 0$

- If $\mathbf{B}$ is left invertible ($\Rightarrow m \geq n$) then:
$$\widehat{\mathbf{u}}_{N-1} = -\mathbf{R}^{-1}\mathbf{B}^T(\mathbf{B}\mathbf{R}^{-1}\mathbf{B})^{-1}\mathbf{A}\mathbf{x}_{N-1} \,.$$
- if $m < n$, then $\mathbf{x}_N = 0$ requires the use of last l commands $\mathbf{u}_{N-l}, \ldots \mathbf{u}_{N-1}$ with l s.t. rank$[\mathbf{A}^{l-1}\mathbf{B} \ldots \mathbf{A}\mathbf{B}\ \mathbf{B}] = n$.

Let us consider the single input case: $m = 1$ **then** $l = n$
Step # 1: at time $N-n$: the state is $\mathbf{x}_{N-n}$. Then

$$\begin{aligned}
\mathbf{x}_{N-n+1} &= \mathbf{A}\mathbf{x}_{N-n} + \mathbf{B}\mathbf{u}_{N-n} \\
\mathbf{x}_{N-n+2} &= \mathbf{A}^2\mathbf{x}_{N-n} + \mathbf{A}\mathbf{B}\mathbf{u}_{N-n} + \mathbf{B}\mathbf{u}_{N-n+1} \\
&= \mathbf{A}^2\mathbf{x}_{N-n} + [\mathbf{A}\mathbf{B}\ \mathbf{B}] \begin{bmatrix} \mathbf{u}_{N-n} \\ \mathbf{u}_{N-n+1} \end{bmatrix} \\
\vdots \quad & \quad \vdots \\
\mathbf{0} = \mathbf{x}_N &= \mathbf{A}^n\mathbf{x}_{N-n} + \underbrace{[\mathbf{A}^{n-1}\mathbf{B}\ \mathbf{A}^{n-2}\mathbf{B} \ldots \mathbf{A}\mathbf{B}\ \mathbf{B}]}_{\mathcal{C}:\ \text{controllability matrix}} \begin{bmatrix} \mathbf{u}_{N-n} \\ \mathbf{u}_{N-n+1} \\ \vdots \\ \mathbf{u}_{N-2} \\ \mathbf{u}_{N-1} \end{bmatrix}
\end{aligned}$$

$\mathcal{C}$ must be invertible (i.e. rank$(\mathcal{C}) = n$) $\Rightarrow$ **the system must be controllable**. Then:

$$\begin{bmatrix} \widehat{\mathbf{u}}_{N-n} \\ \widehat{\mathbf{u}}_{N-n+1} \\ \vdots \\ \widehat{\mathbf{u}}_{N-1} \end{bmatrix} = -\mathcal{C}^{-1}\mathbf{A}^n\mathbf{x}_{N-n} \,.$$

Step # 2: let us compute the optimal cost-to-go function: $\widehat{R}_{N-n}$

$$
\begin{aligned}
&= \sum_{k=N-n}^{N-1} \mathbf{x}_k^T \mathbf{Q}\mathbf{x}_k + \hat{\mathbf{u}}_k^T \mathbf{R}\hat{\mathbf{u}}_k \left(= \sum_{k=N-n}^{N-1} \mathbf{x}_k^T \mathbf{Q}\mathbf{x}_k + \mathbf{R}\hat{\mathbf{u}}_k^2) \right) \\
&= (\star)^T \underbrace{\begin{bmatrix} \mathbf{Q} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{Q} \end{bmatrix}}_{\mathcal{Q}:\, n^2\times n^2} \left(\underbrace{\begin{bmatrix} \mathbf{1}_n \\ \mathbf{A} \\ \vdots \\ \mathbf{A}^{n-1} \end{bmatrix}}_{\mathcal{I}_{\mathbf{A}}^n:\, n^2\times n} \mathbf{x}_{N-n} + \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{B} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{AB} & \mathbf{B} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{A}^{n-2}\mathbf{B} & \cdots & \mathbf{AB} & \mathbf{B} & \mathbf{0} \end{bmatrix}}_{\mathcal{T}:\, n^2\times n} \begin{bmatrix} \hat{\mathbf{u}}_{N-n} \\ \vdots \\ \vdots \\ \hat{\mathbf{u}}_{N-1} \end{bmatrix} \right) \\
&\quad + \begin{bmatrix} \hat{\mathbf{u}}_{N-n} \\ \vdots \\ \hat{\mathbf{u}}_{N-1} \end{bmatrix}^T \mathbf{R} \begin{bmatrix} \hat{\mathbf{u}}_{N-n} \\ \vdots \\ \hat{\mathbf{u}}_{N-1} \end{bmatrix} \\
&= \mathbf{x}_{N-n}^T \underbrace{\left(\left(\mathcal{I}_{\mathbf{A}}^n - \mathcal{T}\mathcal{C}^{-1}\mathbf{A}^n\right)^T \mathcal{Q} \left(\mathcal{I}_{\mathbf{A}}^n - \mathcal{T}\mathcal{C}^{-1}\mathbf{A}^n\right) + {\mathbf{A}^n}^T \mathcal{C}^{-T}\mathbf{R}\mathcal{C}^{-1}\mathbf{A}^n \right)}_{\mathbf{P}_{N-n}} \mathbf{x}_{N-n} \qquad (8)
\end{aligned}
$$

$\mathbf{P}_{N-n}$ is a sum of quadratic terms: $\mathbf{P}_{N-n} \succeq 0$ and is symmetric.

$$\widehat{R}_{N-n}(\mathbf{x}_{N-n}) = \mathbf{x}_{N-n}^T \mathbf{P}_{N-n} \mathbf{x}_{N-n} \ .$$

Step # 3: initialization: $i = N - n$: $\mathbf{P}_i = \mathbf{P}_{N-n}$, $\widehat{R}_i(\mathbf{x}_i) = \mathbf{x}_i^T \mathbf{P}_i \mathbf{x}_i$.
Step # 4 at time $i - 1 = j$: Compute $R_j(\mathbf{x}_j, \mathbf{u}_j)$:

$$
\begin{aligned}
R_j(\mathbf{x}_j, \mathbf{u}_j) &= \mathbf{x}_j^T \mathbf{Q}\mathbf{x}_j + \mathbf{u}_j^T \mathbf{R}\mathbf{u}_j \\
&\quad + (\mathbf{A}\mathbf{x}_j + \mathbf{B}\mathbf{u}_j)^T \mathbf{P}_{j+1} (\mathbf{A}\mathbf{x}_j + \mathbf{B}\mathbf{u}_j) \\
&= \mathbf{x}_j^T \left(\mathbf{Q} + \mathbf{A}^T\mathbf{P}_{j+1}\mathbf{A}\right) \mathbf{x}_j + 2\mathbf{x}_j^T \mathbf{A}^T \mathbf{P}_{j+1} \mathbf{B}\mathbf{u}_j \\
&\quad + \mathbf{u}_j^T \left(\mathbf{R} + \mathbf{B}^T\mathbf{P}_{j+1}\mathbf{B}\right) \mathbf{u}_j \ .
\end{aligned}
$$

Step # 5: Optimize $R_j(\mathbf{x}_j, \mathbf{u}_j)$ w.r.t. $\mathbf{u}_j$

$$\frac{\partial R_j}{\partial \mathbf{u}_j}(\mathbf{x}_j, \hat{\mathbf{u}}_j) = 0 = 2\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{A}\mathbf{x}_j + 2\left(\mathbf{R} + \mathbf{B}^T\mathbf{P}_{j+1}\mathbf{B}\right)\hat{\mathbf{u}}_j \ .$$

First order optimality condition is sufficient since J is quadratic. Thus:

$$\widehat{\mathbf{u}}_j = -\underbrace{\left(\mathbf{R}+\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{A}}_{\mathbf{K}_j}\,\mathbf{x}_j = -\mathbf{K}_j\mathbf{x}_j\ .$$

The optimal control is a **linear (non-stationary) state feedback.**
Then, compute:

$$\begin{aligned}\widehat{R}_j(\mathbf{x}_j) &= \mathbf{x}_j^T\underbrace{\left(\mathbf{Q}+\mathbf{A}^T\mathbf{P}_{j+1}\mathbf{A}-\mathbf{A}^T\mathbf{P}_{j+1}\mathbf{B}\left(\mathbf{R}+\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{A}\right)}_{\mathbf{P}_j}\mathbf{x}_j \\ &= \mathbf{x}_j^T\mathbf{P}_j\mathbf{x}_j\ .\end{aligned}$$

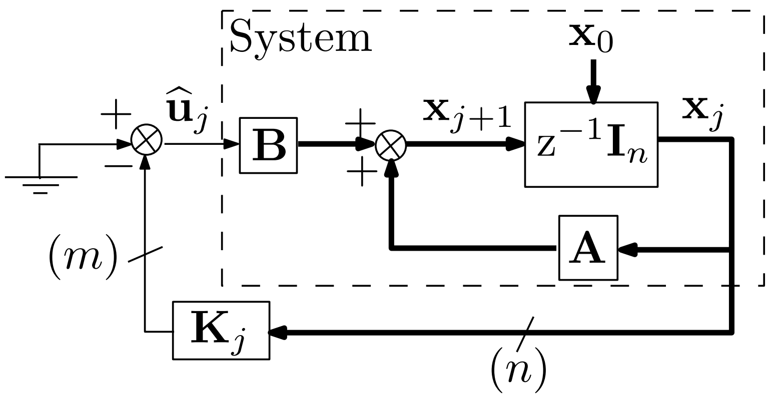
Step # 6: for each time $j = N-n-1,\dots,0$, one can compute all the $\mathbf{P}_j$, $\mathbf{K}_j$ and $\widehat{\mathbf{u}}_j$ by solving the **recurrent Riccati equation**:

$$\mathbf{P}_j = \mathbf{Q}+\mathbf{A}^T\mathbf{P}_{j+1}\mathbf{A}-\mathbf{A}^T\mathbf{P}_{j+1}\mathbf{B}\left(\mathbf{R}+\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{A} \qquad (9)$$

to be initialized with $j+1 = N-n$ and $\mathbf{P}_{N-n}$ given by eq. (8). Then:

$$\mathbf{K}_j = \left(\mathbf{R}+\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}_{j+1}\mathbf{A} \quad \text{and: } \widehat{\mathbf{u}}_j = -\mathbf{K}_j\mathbf{x}_j\ . \qquad (10)$$

Step # 7: The $\mathbf{P}_j$ and the $\mathbf{K}_j$ do not depend on the current state $\mathbf{x}_j$ and can be computed off-line from eqs. (9) and (10). Step # 7 can be omitted for a closed loop implementation of the optimal control law according to the sketch depicted in the following Figure which assumed that a state measurement system (navigation system) provides $\mathbf{x}_j$.



For a **blind** open-loop implementation, the off-line computation of $\widehat{\mathbf{u}}_j = -\mathbf{K}_j\widehat{\mathbf{x}}_j$ requires the integration of (1) from $\widehat{\mathbf{x}}_0 = \mathbf{x}_0$ (known) to find the predicted optimal trajectory $\widehat{\mathbf{x}}_{j+1} = (\mathbf{A}-\mathbf{B}\mathbf{K}_j)\widehat{\mathbf{x}}_j$, $j = 0,\dots N-1$ (you can check that $\widehat{\mathbf{x}}_N = \mathbf{0}$). Then, you can cross your fingers and hope that an external disturbance will not perturb too much the real state trajectory.

Additional remarks:

- the optimal index performance is: $\widehat{J} = \widehat{R}_0 = \mathbf{x}_0^T \mathbf{P}_0 \mathbf{x}_0$.
- in the case of an infinite time horizon ($N \to \infty$), the optimal cost-to-go function does not depend on the time j and depends only the state: $\widehat{R}_j(\mathbf{x}_j) = \widehat{R}(\mathbf{x}_j) = \mathbf{x}_j^T \mathbf{P} \mathbf{x}_j$. Thus, the solution of the recurrent RICCATI equation becomes constant: $\mathbf{P}_j = \mathbf{P}_{j+1} = \mathbf{P}$. $\mathbf{P}$ is **unique positive** solution of the **discrete-time algebraic Riccati equation** (see also: function `dare` in MATLAB):

$$\mathbf{P} = \mathbf{Q} + \mathbf{A}^T\mathbf{P}\mathbf{A} - \mathbf{A}^T\mathbf{P}\mathbf{B}\left(\mathbf{R} + \mathbf{B}^T\mathbf{P}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}\mathbf{A}$$

and the optimal control law is a linear stationary state feedback: $\widehat{\mathbf{u}}_j = -\mathbf{K}\mathbf{x}_j$, $\forall j$ with:

$$\mathbf{K} = \left(\mathbf{R} + \mathbf{B}^T\mathbf{P}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}\mathbf{A} \,.$$

Example: $x_{k+1} = x_k + u_k$, $J = \sum_{k=0}^{N-1} u_k^2$. Check your previous results using recurrent RICCATI equation (9).

P.M.P: (simplified) problem statement

Let us consider a non-linear, continuous-time, dynamic system described by a state-space representation:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \,, \quad \mathbf{x} = \mathbf{x}(t) \in \mathbb{R}^n \,, \quad \mathbf{u} = \mathbf{u}(t) \in \mathbb{R}^m \,, \quad \mathbf{x}(0) \text{ is known.} \tag{11}$$

Within a given time horizon t_f, one wishes to impose the final state $\mathbf{x}(t_f)$. Without loss of generality, one can choose:

$$\mathbf{x}(t_f) = \mathbf{0} \,, \quad (t_f \text{ is given}) \,, \tag{12}$$

while minimizing a performance index :

$$J = \int_0^{t_f} r\left(\mathbf{x}(t), \mathbf{u}(t)\right) dt \,. \tag{13}$$

From D.P. to P.M.P.: Hamilton-Jacobi-Bellman equation

The BELLMAN optimality principle is applied to the discrete-time system obtained using a first order expansion:

$$\mathbf{x}(t+dt) = \mathbf{x}(t) + \mathbf{f}\left(\mathbf{x},\mathbf{u}\right)\, dt\ .$$

The **optimal cost-to-go** function at time t is defined by:

$$\widehat{R}\left(\mathbf{x}(t),t\right) = \min_{\mathbf{u}([t,t_f])} \left(\int_t^{t_f} r\left(\mathbf{x}(t),\mathbf{u}(t)\right) dt \right)\ .$$

Thanks to the BELLMAN principle: if $\widehat{R}(\mathbf{x}(t+dt),t+dt)$ is known at the time $t+dt$, then:

$$\widehat{R}(\mathbf{x}(t),t) = \min_{\mathbf{u}(t)} \left(r(\mathbf{x}(t),\mathbf{u}(t))dt + \widehat{R}(\mathbf{x}(t+dt),t+dt) \right)$$

Dynamic Programming (D.P.) Pontryagin's Minimum Principle (P.M.P.)

Assuming $\widehat{R}(\mathbf{x}(t),t)$ is continuous and differentiable, a first order expansion provides:

$$\begin{aligned}
\widehat{R}(\mathbf{x}(t+dt),t+dt) &= \widehat{R}(\mathbf{x}(t),t) + \dot{\widehat{R}}(\mathbf{x}(t),t)dt \\
&= \widehat{R}(\mathbf{x}(t),t) + \frac{\partial \widehat{R}}{\partial \mathbf{x}}^T \dot{\mathbf{x}}dt + \frac{\partial \widehat{R}}{\partial t} dt
\end{aligned}$$

thus:

$$\begin{aligned}
\widehat{R}(\mathbf{x}(t),t) &= \min_{\mathbf{u}(t)} \left(r(\mathbf{x},\mathbf{u})dt + \widehat{R}(\mathbf{x}(t),t) + \frac{\partial \widehat{R}}{\partial \mathbf{x}}^T \mathbf{f}(\mathbf{x},\mathbf{u})dt + \frac{\partial \widehat{R}}{\partial t} dt \right) \\
&= \widehat{R}(\mathbf{x}(t),t) + \frac{\partial \widehat{R}}{\partial t} dt + \min_{\mathbf{u}(t)} \left(r(\mathbf{x},\mathbf{u})dt + \frac{\partial \widehat{R}}{\partial \mathbf{x}}^T \mathbf{f}(\mathbf{x},\mathbf{u})dt \right)
\end{aligned}$$

Hamilton-Jacobi-Bellman equation:

$$\frac{\partial \widehat{R}}{\partial t} = -\min_{\mathbf{u}(t)} \left(r(\mathbf{x},\mathbf{u}) + \frac{\partial \widehat{R}}{\partial \mathbf{x}}^T \mathbf{f}(\mathbf{x},\mathbf{u}) \right) \qquad (14)$$

Hamilton-Jacobi-Bellman (H.J.B.) equation resolution

H.J.B. equation is "touchy": a P.D.E involving the function $\min$. To solve it, one can introduce additional variables, called **co-states** in the co-state vector:

$$\psi(t) = \frac{\partial \widehat{R}}{\partial \mathbf{x}}(\mathbf{x}(t), t) \quad \text{also noted: } \psi = \frac{\partial \widehat{R}}{\partial \mathbf{x}}$$

and the Hamiltonian:

$$\mathcal{H} = r(\mathbf{x}, \mathbf{u}) + \psi^T \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad \text{also noted: } \mathcal{H} = r + \psi^T \mathbf{f}$$

to transform the H.J.B equation into a first order differential equation:

$$\frac{\partial \widehat{R}}{\partial t} = -\min_{\mathbf{u}(t)}(\mathcal{H}) \qquad \Rightarrow \widehat{\mathbf{u}}(t) = \arg\min_{\mathbf{u}(t)}(\mathcal{H}) \, .$$

The optimal control minimizes the Hamiltonian.

$\widehat{\mathbf{u}}(t)$ depends on $\widehat{x}(t)$ (the state of the system: OK) but also depends on $\psi(t)$ (?). $\psi(t)$ can be characterized by its evolution. Indeed:

$$\dot{\psi}(t) = \frac{d}{dt}\frac{\partial \widehat{R}}{\partial \mathbf{x}}(\mathbf{x}(t), t) = \frac{\partial^2 \widehat{R}}{\partial \mathbf{x}^2}\mathbf{f} + \frac{\partial}{\partial t}\frac{\partial \widehat{R}}{\partial \mathbf{x}} = \frac{\partial^2 \widehat{R}}{\partial \mathbf{x}^2}\mathbf{f} + \frac{\partial}{\partial \mathbf{x}}\frac{\partial \widehat{R}}{\partial t} \, .$$

Let us consider H.J.B eq. (14) at the optimum:
$\frac{\partial \widehat{R}}{\partial t} = -\left(r(\mathbf{x}, \widehat{\mathbf{u}}) + \frac{\partial \widehat{R}}{\partial \mathbf{x}}^T \mathbf{f}(\mathbf{x}, \widehat{\mathbf{u}}) \right)$ and its gradient w.r.t $\mathbf{x}$:

$$\frac{\partial}{\partial \mathbf{x}}\frac{\partial \widehat{R}}{\partial t} = -\frac{\partial r}{\partial \mathbf{x}} - \frac{\partial^2 \widehat{R}}{\partial \mathbf{x}^2}\mathbf{f} - \frac{\partial \mathbf{f}^T}{\partial \mathbf{x}}\frac{\partial \widehat{R}}{\partial \mathbf{x}}$$

$$\text{Thus: } \dot{\psi} = -\frac{\partial r}{\partial \mathbf{x}} - \frac{\partial \mathbf{f}^T}{\partial \mathbf{x}}\frac{\partial \widehat{R}}{\partial \mathbf{x}} = -\frac{\partial r}{\partial \mathbf{x}} - \frac{\partial \mathbf{f}^T}{\partial \mathbf{x}}\psi$$

Co-state dynamics:

$$\dot{\psi} = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}} \, . \tag{15}$$

Pontryagin's Minimum Principle (P.M.P.)

P.M.P. (summary of the simplified case)

$$\begin{aligned}
\text{model} &: \quad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u}) \quad \mathbf{x}(0) \text{ is known} && (16)\\
\text{constraint} &: \quad \mathbf{x}(t_f) = \mathbf{0}\ , \quad (t_f \text{ is given}) && (17)\\
\text{performance index} &: \quad J = \int_0^{t_f} r(\mathbf{x},\mathbf{u})\,dt && (18)\\
\text{Hamiltonian} &: \quad \mathcal{H} = r + \boldsymbol{\psi}^T\mathbf{f} && (19)\\
\text{Optimal control} &: \quad \widehat{\mathbf{u}} = \arg\min_{\mathbf{u}(t)}(\mathcal{H}) = \widehat{\mathbf{u}}(\mathbf{x},\boldsymbol{\psi}) && (20)\\
\text{co-state dynamics} &: \quad \dot{\boldsymbol{\psi}} = -\frac{\partial\mathcal{H}}{\partial\mathbf{x}} = \mathbf{f}_{co}(\mathbf{x},\mathbf{u},\boldsymbol{\psi}) && (21)
\end{aligned}$$

Substituting $\mathbf{u}$ by the optimal value $\widehat{\mathbf{u}}$ (eq. (20)) in (16) and (21), one obtain a **two point boundary-value problem** with $2n$ variables to be integrated tacking into account initial conditions $\mathbf{x}(0)$ and terminal conditions $\mathbf{x}(t_f)$.

Once one has integrated the two point boundary-value problem, one can expressed $\boldsymbol{\psi}(0)$ as a function of $\mathbf{x}(0)$ and t_f and the initial command:

$$\widehat{\mathbf{u}}(0) = \widehat{\mathbf{u}}_0(\mathbf{x}(0), t_f)\ .$$

Then one can directly expressed the closed-loop optimal control law assuming that, at current time t ($\in [0, t_f]$), a measurement system (navigation filter) provides $\mathbf{x}(t)$. The calculus of the current optimal control $\widehat{\mathbf{u}}(t)$ is the same problem than the previous one, just changing $\mathbf{x}(0)$ by $\mathbf{x}(t)$ and t_f by $t_f - t$. Thus:

$$\widehat{\mathbf{u}}(t) = \widehat{\mathbf{u}}_0(\mathbf{x}(t), t_f - t)\ .$$

In a more general case: the P.M.P. can also consider additional constraints and the general problem reads:

P.M.P. (general case)

$$\begin{aligned}
\text{model} \quad &: \quad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u},t)\\
\text{inital constraints} \quad &: \quad \mathbf{h}(\mathbf{x}(0)) = \mathbf{0}\\
\text{final constraints} \quad &: \quad \mathbf{l}(\mathbf{x}(t_f)) = \mathbf{0}\\
\text{instant. constraints} \quad &: \quad \boldsymbol{\gamma}(\mathbf{x},\mathbf{u},t) \le 0\\
\text{global constraints} \quad &: \quad \int_0^{t_f} \mathbf{g}(\mathbf{x},\mathbf{u},t)dt \le E\\
\text{performance index} \quad &: \quad J = r_0(\mathbf{x}(0)) + \int_0^{t_f} r\,(\mathbf{x},\mathbf{u},t)\,dt + r_f(\mathbf{x}(t_f))
\end{aligned}$$

For the general solution, the reader is advise to refer to (page 70): http://michel.llibre.pagesperso-orange.fr/docs/CmdeOpt_llibre.pdf.
Home work: bang-bang control with model $I\ddot{\theta} = u$, $|u| < \bar{u}$, $J = \int_0^{t_f} dt = t_f$, $\theta(t_f) = 0$ and $\dot{\theta}(t_f) = 0$.

Linear system and Quadratic index (LQ) particular case

In that case, the model is linear:

$$\mathbf{f}(\mathbf{x},\mathbf{u},t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)\ .$$

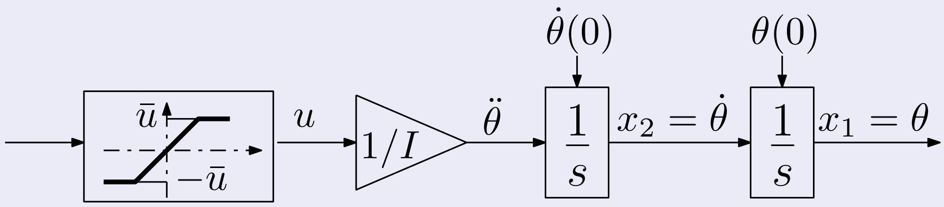
and the performance index is quadratic:

$$r\,(\mathbf{x},\mathbf{u},t) = \frac{1}{2}\left(\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)\right) \quad \text{with: } \mathbf{Q} \succeq 0,\ \mathbf{R} \succ 0\ .$$

This case is detailed in https://personnel.isae-supaero.fr/daniel-alazard/goodies-demos-210/pontryagin-s-minimum-principle.html and will be directly applied to the orbital rendez-vous of two spacecraft during the MATLAB labwork.

Example: bang-bang control

Let us consider the control of the attitude $\theta\,(rd)$ of an inertia $I\,(Kg\,m^2)$ with a torque actuator $u\,(Nm)$ under saturation constraints $|u| \leq \bar{u}$.



Model: $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \quad \rightarrow \quad \left\{ \begin{array}{lcl} \dot{\theta}(t) & = & \dot{\theta}(t) \\ \ddot{\theta}(t) & = & \frac{1}{I}u(t) \end{array} \right.$

Initial constraints: $\mathbf{h}(\mathbf{x}(0)) = \mathbf{0} \quad \rightarrow \quad \left\{ \begin{array}{lcl} \theta(0) - \theta_0 & = & 0 \\ \dot{\theta}(0) - \dot{\theta}_0 & = & 0 \end{array} \right.$

Final constraints: $\mathbf{l}(\mathbf{x}(t_f)) = \mathbf{0} \quad \rightarrow \quad \left\{ \begin{array}{lcl} \theta(t_f) & = & 0 \\ \dot{\theta}(t_f) & = & 0 \end{array} \right.$

Instant. constraints: $\boldsymbol{\gamma}(\mathbf{x}, \mathbf{u}, t) \leq 0 \quad \rightarrow \quad \left\{ \begin{array}{rcl} u(t) - \bar{u} & \leq & 0 \\ -u(t) - \bar{u} & \leq & 0 \end{array} \right.$

Perf index: $J = \int_0^{t_f} r(\mathbf{x}, \mathbf{u}, t) = t_f \quad \rightarrow \quad r(.) = 1$ **(minimal time)**

P.M.P. with instant. constraints (no global constraints)

$$\text{Hamiltonian} \quad : \quad \mathcal{H} = \underbrace{r + \boldsymbol{\mu}^T \boldsymbol{\gamma}}_{\text{Lagrangian: } \mathcal{L}} + \boldsymbol{\psi}^T \mathbf{f} \qquad (22)$$

$\boldsymbol{\mu}$ is the vector of the Kuhn-Tucker parameters associated to the inequality constraints $\boldsymbol{\gamma}$ s.t. $\widehat{\mu}_i \gamma_i(\widehat{\mathbf{x}}, \widehat{\mathbf{u}}, t) = 0$ and $\widehat{\mu}_i \geq 0, \quad \forall\, i$.

$$\text{Optimality conditions} \quad : \quad \{\widehat{\mathbf{u}}, \widehat{\boldsymbol{\mu}}\} = \arg \min_{\substack{\mathbf{u}, \boldsymbol{\mu} \,/ \\ \mu_i \geq 0,\ \forall\, i}} (\mathcal{H}) \qquad (23)$$

$$\text{co-state dynamics} \quad : \quad \dot{\boldsymbol{\psi}} = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}} \qquad (24)$$

Example: bang-bang control

$$\mathcal{H} = 1 + \mu_1(u - \bar{u}) + \mu_2(-u - \bar{u}) + \psi_1 \dot{\theta} + \frac{\psi_2}{I} u$$

- $\frac{\partial \mathcal{H}}{\partial u} = 0 \quad \Rightarrow \quad \frac{\psi_2}{I} + \mu_1 - \mu_2 = 0$
- $\frac{\partial \mathcal{H}}{\partial \mu_1} = 0 \quad \Rightarrow \quad u = \bar{u}$ and $\mu_1 > 0$ otherwise: $u < \bar{u}$ and $\mu_1 = 0$
- $\frac{\partial \mathcal{H}}{\partial \mu_2} = 0 \quad \Rightarrow \quad u = -\bar{u}$ and $\mu_2 > 0$ otherwise: $u > -\bar{u}$ and $\mu_2 = 0$

$\widehat{u}$, $\widehat{\psi}_1$, $\widehat{\psi}_2$ minimize $\mathcal{H}$, thus:

- if $\psi_2 > 0$ then: $\widehat{\mu}_1 = 0$, $\widehat{\mu}_2 = \frac{\psi_2}{I}$ and $\widehat{u} = -\bar{u}$,
- if $\psi_2 < 0$ then: $\widehat{\mu}_2 = 0$, $\widehat{\mu}_1 = -\frac{\psi_2}{I}$ and $\widehat{u} = \bar{u}$,

Optimal control: it always saturates the actuator

$$\widehat{u}(t) = -\bar{u}\ \mathrm{sign}(\psi_2(t))$$

Dynamic Programming (D.P.) Pontryagin's Minimum Principle (P.M.P.)

Example: bang-bang control

Co-state dynamics:

- $\dot{\psi}_1 = -\frac{\partial \mathcal{H}}{\partial \theta} = 0 \quad \Rightarrow \quad \psi_1(t) = cte = \psi_{10}$,
- $\dot{\psi}_2 = -\frac{\partial \mathcal{H}}{\partial \dot{\theta}} = -\psi_1 \quad \Rightarrow \quad \psi_2(t) = -\psi_{10}\, t + \psi_{20}$.

$\psi_2(t)$ is linear in t and thus $\psi_2(t)$ changes its sign only one time at $t_c = \psi_{20}/\psi_{10} \Rightarrow$ **one commutation** between $\pm\bar{u}$ and $\mp\bar{u}$.
Let θ_c and $\dot{\theta}_c$ be the state at $t = t_c$, then the integration of the state equation between t_c and t_f is:

$$\left\{ \begin{array}{lcl} \theta(t_f) = 0 & = & \theta_c + \dot{\theta}_c(t_f - t_c) - \mathrm{sign}(\psi_2)\frac{\bar{u}}{2I}(t_f - t_c)^2 \\ \dot{\theta}(t_f) = 0 & = & \dot{\theta}_c - \mathrm{sign}(\psi_2)\frac{\bar{u}}{I}(t_f - t_c) \end{array} \right.$$

$$t_f - t_c = \frac{\dot{\theta}_c I}{\bar{u}\ \mathrm{sign}(\psi_2)} \geq 0 \Rightarrow \mathrm{sign}(\psi_2) = \mathrm{sign}(\dot{\theta}_c) \Rightarrow t_f - t_c = \frac{|\dot{\theta}_c| I}{\bar{u}}$$

Commutation trajectory

$$\theta_c + \frac{I}{2\bar{u}}\dot{\theta}_c|\dot{\theta}_c| = 0\ . \qquad (25)$$

Example: bang-bang control

The integration of the state equation between $t = 0$ and t_c is:

$$\left\{\begin{array}{lcl}\theta_c & = & \theta_0 + \dot{\theta}_0 t_c + \text{sign}(\dot{\theta}_c)\frac{\bar{u}}{2I}t_c^2 \\ \dot{\theta}_c & = & \dot{\theta}_0 + \text{sign}(\dot{\theta}_c)\frac{\bar{u}}{I}t_c \quad \Rightarrow \quad t_c = (\dot{\theta}_c - \dot{\theta}_0)\frac{I}{\bar{u}}\text{sign}(\dot{\theta}_c) \geq 0\,.\end{array}\right.$$

and with (25): $\dot{\theta}_c^2 = \frac{\dot{\theta}_0^2}{2} - \text{sign}(\dot{\theta}_c)\frac{\bar{u}}{I}\theta_0 \ \geq 0.$

If the initial state is such that $\theta_0 + \frac{I}{2\bar{u}}\dot{\theta}_0|\dot{\theta}_0| > 0$ then:

- $u([0,\ t_c[) = -\bar{u}$ and $u([t_c,\ t_f[) = \bar{u}$,
- $\dot{\theta}_c < 0$ and $\dot{\theta}_c = -\sqrt{\frac{\dot{\theta}_0^2}{2} + \frac{\bar{u}}{I}\theta_0}$
- $t_c = \left(\dot{\theta}_0 + \sqrt{\frac{\dot{\theta}_0^2}{2} + \frac{\bar{u}}{I}\theta_0}\right)\frac{I}{\bar{u}}$
- $t_f = \left(\dot{\theta}_0 + 2\sqrt{\frac{\dot{\theta}_0^2}{2} + \frac{\bar{u}}{I}\theta_0}\right)\frac{I}{\bar{u}}$

Example: bang-bang control

The integration of the state equation between $t = 0$ and t_c is:

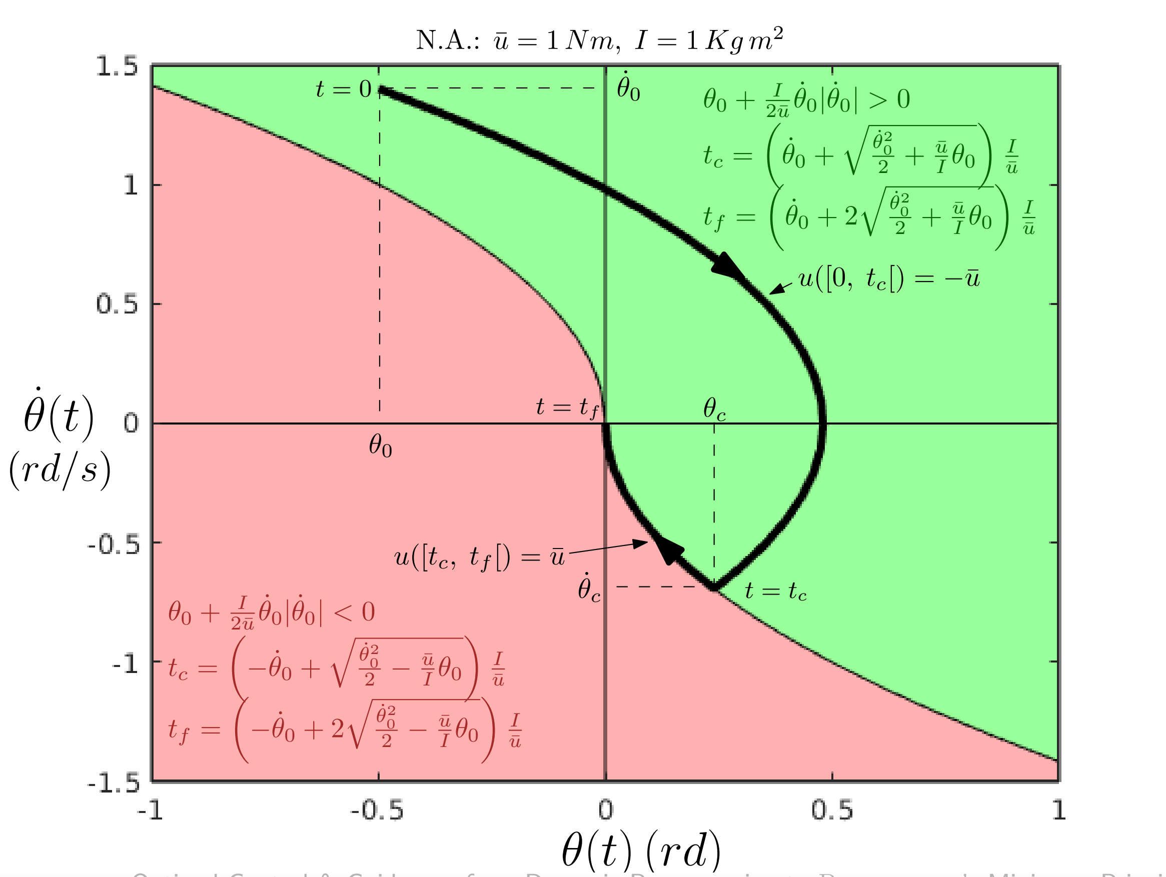
$$\left\{\begin{array}{lcl}\theta_c & = & \theta_0 + \dot{\theta}_0 t_c + \text{sign}(\dot{\theta}_c)\frac{\bar{u}}{2I}t_c^2 \\ \dot{\theta}_c & = & \dot{\theta}_0 + \text{sign}(\dot{\theta}_c)\frac{\bar{u}}{I}t_c \quad \Rightarrow \quad t_c = (\dot{\theta}_c - \dot{\theta}_0)\frac{I}{\bar{u}}\text{sign}(\dot{\theta}_c) \geq 0\,.\end{array}\right.$$

and with (25): $\dot{\theta}_c^2 = \frac{\dot{\theta}_0^2}{2} - \text{sign}(\dot{\theta}_c)\frac{\bar{u}}{I}\theta_0 \ \geq 0.$

If the initial state is such that $\theta_0 + \frac{I}{2\bar{u}}\dot{\theta}_0|\dot{\theta}_0| < 0$ then:

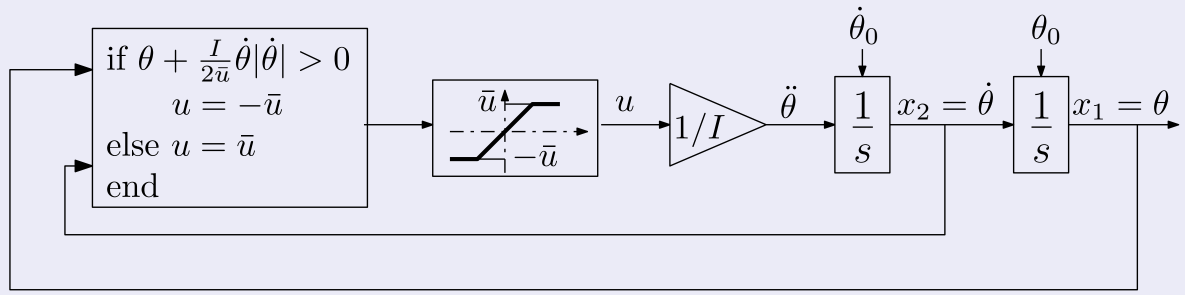
- $u([0,\ t_c[) = \bar{u}$ and $u([t_c,\ t_f[) = -\bar{u}$,
- $\dot{\theta}_c > 0$ and $\dot{\theta}_c = \sqrt{\frac{\dot{\theta}_0^2}{2} - \frac{\bar{u}}{I}\theta_0}$
- $t_c = \left(-\dot{\theta}_0 + \sqrt{\frac{\dot{\theta}_0^2}{2} - \frac{\bar{u}}{I}\theta_0}\right)\frac{I}{\bar{u}}$
- $t_f = \left(-\dot{\theta}_0 + 2\sqrt{\frac{\dot{\theta}_0^2}{2} - \frac{\bar{u}}{I}\theta_0}\right)\frac{I}{\bar{u}}$

Example: bang-bang control



Example: bang-bang control

Closed-loop implementation



⚠: risk of a limit cycle around 0!!

Optimal control:

Linear system, quadratic performance index, fixed horizon and fixed final state

Contents

- Problem:
- Solution using Pontryagin's minimum principle :
- Exercises

Problem:

Let us consider the linear system:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad \mathbf{x} \in \mathbb{R}^n;\ \mathbf{u} \in \mathbb{R}^m \ (1)$$

From a given initial state $\mathbf{x}_0 = \mathbf{x}(0)$, the objective is to bring back the state to 0 within a given time horizon t_f ($\mathbf{x}(t_f) = 0$) while minimizing the quadratic performance index:

$$J = \frac{1}{2}\int_0^{t_f}(\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t))dt$$

where $\mathbf{Q}$ and $\mathbf{R}$ are given weighting matrices with $\mathbf{Q} \geq 0$ and $\mathbf{R} > 0$.

Solution using Pontryagin's minimum principle :

- **The Hamiltonian reads** :

$$\mathcal{H} = \frac{1}{2}(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}) + \mathbf{\Psi}^T(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u})$$

where $\mathbf{\Psi} \in \mathbb{R}^n$ is the costate vector.

- **the optimal control minimizes** $\mathcal{H}$ $\forall t$:

$$\frac{\partial \mathcal{H}}{\partial \mathbf{u}}_{|\mathbf{u}=\hat{\mathbf{u}}} = 0 = \mathbf{R}\hat{\mathbf{u}} + \mathbf{B}^T\mathbf{\Psi} \ \Rightarrow\ \hat{\mathbf{u}} = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{\Psi} \ (2)$$

- **Costate dynamics** :

$$\dot{\mathbf{\Psi}} = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}} \ \Rightarrow\ \dot{\mathbf{\Psi}} = -\mathbf{Q}\mathbf{x} - \mathbf{A}^T\mathbf{\Psi} \ (3)$$

- **State-costate dynamics** : (1), (2) and (3) leads to:

$$\left\{\begin{array}{ccccc} \dot{\mathbf{x}} & = & \mathbf{A}\mathbf{x} & - & \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{\Psi} \\ \dot{\mathbf{\Psi}} & = & -\mathbf{Q}\mathbf{x} & - & \mathbf{A}^T\mathbf{\Psi} \end{array}\right. \Rightarrow \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{\Psi}} \end{bmatrix} = \mathbf{H}\begin{bmatrix} \mathbf{x} \\ \mathbf{\Psi} \end{bmatrix} (4)$$

with

$$\boxed{\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix}}.$$

$\mathbf{H}$ is the $2n \times 2n$ Hamiltonian matrix associated to such a control problem. (4) can be intregrated taken into account boundary conditions on the state-costate augmented vector $[\mathbf{x}^T \quad \mathbf{\Psi}^T]^T$:

- initial conditions on $\mathbf{x}$: $\mathbf{x}(0) = \mathbf{x}_0$ (5),
- terminal conditions on $\mathbf{x}$: $\mathbf{x}(t_f) = 0$ (6).

The set of equations (4), (5) and (6) is also called a **two point boundary-value problem**.

- **Integration of the two point boundary-value problem** :

$$\left[\begin{array}{c}\mathbf{x}(t_f)=0\\ \mathbf{\Psi}(t_f)\end{array}\right] = e^{\mathbf{H}t_f}\left[\begin{array}{c}\mathbf{x}(0)=\mathbf{x}_0\\ \mathbf{\Psi}(0)\end{array}\right] = \left[\begin{array}{cc}e_{11}^{\mathbf{H}t_f} & e_{12}^{\mathbf{H}t_f}\\ e_{21}^{\mathbf{H}t_f} & e_{22}^{\mathbf{H}t_f}\end{array}\right]\left[\begin{array}{c}\mathbf{x}(0)=\mathbf{x}_0\\ \mathbf{\Psi}(0)\end{array}\right]$$

where $e_{ij}^{\mathbf{H}t_f}, i,j=1,2$ are the 4 $n \times n$ submatrices partionning $e^{\mathbf{H}t_f}$ (WARNING !!: $e_{ij}^{\mathbf{H}t_f} \neq e^{\mathbf{H}_{ij}t_f}$).

Then one can easily derive the initial value of the costate:

$$\mathbf{\Psi}(0) = -\left[e_{12}^{\mathbf{H}t_f}\right]^{-1} e_{11}^{\mathbf{H}t_f}\,\mathbf{x}_0 = \mathbf{P}(0)\,\mathbf{x}_0\ .$$

where $\mathbf{P}(0) = -\left[e_{12}^{\mathbf{H}t_f}\right]^{-1} e_{11}^{\mathbf{H}t_f}$ depends only on the problem data: $\mathbf{A}$, $\mathbf{B}$, $\mathbf{Q}$, $\mathbf{R}$, t_f and not on $\mathbf{x}_0$.

- **Optimal control initial value**: from equation (2):

$$\widehat{\mathbf{u}}(0) = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(0)\,\mathbf{x}_0\ .$$

- **Closed-loop optimal control at any time** t : at time $t \in [0,\ t_f[$, assuming that the current state $\mathbf{x}(t)$ is known (using a measurement system), the objective is still to bring back the final state to 0 ($\mathbf{x}(t_f)=0$) but the time horizon is now $t_f - t$. The calculus of the current optimal control $\widehat{\mathbf{u}}(t)$ is the same problem than the previous one, just changing $\mathbf{x}_0$ by $\mathbf{x}(t)$ and t_f by $t_f - t$. Thus:

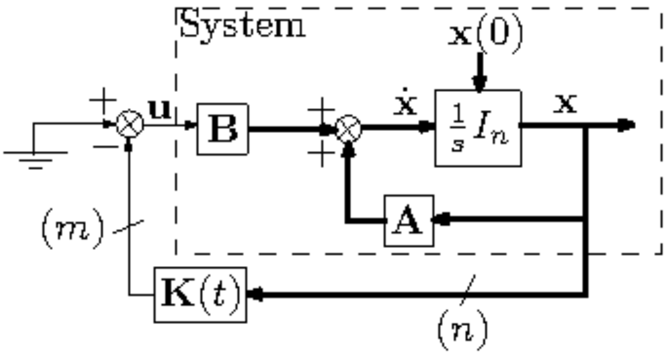
$$\mathbf{\Psi}(t) = \mathbf{P}(t)\,\mathbf{x}(t)\ \text{with:}\ \boxed{\mathbf{P}(t) = -\left[e_{12}^{\mathbf{H}(t_f-t)}\right]^{-1} e_{11}^{\mathbf{H}(t_f-t)}}\ ,$$

$$\widehat{\mathbf{u}}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)\,\mathbf{x}(t) = -\mathbf{K}(t)\,\mathbf{x}(t)\ .$$

with:

$$\boxed{\mathbf{K}(t) = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)}$$

the time-varying state feedback to be implemented in closed-loop according to the following Figure:



Remark: $\mathbf{P}(t_f)$ is not defined since $e_{12}^{\mathbf{H}0} = \mathbf{0}_{n\times n}$ and is not invertible.

- **Optimal state trajectories** : The integration of equation (4) between 0 and t ($\forall\ t \in [0,\ t_f[$) leads to (first n row):

$$\mathbf{x}(t) = e_{11}^{\mathbf{H}t}\,\mathbf{x}_0 + e_{12}^{\mathbf{H}t}\,\mathbf{\Psi}(0) = \left(e_{11}^{\mathbf{H}t} - e_{12}^{\mathbf{H}t}\left[e_{12}^{\mathbf{H}t_f}\right]^{-1} e_{11}^{\mathbf{H}t_f}\right)\,\mathbf{x}_0 = \mathbf{\Phi}(t_f,t)\,\mathbf{x}_0\ .$$

where:

$$\boxed{\mathbf{\Phi}(t_f,t) = e_{11}^{\mathbf{H}t} - e_{12}^{\mathbf{H}t}\left[e_{12}^{\mathbf{H}t_f}\right]^{-1} e_{11}^{\mathbf{H}t_f}}$$

is called the transition matrix.

- **Optimal performance index** :

For any $t \in [0,\ t_f[$ and a current state $\mathbf{x}$ one can define the cost-to-go function (or value-function) $\widehat{\mathcal{R}}(\mathbf{x},t)$ as:

$$\mathcal{R}(\mathbf{x},t) = \frac{1}{2}\int_t^{t_f}(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u})d\tau$$

and the optimal cost-to-go function as:

$$\widehat{\mathcal{R}}(\mathbf{x},t) = \frac{1}{2}\int_t^{t_f}(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \widehat{\mathbf{u}}^T\mathbf{R}\widehat{\mathbf{u}})d\tau$$

$$\widehat{\mathcal{R}}(\mathbf{x},t) = \frac{1}{2}\int_t^{t_f}(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{\Psi}^T\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{\Psi})d\tau\ .$$

From equation (4): one can derive that:

$$\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{\Psi} = \mathbf{A}\mathbf{x} - \dot{\mathbf{x}}\ ,$$

$$\mathbf{Q}\mathbf{x} = -\mathbf{A}^T\mathbf{\Psi} - \dot{\mathbf{\Psi}}$$

Thus (after simplification):

$$\widehat{\mathcal{R}}(\mathbf{x},t) = \frac{1}{2}\int_t^{t_f}(-\mathbf{x}^T\dot{\mathbf{\Psi}} - \mathbf{\Psi}^T\dot{\mathbf{x}})d\tau = -\frac{1}{2}\int_t^{t_f}\frac{d\,(\mathbf{x}^T\mathbf{\Psi})}{d\tau}d\tau = 0 + \frac{1}{2}\mathbf{x}^T(t)\mathbf{\Psi}(t)$$

Thus:

$\widehat{\mathcal{R}}(\mathbf{x},t) = \frac{1}{2}\mathbf{x}^T(t)\mathbf{P}(t)\mathbf{x}(t)$

From this last equation, on can find again the definition of the costate $\mathbf{\Psi}$ used to solve the Hamilton–Jacobi–Bellman equation:

$$\mathbf{\Psi}(t) = \frac{\partial\widehat{\mathcal{R}}(\mathbf{x},t)}{\partial\mathbf{x}}$$

The optimal performance index is: $\widehat{J} = \widehat{\mathcal{R}}(\mathbf{x}_0,0) = \frac{1}{2}\mathbf{x}_0^T\mathbf{P}(0)\mathbf{x}_0$.

Exercises

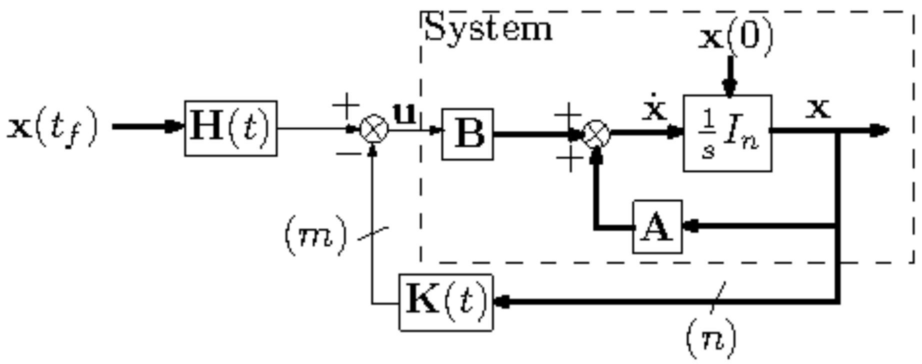
- **Exo #1:** show that $\mathbf{P}(t)$ is the solution of the matrix Riccati differential equation:

$$\dot{\mathbf{P}} = -\mathbf{P}\mathbf{A} - \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} - \mathbf{Q}$$

also written as:

$$\dot{\mathbf{P}} = [-\mathbf{P}\;\;\mathbf{I}_n]\,\mathbf{H}\begin{bmatrix}\mathbf{I}_n\\ \mathbf{P}\end{bmatrix}\;.$$

- **Exo #2:** considering now that $\mathbf{x}(t_f) = \mathbf{x}_f \neq \mathbf{0}$, compute the time-variant state feedback gain $\mathbf{K}(t)$ and the time-variant feedforward gain $\mathbf{H}(t)$ of the optimal closed-loop control law to be implemented according to the following Figure.



Published with MATLAB® R2019b

Linear Quadratic Optimal Control

MAE2 - Aerospace System & Control (ASC)

Thierry MIQUEL
thierry.miquel@enac.fr

ISAE-Supaero

February 2, 2023

Historical perspective

- The application of optimal control theory to the practical design of multivariable control systems started in the 1960s[1]
- In 1960 three major papers were published by R. Kalman and coworkers, working in the U.S.
- In control theory, Kalman introduced linear algebra and matrices, so that systems with multiple inputs and outputs could easily be treated
- He also formalized the notion of optimality in control theory by minimizing a very general quadratic generalized energy function

[1] https://lewisgroup.uta.edu/history.htm

Scope of the lecture

- This lecture focuses on **LQ (Linear Quadratic)** theory
  - **Linear**: considered plants are assumed to be a Linear Time Invariant (LTI)
  - **Quadratic**: the cost to be minimized is quadratic

Outline

1. Fundamentals of Linear Quadratic Regulator (LQR)
2. Robustness property of LQR
3. Design methods
   - Chang-Letov (or symmetric root locus) design procedure
   - Mirror property
   - Tracking
4. Linear Quadratic Gaussian (LQG) control

Positive semi-definite matrix

- A **semi-definite positive matrix** $\boldsymbol{Q}$ is denoted $\boldsymbol{Q} \geq 0$
- We remind that a $n \times n$ real symmetric matrix $\boldsymbol{Q} = \boldsymbol{Q}^T$ is called positive semi-definite if and only if we have either
    - $\underline{x}^T \boldsymbol{Q} \underline{x} \geq 0$ for all $\underline{x} \neq 0$;
    - All eigenvalues of $\boldsymbol{Q}$ are non-negative
    - All of the principal (not only leading) minors are non-negative
    - The principal minor of order k is the minor of order k obtained by deleting $n-k$ rows and the $n-k$ columns with the same position than the rows. For instance, in a principal minor where you have deleted rows 1 and 3, you should also delete columns 1 and 3
    - $\boldsymbol{Q}$ can be written as $\boldsymbol{Q}_s^T \boldsymbol{Q}_s$ where matrix $\boldsymbol{Q}_s$ is full row rank

Positive definite matrix

- A **positive definite matrix** $\boldsymbol{R}$ is denoted $\boldsymbol{R} > 0$
- A real $n \times n$ symmetric matrix $\boldsymbol{R} = \boldsymbol{R}^T$ is called positive definite if and only if we have either
    - $\underline{x}^T \boldsymbol{R} \underline{x} > 0$ for all $\underline{x} \neq 0$
    - All eigenvalues of $\boldsymbol{R}$ are strictly positive
    - All of the leading principal minors are strictly positive
    - The leading principal minor of order k is the minor of order k obtained by deleting the last $n-k$ rows and columns
    - $\boldsymbol{R}$ can be written as $\boldsymbol{R}_s^T \boldsymbol{R}_s$ where matrix $\boldsymbol{R}_s$ is square and invertible
- Furthermore a real symmetric matrix $\boldsymbol{M}$ is called negative (semi-)definite if $-\boldsymbol{M}$ is positive (semi-)definite

Examples

Exercise 1

Check that $\boldsymbol{M}_1 = \boldsymbol{M}_1^T = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$ *is not positive definite*

Examples

Exercise 1

Check that $\boldsymbol{M}_1 = \boldsymbol{M}_1^T = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$ *is not positive definite*

The leading principal minors of $\boldsymbol{M}_1$ *are* 1, *which is strictly positive, and* $\det\left(\begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}\right) = 3 - 4 = -1$, *which is negative*

Because all of the leading principal minors are ***not*** *strictly positive, we conclude that matrix* $\boldsymbol{M}_1$ *is not positive definite*

Examples

Exercise 2

Check that $\boldsymbol{M}_2 = \boldsymbol{M}_2^T = \begin{bmatrix} 1 & -2 \\ -2 & 5 \end{bmatrix}$ *is positive definite*

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control

Examples

Exercise 2

Check that $\boldsymbol{M}_2 = \boldsymbol{M}_2^T = \begin{bmatrix} 1 & -2 \\ -2 & 5 \end{bmatrix}$ *is positive definite*

The leading principal minors of $\boldsymbol{M}_1$ *are* 1*, which is strictly positive, and* $\det\left(\begin{bmatrix} 1 & -2 \\ -2 & 5 \end{bmatrix}\right) = 5 - 4 = 1$*, which is also strictly positive*

Because all of the leading principal minors are strictly positive, we conclude that matrix $\boldsymbol{M}_2$ *is positive definite*

Derivative of scalar valued function

- Let $\boldsymbol{K}\underline{x} \in \mathbb{R}$, where $\boldsymbol{K} \in \mathbb{R}^{1\times n}$ is a row vector, be a scalar valued function of vector $\underline{x} \in \mathbb{R}^n$
- The partial derivative of $\boldsymbol{K}\underline{x}$ with respect to vector $\underline{x}$ reads

$$\frac{\partial}{\partial \underline{x}}\boldsymbol{K}\underline{x} = \boldsymbol{K}^T$$

Exercise 3

- *Let* $\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ *and* $\boldsymbol{K} = \begin{bmatrix} 3 & -1 \end{bmatrix}$. *Check that* $\frac{\partial}{\partial \underline{x}}\boldsymbol{K}\underline{x} = \boldsymbol{K}^T$

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control

Derivative of scalar valued function

- Let $\boldsymbol{K}\underline{x} \in \mathbb{R}$, where $\boldsymbol{K} \in \mathbb{R}^{1\times n}$ is a row vector, be a scalar valued function of vector $\underline{x} \in \mathbb{R}^n$
- The partial derivative of $\boldsymbol{K}\underline{x}$ with respect to vector $\underline{x}$ reads

$$\frac{\partial}{\partial \underline{x}}\boldsymbol{K}\underline{x} = \boldsymbol{K}^T$$

Exercise 3

- *Let* $\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ *and* $\boldsymbol{K} = \begin{bmatrix} 3 & -1 \end{bmatrix}$. *Check that* $\frac{\partial}{\partial \underline{x}}\boldsymbol{K}\underline{x} = \boldsymbol{K}^T$

$$\begin{aligned} \boldsymbol{K}\underline{x} &= \begin{bmatrix} 3 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 3x_1 - x_2 \\ \Rightarrow \frac{\partial}{\partial \underline{x}}\boldsymbol{K}\underline{x} &= \begin{bmatrix} \frac{\partial}{\partial x_1}(3x_1 - x_2) \\ \frac{\partial}{\partial x_2}(3x_1 - x_2) \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \end{bmatrix} = \boldsymbol{K}^T \end{aligned}$$

Table of Contents

1. Fundamentals of Linear Quadratic Regulator (LQR)
2. Robustness property of LQR
3. Design methods
   - Chang-Letov (or symmetric root locus) design procedure
   - Mirror property
   - Tracking
4. Linear Quadratic Gaussian (LQG) control

Problem to be solved

- **Problem**: find the control $\underline{u}(t)$ which minimizes the following quadratic performance index

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^{\infty} \left(\underline{x}^T(t)\boldsymbol{Q}\underline{x}(t) + \underline{u}^T(t)\boldsymbol{R}\underline{u}(t)\right)\, dt$$

  - $\boldsymbol{Q} = \boldsymbol{Q}^T \geq 0$
  - $\boldsymbol{R} = \boldsymbol{R}^T > 0$
- Under the following dynamical constraint

$$\begin{cases} \dot{\underline{x}}(t) = \boldsymbol{A}\underline{x}(t) + \boldsymbol{B}\underline{u}(t) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{1}$$

  - $\underline{x}(t)$ is the state vector of dimension n
  - $\underline{u}(t)$ is the control vector of dimension m
  - $\boldsymbol{A}$ is the state (or system) matrix
  - $\boldsymbol{B}$ is the input matrix

Assumptions

- Pair $(\boldsymbol{A}, \boldsymbol{B})$ is controllable, or at least stabilizable
- Pair $(\boldsymbol{A}, \boldsymbol{N})$, where $\boldsymbol{Q} = \boldsymbol{N}^T\boldsymbol{N}$, is observable, or at least detectable
- Reminder
  - The pair $(\boldsymbol{A}, \boldsymbol{B})$ is said stabilizable if the uncontrollable eigenvalues of $\boldsymbol{A}$, if any, have negative real parts. Thus even though not all system modes are controllable, the ones that are not controllable do not require stabilization
  - The pair $(\boldsymbol{A}, \boldsymbol{N})$ is said detectable if the unobservable eigenvalues of $\boldsymbol{A}$, if any, have negative real parts. Thus even though not all system modes are observable, the ones that are not observable do not require stabilization.

Control

- Let $\underline{\lambda}(t) \in \mathbb{R}^n$ be the vector of Lagrange multipliers
- The Hamiltonian function reads

$$H(\underline{x}, \underline{u}, \underline{\lambda}) = \frac{1}{2}\left(\underline{x}^T(t)\boldsymbol{Q}\underline{x}(t) + \underline{u}^T(t)\boldsymbol{R}\underline{u}(t)\right) + \underline{\lambda}^T(t)\left(\boldsymbol{A}\underline{x}(t) + \boldsymbol{B}\underline{u}(t)\right)$$

- The necessary condition for optimality yields

$$\frac{\partial H}{\partial \underline{u}} = \boldsymbol{R}\underline{u}(t) + \boldsymbol{B}^T\underline{\lambda}(t) = \underline{0}$$

- Taking into account that $\boldsymbol{R}$ is a symmetric matrix, we get

$$\underline{u}(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^T\underline{\lambda}(t) \tag{2}$$

- Eliminating $\underline{u}(t)$ in equation (1) reads

$$\begin{cases} \dot{\underline{x}}(t) = \boldsymbol{A}\underline{x}(t) - \boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\underline{\lambda}(t) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{3}$$

Hamiltonian matrix

- The dynamics of Lagrange multipliers $\underline{\lambda}(t)$ is given by

$$\dot{\underline{\lambda}}(t) = -\frac{\partial H}{\partial \underline{x}} = -\boldsymbol{Q}\underline{x}(t) - \boldsymbol{A}^T\underline{\lambda}(t) \tag{4}$$

- Combining (3) and (4) into a single state equation yields

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\underline{\lambda}}(t) \end{bmatrix} = \boldsymbol{H} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} \tag{5}$$

- The $2n \times 2n$ matrix $\boldsymbol{H}$ is called the **Hamiltonian matrix**

Exercise 4

Combine (3) and (4) to get the expression of the ***Hamiltonian matrix H***

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control

Hamiltonian matrix

- The dynamics of Lagrange multipliers $\underline{\lambda}(t)$ is given by

$$\dot{\underline{\lambda}}(t) = -\frac{\partial H}{\partial \underline{x}} = -\boldsymbol{Q}\underline{x}(t) - \boldsymbol{A}^T\underline{\lambda}(t) \tag{4}$$

- Combining (3) and (4) into a single state equation yields

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\underline{\lambda}}(t) \end{bmatrix} = \boldsymbol{H} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} \tag{5}$$

- The $2n \times 2n$ matrix $\boldsymbol{H}$ is called the **Hamiltonian matrix**

Exercise 4

Combine (3) and (4) to get the expression of the ***Hamiltonian matrix H***

$$\boxed{\begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\underline{\lambda}}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & -\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T \\ -\boldsymbol{Q} & -\boldsymbol{A}^T \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} := \boldsymbol{H} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix}}$$

Algebraic Riccati equation (ARE)

- For the considered problem, we have the following relationship between the vector $\underline{\lambda}(t)$ of Lagrange multipliers and the state vector $\underline{x}(t)$ where $\boldsymbol{P} = \boldsymbol{P}^T \geq 0$ is a constant matrix

$$\boxed{\underline{\lambda}(t) = \boldsymbol{P}\underline{x}(t) \text{ where } \boldsymbol{P} = \boldsymbol{P}^T \geq 0} \tag{6}$$

Exercise 5

Combine (5) and (6) to get the ***Algebraic Riccati Equation (ARE)*** *that matrix* ***P*** *solves*

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control

Algebraic Riccati equation (ARE)

- For the considered problem, we have the following relationship between the vector $\underline{\lambda}(t)$ of Lagrange multipliers and the state vector $\underline{x}(t)$ where $\boldsymbol{P} = \boldsymbol{P}^T \geq 0$ is a constant matrix

$$\boxed{\underline{\lambda}(t) = \boldsymbol{P}\underline{x}(t) \text{ where } \boldsymbol{P} = \boldsymbol{P}^T \geq 0} \tag{6}$$

Exercise 5

Combine (5) and (6) to get the ***Algebraic Riccati Equation (ARE)*** *that matrix* ***P*** *solves*

$$
\begin{aligned}
\dot{\underline{\lambda}}(t) = \boldsymbol{P}\dot{\underline{x}}(t) &\Rightarrow -\boldsymbol{Q}\underline{x}(t) - \boldsymbol{A}^T\underline{\lambda}(t) = \boldsymbol{P}\left(\boldsymbol{A}\underline{x}(t) - \boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\underline{\lambda}(t)\right) \\
\underline{\lambda}(t) = \boldsymbol{P}\underline{x}(t) &\Rightarrow -\boldsymbol{Q}\underline{x}(t) - \boldsymbol{A}^T\boldsymbol{P}\underline{x}(t) = \boldsymbol{P}\left(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}\right)\underline{x}(t) \\
&\Rightarrow \left(\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q}\right)\underline{x}(t) = \underline{0}\,\forall \underline{x}(t)
\end{aligned}
$$

$$\Rightarrow \boxed{\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q} := \begin{bmatrix} \boldsymbol{P} & -\mathbb{I}_n \end{bmatrix} \boldsymbol{H} \begin{bmatrix} \mathbb{I}_n \\ \boldsymbol{P} \end{bmatrix} = 0}$$

Closed-loop eigenvalues

- Since $\underline{\lambda}(t) = \boldsymbol{P}\underline{x}(t)$, the control (2) reads

$$\underline{u}(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^T\underline{\lambda}(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}\underline{x}(t)$$

- That is

$$\boxed{\underline{u}(t) = -\boldsymbol{K}\underline{x}(t) \text{ where } \boldsymbol{K} = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}}$$

- Remember that $\boldsymbol{P} = \boldsymbol{P}^T$ is the **positive definite solution** of the algebraic Riccati equation (ARE)
- It can be shown that the eigenvalues of $\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$ (that are the eigenvalues of the closed loop plant) are equal to the n eigenvalues in the open left half plane of the Hamiltonian matrix $\boldsymbol{H}$

Solving the algebraic Riccati equation

- Furthermore
  - The Hamiltonian matrix $\boldsymbol{H}$ has n eigenvalues in the open left half plane and n eigenvalues in the open right half plane (and no pure imaginary eigenvalues)
  - The eigenvalues are symmetric with respects to the imaginary axis: if $\underline{\lambda}$ is and eigenvalue of $\boldsymbol{H}$ then $-\underline{\lambda}$ is also an eigenvalue of $\boldsymbol{H}$
- Let $\begin{bmatrix} \underline{v}_1 & \cdots & \underline{v}_n \end{bmatrix} := \begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \end{bmatrix}$ be the $2n \times n$ matrix whose columns $\underline{v}_k$ comprise all the eigenvectors of $\boldsymbol{H}$ corresponding to the n eigenvalues in the *open left half plane*
- Then $\boldsymbol{X}_1$ is invertible and the positive definite solution of the **algebraic Riccati equation (ARE)** is

$$\boldsymbol{P} = \boldsymbol{X}_2\boldsymbol{X}_1^{-1}$$

Minimum cost achieved

- The minimum cost achieved is given by

$$J^* = \frac{1}{2}\underline{x}^T(0)\boldsymbol{P}\underline{x}(0)$$

Table of Contents

1. Fundamentals of Linear Quadratic Regulator (LQR)
2. Robustness property of LQR
3. Design methods
   - Chang-Letov (or symmetric root locus) design procedure
   - Mirror property
   - Tracking
4. Linear Quadratic Gaussian (LQG) control

Block diagram of full-state feedback control

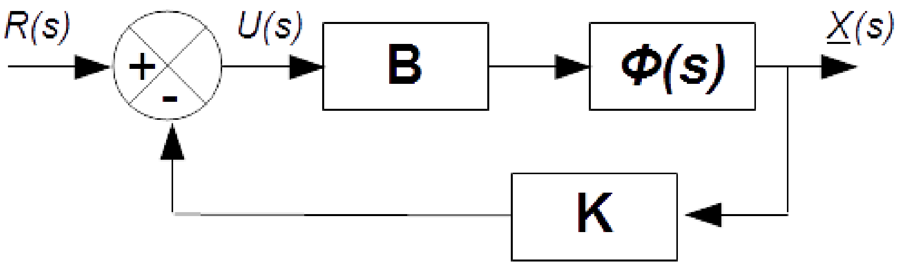
- We consider a linear plant controlled through a state feedback

$$\begin{cases} \dot{\underline{x}}(t) = \boldsymbol{A}\underline{x}(t) + \boldsymbol{B}\underline{u}(t) \\ \underline{u}(t) = -\boldsymbol{K}\underline{x}(t) + \underline{r}(t) \end{cases} \Leftrightarrow \begin{cases} s\underline{X}(s) = \boldsymbol{A}\underline{X}(s) + \boldsymbol{B}\underline{U}(s) \\ \underline{U}(s) = -\boldsymbol{K}\underline{x}(t) + \underline{R}(s) \end{cases}$$

- Let $\boldsymbol{\Phi}(s)$ be resolvent of the state (transition) matrix $\boldsymbol{A}$

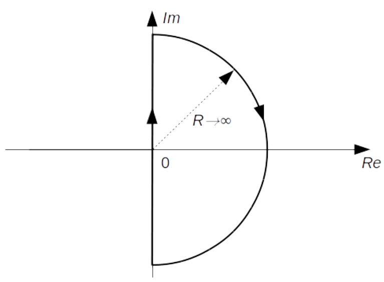
$$\boldsymbol{\Phi}(s) = (s\mathbb{I} - \boldsymbol{A})^{-1}$$

- This leads to the block diagram of the full-state feedback control



Nyquist stability criterion

- The Nyquist plot of $\det(\mathbb{I} + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B})$ is the image of $\det(\mathbb{I} + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B})$ as s goes clockwise around the Nyquist contour



- Generalized (MIMO) Nyquist stability criterion: the number of unstable closed-loop poles (i.e. roots of $\det(s\mathbb{I} - \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K})$) is equal to the number of unstable open-loop poles (i.e. roots of $\det(s\mathbb{I} - \boldsymbol{A})$) plus the number of encirclements of the critical point $(0, 0)$ by the Nyquist plot of $\det(\mathbb{I} + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B})$
- The encirclement is counted positive in the clockwise direction and negative otherwise

Kalman equality

- Kalman has shown the following result, known as **Kalman equality**

$$(\mathbb{I} + \boldsymbol{L}(-s))^T \boldsymbol{R} (\mathbb{I} + \boldsymbol{L}(s)) = \boldsymbol{R} + (\boldsymbol{\Phi}(-s)\boldsymbol{B})^T \boldsymbol{Q} (\boldsymbol{\Phi}(s)\boldsymbol{B})$$

- $\boldsymbol{L}(s)$ is the loop gain

$$\boldsymbol{L}(s) = \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B}$$

- $\boldsymbol{K}$ is the optimal feedback gain (obtained through the algebraic Riccati equation)

Robustness of Linear Quadratic Regulator

- For SISO (Single Input - Single Output) plants, loop gain $\boldsymbol{L}(s) = \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B}$ and $\boldsymbol{R}$ are scalars

Exercise 6

- *Set $\boldsymbol{Q} = \boldsymbol{N}^T\boldsymbol{N}$ and write Kalman equality for SISO plants*

Robustness of Linear Quadratic Regulator

- For SISO (Single Input - Single Output) plants, loop gain $\boldsymbol{L}(s) = \boldsymbol{K\Phi}(s)\boldsymbol{B}$ and $\boldsymbol{R}$ are scalars

Exercise 6

- *Set $\boldsymbol{Q} = \boldsymbol{N}^T\boldsymbol{N}$ and write Kalman equality for SISO plants*

$$(1 + \boldsymbol{L}(-s))\,(1 + \boldsymbol{L}(s)) = 1 + \tfrac{1}{\boldsymbol{R}}\,(\boldsymbol{N\Phi}(-s)\boldsymbol{B})\,(\boldsymbol{N\Phi}(s)\boldsymbol{B})$$

- Substituting $s = j\omega$ yields:

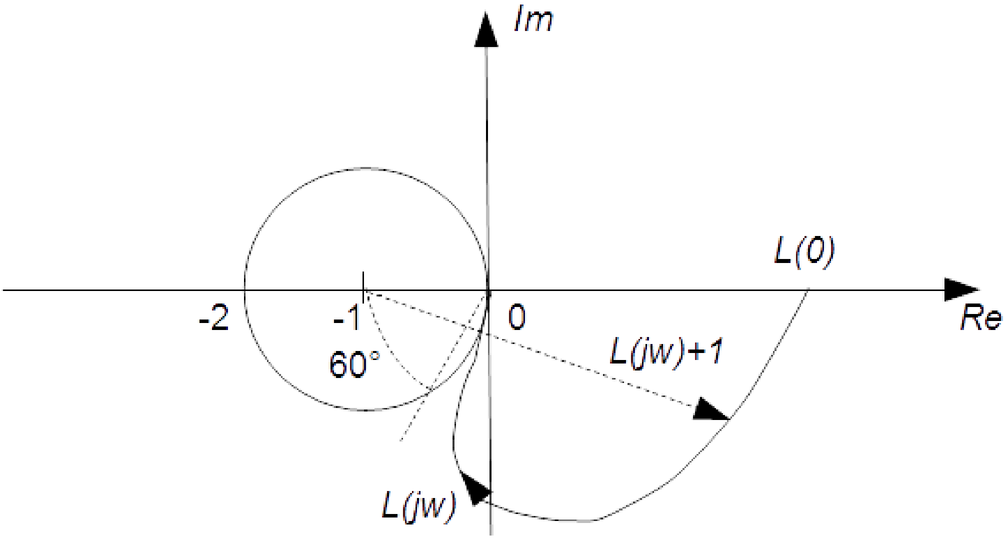
$$\|1 + \boldsymbol{L}(j\omega)\|^2 = 1 + \frac{1}{\boldsymbol{R}}\|\boldsymbol{N\Phi}(j\omega)\boldsymbol{B}\|^2$$

- Therefore:

$$\|1 + \boldsymbol{L}(j\omega)\| \geq 1 \quad \forall\omega \in \mathbb{R}$$

Phase margin of Linear Quadratic Regulator

- $\|1 + \boldsymbol{L}(j\omega)\| \geq 1 \quad \forall\omega \in \mathbb{R}$ indicates that LQR design always leads to a phase margin which is always greater or equal to 60 degrees



Sensitivity function and the complementary sensitivity function

- For SISO plants, the sensitivity function $\boldsymbol{S}(s)$ and the complementary sensitivity function $\boldsymbol{T}(s)$ are defined as follows

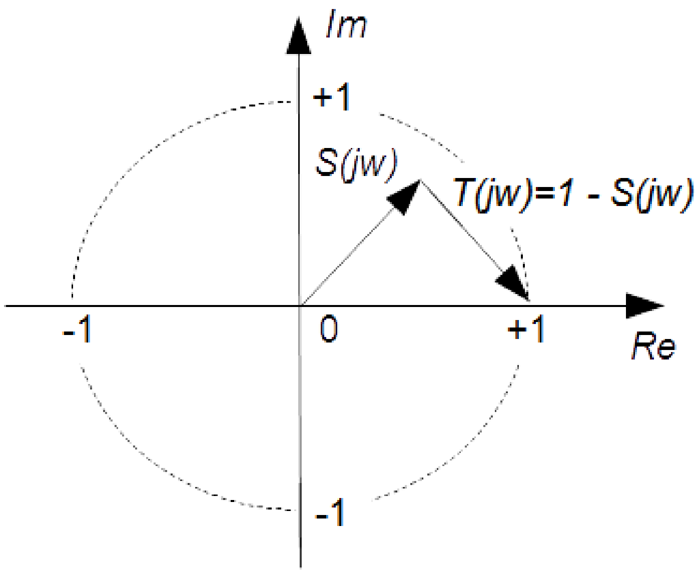
$$\begin{cases} \boldsymbol{S}(s) = \frac{1}{1+\boldsymbol{L}(s)} \\ \boldsymbol{T}(s) = 1 - \boldsymbol{S}(s) = \frac{\boldsymbol{L}(s)}{1+\boldsymbol{L}(s)} \end{cases}$$

- Substituting $s = j\omega$, Kalman's inequality guarantees that

$$\begin{cases} \|\boldsymbol{S}(j\omega)\| \leq 1 \\ \|\boldsymbol{T}(j\omega)\| \leq 2 \end{cases}$$

  - Small sensitivity function $\Leftrightarrow$ good disturbance rejection
  - Complementary sensitivity function close to one $\Leftrightarrow$ good reference tracking
  - Small complementary sensitivity function $\Leftrightarrow$ good noise rejection

Sensitivity function and the complementary sensitivity function



Matlab's hands-on: aircraft pitch control - basics on LQR

The short-period longitudinal dynamics for a medium-sized jet with centre of gravity unusually far aft might be described by the following state equation, where α is the angle of attack, q the pitch rate and u the elevator deflection

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} -1.417 & 1 \\ 2.86 & -1.183 \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ -3.157 \end{bmatrix} u$$

- Compute the open loop poles of the *natural* aircraft. What do you conclude ?
  To stabilize the plant and keep the pitch rate small, the following performance index is chosen:
  $$J(u(t)) = \frac{1}{2} \int_0^\infty q^2(t) + 5u^2(t)\, dt$$
- What are the values of matrices $\boldsymbol{R}$ and $\boldsymbol{Q}$? Check that $\boldsymbol{Q} = \boldsymbol{Q}^T \geq 0$ and $\boldsymbol{R} = \boldsymbol{R}^T > 0$

- What is the value of 1×2 matrix $\boldsymbol{N}$ such that $\boldsymbol{Q}$ reads $\boldsymbol{Q} = \boldsymbol{N}^T\boldsymbol{N}$
- Check that $(\boldsymbol{A}, \boldsymbol{B})$ is controllable and that $(\boldsymbol{A}, \boldsymbol{N})$ is observable
- Compute the positive solution of the algebraic Riccati equation $\boldsymbol{P}$ thanks to the eigenvectors of the Hamiltonian matrix as well the optimal gain $\boldsymbol{K}$. Check that you get the same result either with matlab (command *care* or *icare* for the newest versions)
- Plot the Bode diagram of $\boldsymbol{L}(s) = \boldsymbol{K}(s\mathbb{I} - \boldsymbol{A})^{-1}\boldsymbol{B}$ (*loop gain transfer function*) and compute the phase margin
- Plot the Nyquist diagram of $\boldsymbol{L}(s) = \boldsymbol{K}\,(s\mathbb{I} - \boldsymbol{A})^{-1}\,\boldsymbol{B}$ (*loop gain transfer function*)

Table of Contents

1. Fundamentals of Linear Quadratic Regulator (LQR)
2. Robustness property of LQR
3. Design methods
   - Chang-Letov (or symmetric root locus) design procedure
   - Mirror property
   - Tracking
4. Linear Quadratic Gaussian (LQG) control

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control
Chang-Letov (or symmetric root locus) design procedure
Mirror property
Tracking

Table of Contents

1. Fundamentals of Linear Quadratic Regulator (LQR)
2. Robustness property of LQR
3. Design methods
   - Chang-Letov (or symmetric root locus) design procedure
   - Mirror property
   - Tracking
4. Linear Quadratic Gaussian (LQG) control

Purpose

- The purpose of this section is to have some insight on how to drive the modes of the closed-loop plant thanks to the LQR design
- Cost to be minimized

$$J(u(t)) = \frac{1}{2}\int_0^{\infty}\left(\underline{x}^T(t)\boldsymbol{Q}\underline{x}(t) + u^T(t)\boldsymbol{R}u(t)\right)\,dt$$

- If we set $\boldsymbol{Q} = \boldsymbol{N}^T\boldsymbol{N}$, then $J(u(t))$ reads

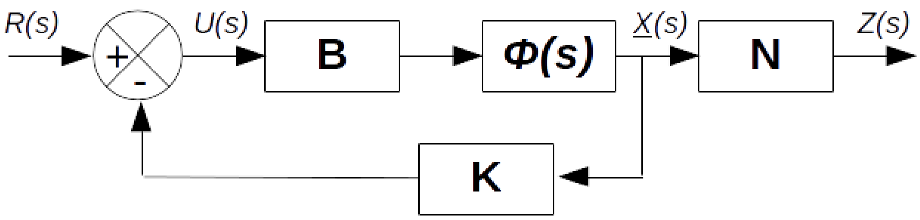
$$\begin{aligned} J(u(t)) &= \tfrac{1}{2}\int_0^{\infty}\left(\underline{x}^T(t)\boldsymbol{N}^T\boldsymbol{N}\underline{x}(t) + \boldsymbol{R}u^2(t)\right)\,dt \\ &= \tfrac{1}{2}\int_0^{\infty}\left(\underline{z}^T(t)\underline{z}(t) + \boldsymbol{R}u^2(t)\right)\,dt \end{aligned}$$

- This cost shall be minimized under the following dynamical constraint

$$\begin{cases} \underline{\dot{x}}(t) = \boldsymbol{A}\underline{x}(t) + \boldsymbol{B}u(t) \\ \underline{z}(t) = \boldsymbol{N}\underline{x}(t) \end{cases}$$

Closed-loop block diagram

- The block diagram of the *closed-loop* plant with $u = -\boldsymbol{K}\underline{x} + r$ is the following



- Matrix $\boldsymbol{\Phi}(s)$ is the resolvent of the state (transition) matrix $\boldsymbol{A}$

$$\boldsymbol{\Phi}(s) = (s\mathbb{I} - \boldsymbol{A})^{-1}$$

- In the time domain, we have

$$\begin{cases} \underline{\dot{x}}(t) = \boldsymbol{A}\underline{x}(t) + \boldsymbol{B}\underline{u}(t) \\ \underline{u}(t) = -\boldsymbol{K}\underline{x}(t) + r(t) \end{cases}$$

Closed-loop transfer function

- The open-loop characteristic polynomial $D(s)$ is given by

$$D(s) := \det(s\mathbb{I} - \boldsymbol{A})$$

- The dynamics of the closed-loop system reads:

$$\dot{\underline{x}}(t) = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\,\underline{x}(t) + \boldsymbol{B}r(t)$$

- The closed-loop characteristic polynomial $\beta(s)$ is given by

$$\beta(s) := \det(s\mathbb{I} - \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K})$$

Eigenvalues of full-state feedback control

- Hsu-Chen theorem states that the following relation holds

$$\boxed{\frac{\beta(s)}{D(s)} := \frac{\det\left(s\mathbb{I} - \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K}\right)}{\det\left(s\mathbb{I} - \boldsymbol{A}\right)} = \det\left(\mathbb{I} + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B}\right)}$$

- To get Hsu-Chen theorem we may use the following relations
  - Sylvester's determinant theorem

$$\det\left(\mathbb{I}_m + \boldsymbol{M}_1\boldsymbol{M}_2\right) = \det\left(\mathbb{I}_n + \boldsymbol{M}_2\boldsymbol{M}_1\right)$$

  - If $\boldsymbol{M}_3$ and $\boldsymbol{M}_4$ are square matrices of equal size then

$$\det\left(\boldsymbol{M}_3\boldsymbol{M}_4\right) = \det\left(\boldsymbol{M}_3\right)\det\left(\boldsymbol{M}_4\right)$$

- For SISO (Single Input - Single Output) plants, loop gain $\boldsymbol{L}(s) := \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B}$ is scalar. Thus the following relation holds

$$\frac{\beta(s)}{D(s)} := \frac{\det\left(s\mathbb{I} - \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K}\right)}{\det\left(s\mathbb{I} - \boldsymbol{A}\right)} = 1 + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B} := 1 + \boldsymbol{L}(s)$$

Eigenvalues of full-state feedback control (cont.)

- Moreover the roots of $\det(\mathbb{I} + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B})$ are exactly the roots of $\det(s\mathbb{I} - \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K})$
- Indeed $\boldsymbol{\Phi}(s)$ is computed as the adjugate of $(s\mathbb{I} - \boldsymbol{A})$ divided by $\det(s\mathbb{I} - \boldsymbol{A})$, which is actually the denominator of $\det(\mathbb{I} + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B})$
- Thus, the eigenvalues of the full-state feedback loop are the roots of $\det(\mathbb{I} + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B})$

$$\det(s\mathbb{I} - \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K})|_{s=\lambda} = 0 \Leftrightarrow \det(\mathbb{I} + \boldsymbol{K}\boldsymbol{\Phi}(s)\boldsymbol{B})|_{s=\lambda} = 0$$

Chang-Letov (or symmetric root locus) design procedure

Exercise 7

- *Write the Kalman equality for SISO plants (remember that $\boldsymbol{Q} = \boldsymbol{N}^T\boldsymbol{N}$)*

Chang-Letov (or symmetric root locus) design procedure

Exercise 7

- *Write the Kalman equality for SISO plants (remember that* $\boldsymbol{Q} = \boldsymbol{N}^T\boldsymbol{N}$*)*
- *For SISO plants, weight* $\boldsymbol{R}$ *is scalar. Thus the Kalman equality reads*

$$(1 + \boldsymbol{L}(-s))\,(1 + \boldsymbol{L}(s)) = 1 + \tfrac{1}{\boldsymbol{R}}\,(\boldsymbol{N\Phi}(-s)\boldsymbol{B})\,(\boldsymbol{N\Phi}(s)\boldsymbol{B})$$

- *Moreover for SISO plants,* $\frac{\beta(s)}{D(s)} := \frac{det(s\mathbb{I}-\boldsymbol{A}+\boldsymbol{BK})}{det(s\mathbb{I}-\boldsymbol{A})} = 1 + \boldsymbol{L}(s)$
- *We finally get*

$$\frac{\beta(-s)}{D(-s)}\frac{\beta(s)}{D(s)} = 1 + \tfrac{1}{\boldsymbol{R}}\,(\boldsymbol{N\Phi}(-s)\boldsymbol{B})\,(\boldsymbol{N\Phi}(s)\boldsymbol{B})$$

Symmetric root locus

- From the previous result, Chang-Letov have rewritten Kalman equality as follows

$$\frac{\beta(-s)}{D(-s)}\frac{\beta(s)}{D(s)} = 1 + \tfrac{1}{\boldsymbol{R}}F(-s)F(s)$$

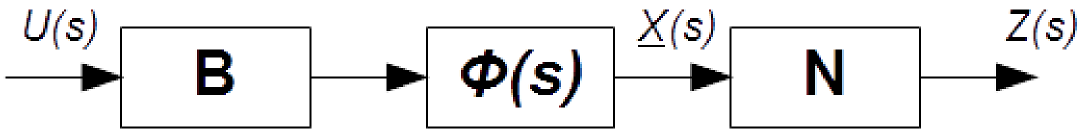
- where the *fictitious* transfer function $F(s)$ reads

$$F(s) := \frac{N(s)}{D(s)} = \boldsymbol{N\Phi}(s)\boldsymbol{B}$$

- Thus, the closed-loop eigenvalues when $k_p := 1/\boldsymbol{R}$ varies from 0 to $+\infty$ are obtained from the root locus of $F(s)F(-s)$
- This is the so called **symmetric root locus** design procedure

Symmetric root locus (cont.)

- This leads to a full-state feedback control with *fictitious* output z



- Output matrix $\boldsymbol{N}$ is a design parameter
- Closed loop characteristic polynomial reads

$$\beta(s)\beta(-s) = D(s)D(-s) + \frac{1}{\boldsymbol{R}}N(s)N(-s) \text{ where } \boldsymbol{N\Phi}(s)\boldsymbol{B} := \frac{N(s)}{D(s)}$$

- Once $\boldsymbol{N}$ is set, then $\boldsymbol{Q} = \boldsymbol{N}^T\boldsymbol{N}$

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control
Chang-Letov (or symmetric root locus) design procedure
Mirror property
Tracking

Comment on the symmetric root locus

- For *high* values of R we have:

$$\beta(s)\beta(-s) = D(s)D(-s) + \frac{1}{\boldsymbol{R}}N(s)N(-s) \xrightarrow[\boldsymbol{R}\to\infty]{} D(s)D(-s)$$

  - When $\boldsymbol{R} \to \infty$, the closed-loop poles tend towards the n roots of $D(s)D(-s)$ with negative real part
- For *small* values of $\boldsymbol{R}$ we have:

$$\beta(s)\beta(-s) = D(s)D(-s) + \frac{1}{\boldsymbol{R}}N(s)N(-s) \xrightarrow[\boldsymbol{R}\to 0]{} \frac{1}{\boldsymbol{R}}N(s)N(-s)$$

  - When $\boldsymbol{R} \to 0$, $m = deg(N(s))$ closed-loop poles tend towards the m roots of $N(s)N(-s)$ with negative real part
  - The remaining $n - m$ closed-loop poles asymptotically approach infinity in the left half plane

$$(-1)^n s^{2(n-m)} \approx \frac{b_m^2}{\boldsymbol{R}}(-1)^m \text{ where } N(s) = b_m\, s^m + b_{m-1}\, s^{m-1} + \cdots$$

Table of Contents

1. Fundamentals of Linear Quadratic Regulator (LQR)
2. Robustness property of LQR
3. Design methods
   - Chang-Letov (or symmetric root locus) design procedure
   - Mirror property
   - Tracking
4. Linear Quadratic Gaussian (LQG) control

Mirror property

- We recall the expression of the $2n \times 2n$ Hamiltonian matrix $\boldsymbol{H}$:

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{A} & -\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T \\ -\boldsymbol{Q} & -\boldsymbol{A}^T \end{bmatrix}$$

- This corresponds to the following algebraic Riccati equation:

$$\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q} = 0$$

- Setting $\boldsymbol{Q} = \boldsymbol{Q}^T = 2\alpha\boldsymbol{P}$, where $\alpha \geq 0$ is a design parameter, the algebraic Riccati equation reads:

$$(\boldsymbol{A} + \alpha\mathbb{I})^T \boldsymbol{P} + \boldsymbol{P}(\boldsymbol{A} + \alpha\mathbb{I}) - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} = 0$$

- This corresponds to the following Hamiltonian matrix $\boldsymbol{H}$:

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{A} + \alpha\mathbb{I} & -\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T \\ 0 & -(\boldsymbol{A} + \alpha\mathbb{I})^T \end{bmatrix}$$

Mirror property (cont.)

- Let λ_i be the open-loop eigenvalues, that are the eigenvalues of matrix $\boldsymbol{A}$
- Let λ_{Ki} be the closed-loop eigenvalues, that are the eigenvalues of matrix $\boldsymbol{A} - \boldsymbol{BK}$
- Given a controllable pair $(\boldsymbol{A}, \boldsymbol{B})$, a positive definite symmetric matrix $\boldsymbol{R}$ and a constant $\alpha \geq 0$, the closed-loop eigenvalues λ_{Ki} have the following *mirror property*:

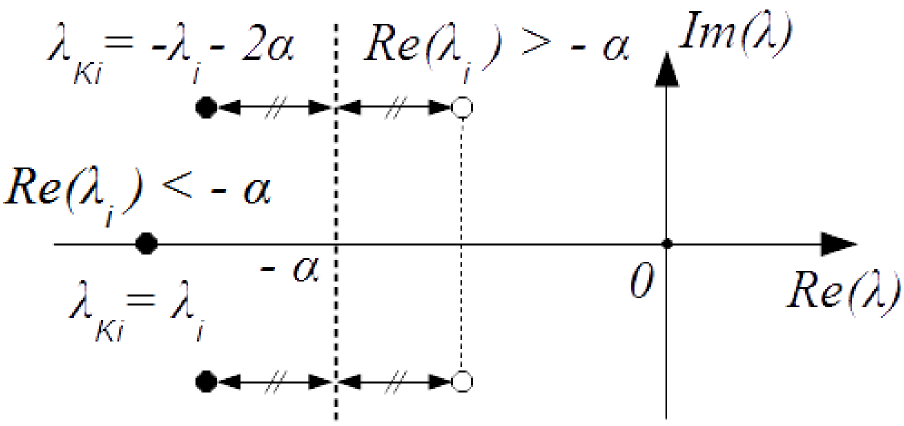
$$\boldsymbol{Q} = \boldsymbol{Q}^T = 2\alpha \boldsymbol{P} \Rightarrow \begin{cases} Re(\lambda_i) \leq -\alpha \Rightarrow \lambda_{Ki} = \lambda_i \\ Re(\lambda_i) > -\alpha \Rightarrow \lambda_{Ki} = -\lambda_i - 2\alpha \end{cases}$$

- Thus

$$\begin{cases} Re(\lambda_{Ki}) \leq -\alpha \\ Im(\lambda_{Ki}) = Im(\lambda_i) \end{cases} \quad \forall\ i = 1, \cdots, n$$

Mirror property (cont.)

- The *mirror property* is illustrated hereafter



- Once the algebraic Riccati equation is solved in $\boldsymbol{P}$ the classical LQR design is applied:

$$\begin{cases} u(t) = -\boldsymbol{K}\underline{x}(t) \\ \boldsymbol{K} = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} \end{cases}$$

Table of Contents

1. Fundamentals of Linear Quadratic Regulator (LQR)
2. Robustness property of LQR
3. Design methods
   - Chang-Letov (or symmetric root locus) design procedure
   - Mirror property
   - Tracking
4. Linear Quadratic Gaussian (LQG) control

Control with feedforward gain

- We will consider in this section the following linear system, where $\underline{x}(t)$ is the state vector, $\underline{u}(t)$ the control and $\underline{y}(t)$ the controlled output (that is the output of interest):

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\,\underline{x}(t) \end{cases}$$

- We will assume that control $\underline{u}(t)$ has the following expression where $\mathbf{F}$ is the feedforward matrix gain and where $\underline{r}(t)$ is the commanded value for the output $\underline{y}(t)$:

$$\underline{u}(t) = -\mathbf{K}\,\underline{x}(t) + \mathbf{F}\,\underline{r}(t)$$

Suboptimal tracking

- First the commanded value $\underline{r}(t)$ is set to zero and the gain $\boldsymbol{K}$ is computed to solve the Linear Quadratic Regulator (LQR) problem
- Then the feedforward matrix gain $\boldsymbol{F}$ is computed such that the steady state value of output $\underline{y}(t)$ is equal to the commanded value $\underline{r}(t) := \underline{y}_c$.
- Assuming that $\dot{\underline{x}} = \underline{0}$, we get

$$\begin{cases} \underline{0} = (\boldsymbol{A} - \boldsymbol{BK})\underline{x} + \boldsymbol{BF}\underline{y}_c \Leftrightarrow \underline{x} = -(\boldsymbol{A} - \boldsymbol{BK})^{-1}\boldsymbol{BF}\underline{y}_c \\ \underline{y} = \boldsymbol{C}\,\underline{x} \end{cases}$$

That is:

$$\underline{y} = -\boldsymbol{C}(\boldsymbol{A} - \boldsymbol{BK})^{-1}\boldsymbol{BF}\underline{y}_c$$

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control
Chang-Letov (or symmetric root locus) design procedure
Mirror property
Tracking

Feedforward gain computation

- Setting $\underline{y}$ to $\underline{y}_c$ and assuming that the size of the output vector $\underline{y}(t)$ is the same than the size of the control vector $\underline{u}$ (square plant) leads to the following expression of the feedforward gain $\boldsymbol{F}$:

$$\underline{y} = \underline{y}_c \Rightarrow \boldsymbol{F} = -\left(\boldsymbol{C}\left(\boldsymbol{A} - \boldsymbol{BK}\right)^{-1}\boldsymbol{B}\right)^{-1}$$

- For a square plant the feedforward gain $\boldsymbol{F}$ is nothing than the inverse of the closed-loop static gain (the closed-loop static gain is obtained by setting the Laplace variable s to 0 in the expression of the closed-loop transfer function).

Plant augmented with integrator

- An alternative to make the steady state error exactly equal to zero in response to a step for the commanded value $\underline{r}(t) = \underline{y}_c$ is to replace the feedforward matrix gain $\boldsymbol{F}$ by an integrator which will cancel the steady state error whatever the input step
- The advantage of adding an integrator is that it eliminates the need to determine the feedforward matrix gain $\boldsymbol{F}$ which could be difficult because of the uncertainty in the model
- By augmenting the system with the integral error the LQR routine will choose the value of the integral gain automatically

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control
Chang-Letov (or symmetric root locus) design procedure
Mirror property
Tracking

Plant augmented with integrator (cont.)

- The integrator is denoted $\boldsymbol{T}/s$, where $\boldsymbol{T} \neq 0$ is a constant
- Let $\underline{x}_i$ be the additional component of the state vector which is proportional to the integral of the error $\underline{e}(t) = \underline{r}(t) - \underline{y}(t)$
- Adding an integrator augments the system's dynamics as follows

$$\begin{cases} \dot{\underline{x}}(t) = \boldsymbol{A}\underline{x}(t) + \boldsymbol{B}\underline{u}(t) \\ \underline{y}(t) = \boldsymbol{C}\,\underline{x}(t) \\ \dot{\underline{x}}_i(t) = \boldsymbol{T}\underline{e}(t) = \boldsymbol{T}\left(\underline{r}(t) - \underline{y}(t)\right) = \boldsymbol{T}\underline{r}(t) - \boldsymbol{T}\boldsymbol{C}\,\underline{x}(t) \end{cases}$$

$$\Leftrightarrow \begin{cases} \frac{d}{dt}\begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & 0 \\ -\boldsymbol{T}\boldsymbol{C} & 0 \end{bmatrix}\begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B} \\ 0 \end{bmatrix}\underline{u}(t) + \begin{bmatrix} 0 \\ \boldsymbol{T} \end{bmatrix}\underline{r}(t) \\ \underline{y}(t) = \begin{bmatrix} \boldsymbol{C} & 0 \end{bmatrix}\begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} \end{cases}$$

Plant augmented with integrator (cont.)

- The suboptimal control is found by solving the LQR regulation problem where $\underline{r} = \underline{0}$:
- The augmented state space model reads:

$$\frac{d}{dt}\begin{bmatrix}\underline{x}(t)\\ \underline{x}_i(t)\end{bmatrix} = \dot{\underline{x}}_a(t) = \boldsymbol{A}_a\underline{x}_a(t) + \boldsymbol{B}_a\underline{u}(t)$$

$$\text{where}\begin{cases}\boldsymbol{A}_a = \begin{bmatrix}\boldsymbol{A} & 0\\ -\boldsymbol{T}\boldsymbol{C} & 0\end{bmatrix}\\ \boldsymbol{B}_a = \begin{bmatrix}\boldsymbol{B}\\ 0\end{bmatrix}\end{cases}$$

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control
Chang-Letov (or symmetric root locus) design procedure
Mirror property
Tracking

Plant augmented with integrator (cont.)

- The performance index $J(\underline{u}(t))$ to be minimized is the following:

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^{\infty} \underline{x}_a^T(t)\boldsymbol{Q}_a\underline{x}_a(t) + \underline{u}^T(t)\boldsymbol{R}\underline{u}(t)\, dt$$

Where, denoting by $\boldsymbol{N}_a$ a design matrix, matrix $\boldsymbol{Q}_a$ is defined as follows:

$$\boldsymbol{Q}_a = \boldsymbol{N}_a^T\boldsymbol{N}_a$$

- Note that design matrix $\boldsymbol{N}_a$ shall be chosen such pair $(\boldsymbol{A}_a, \boldsymbol{N}_a)$ is detectable.

Plant augmented with integrator (cont.)

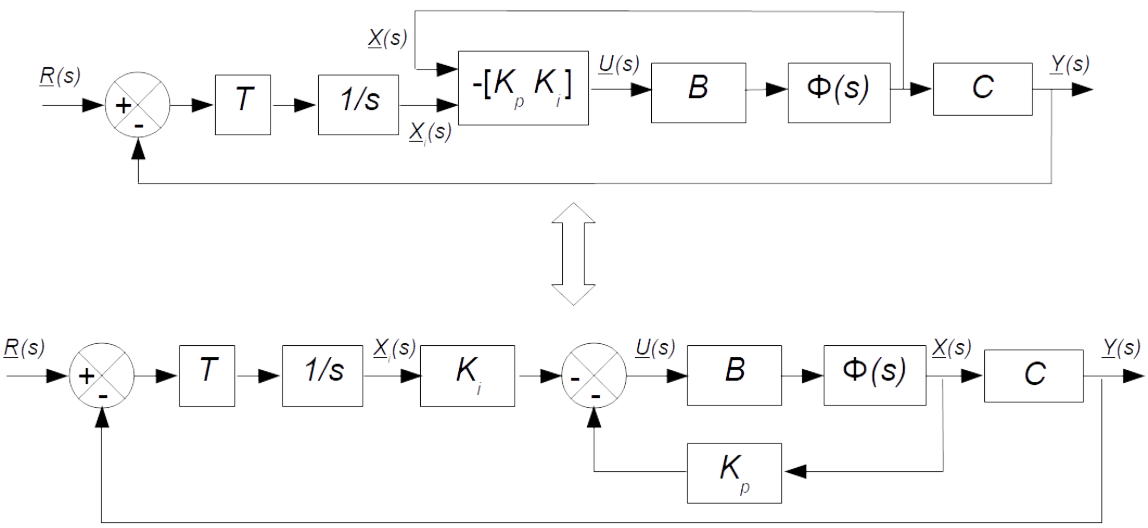
- Assuming that pair $(\boldsymbol{A}_a, \boldsymbol{B}_a)$ is stabilizable and pair $(\boldsymbol{A}_a, \boldsymbol{N}_a)$ is detectable the algebraic Riccati equation can be solved
- This leads to the following expression of the control $\underline{u}(t)$

$$
\begin{aligned}
\underline{u}(t) &= -\boldsymbol{K}_a \underline{x}_a(t) \\
&= -\boldsymbol{R}^{-1}\boldsymbol{B}_a^T\boldsymbol{P}\,\underline{x}_a(t) \\
&= -\boldsymbol{R}^{-1}\begin{bmatrix}\boldsymbol{B}^T & 0\end{bmatrix}\begin{bmatrix}\boldsymbol{P}_{11} & \boldsymbol{P}_{12} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22}\end{bmatrix}\begin{bmatrix}\underline{x}(t) \\ \underline{x}_i(t)\end{bmatrix} \\
&= -\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}_{11}\,\underline{x}(t) - \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}_{12}\,\underline{x}_i(t) \\
&:= -\boldsymbol{K}_p\,\underline{x}(t) - \boldsymbol{K}_i\,\underline{x}_i(t)
\end{aligned}
$$

- Term $\boldsymbol{K}_p = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}_{11}$ is the *proportional* gain of the control
- Term $\boldsymbol{K}_i = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}_{12}$ is the *integral* gain of the control
- Pre-filter gain $\boldsymbol{F}$ no more exists

Plant augmented with integrator (cont.)

The corresponding bloc diagram is shown as follows where $\boldsymbol{\Phi}(s) = (s\mathbb{I} - \boldsymbol{A})^{-1}$.



Matlab's hands-on: aircraft pitch control - LQR design procedures

The short-period longitudinal dynamics for a medium-sized jet with centre of gravity unusually far aft might be described by the following state equation, where α is the angle of attack, q the pitch rate and u the elevator deflection

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} -1.417 & 1 \\ 2.86 & -1.183 \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ -3.157 \end{bmatrix} u$$

The following performance index is chosen

$$J(u(t)) = \frac{1}{2} \int_0^{\infty} q^2(t) + \boldsymbol{R} u^2(t)\, dt$$

- Taking $\boldsymbol{R} = 1000$ (control is weighted very heavily), compute the value of $\boldsymbol{P}$, solution of the algebraic Riccati equation, as well as the optimal gain $\boldsymbol{K}$

Let:

- $G(s) = \boldsymbol{N} (s\mathbb{I} - \boldsymbol{A})^{-1} \boldsymbol{B} = \frac{N(s)}{D(s)}$ be the open loop transfer function;
- $D(s) = \det (s\mathbb{I} - \boldsymbol{A})$ be the open loop characteristics polynomial;
- $N(s)$ be the numerator of $G(s) = \boldsymbol{N} (s\mathbb{I} - \boldsymbol{A})^{-1} \boldsymbol{B}$;
- $\beta(s) = \det (s\mathbb{I} - \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K})$ be the closed-loop characteristic polynomial.
- Check that:

$$\beta(s)\beta(-s) = D(s)D(-s) + \frac{1}{\boldsymbol{R}} N(s)N(-s)$$

- Compute the poles of the closed-loop system, which are the roots of $\det(s\mathbb{I} - \boldsymbol{A} + \boldsymbol{BK})$, and compare them with the poles of the open loop system, which are the roots of $\det(s\mathbb{I} - \boldsymbol{A})$. What do you conclude on poles placement when $\boldsymbol{R} \to \infty$?
- Plot the root locus of $G(s)G(-s) = \frac{N(s)N(-s)}{D(s)D(-s)}$. Using the *Data Cursor* facilities choose a value for the loop gain k_p, set $\boldsymbol{R} = 1/k_p$ and check the values of the poles of the closed-loop system

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control
Chang-Letov (or symmetric root locus) design procedure
Mirror property
Tracking

- Taking $\boldsymbol{R} = 0.01$ (the control is cheap), compute the value of $\boldsymbol{P}$, solution of the algebraic Riccati equation, as well as the optimal gain $\boldsymbol{K}$
- Compute the poles of the closed-loop system and compare them with the zeros of the open loop system $G(s)$. What do you conclude on poles placement when $\boldsymbol{R} \to 0$?
- How can you choose weighting matrix $\boldsymbol{Q}$ such that all the closed-loop poles are faster than 2 rad/sec ?

- Assuming that $u(t) = -\boldsymbol{K}\underline{x}(t) + \boldsymbol{F}q_c(t)$, where $q_c(t)$ is the commanded pitch rate and $\boldsymbol{F}$ the pre-filter gain, plot the step response of the closed-loop transfer function $\frac{Q(s)}{Q_c(s)}$ assuming that $\boldsymbol{F} = 1$
- Compute the value of the pre-filter gain $\boldsymbol{F}$ such that the steady state error is zero for a step input. Plot the step response of the closed-loop transfer function
- Design an LQ tracker which enables to track the commanded pitch rate q_c by adding an integrator in the loop.

Table of Contents

1 Fundamentals of Linear Quadratic Regulator (LQR)

2 Robustness property of LQR

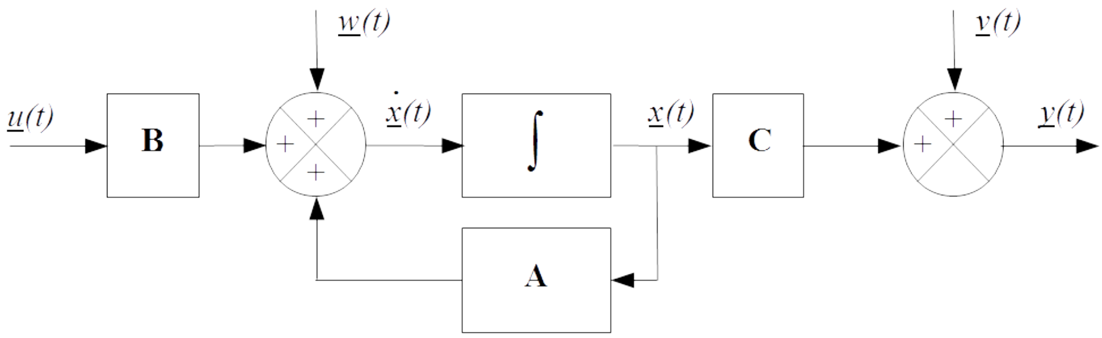
3 Design methods
- Chang-Letov (or symmetric root locus) design procedure
- Mirror property
- Tracking

4 Linear Quadratic Gaussian (LQG) control

Plant model

- Now, the process to be controlled is described by the following linear time invariant model where $\underline{w}(t)$ and $\underline{v}(t)$ are random vectors which represents the process noise and the measurement noise, respectively:

$$\begin{cases} \dot{\underline{x}}(t) = \boldsymbol{A}\underline{x}(t) + \boldsymbol{B}\underline{u}(t) + \underline{w}(t) \\ \underline{y}(t) = \boldsymbol{C}\underline{x}(t) + \underline{v}(t) \end{cases}$$



Assumptions

- Random vectors $\underline{w}(t)$ and $\underline{v}(t)$ are zero mean Gaussian noise. Let $p(\underline{w})$ and $p(\underline{v})$ be the probability density function (pdf) of random vectors $\underline{w}(t)$ and $\underline{v}(t)$. Then:

$$\begin{cases} p(\underline{w}) = \dfrac{1}{(2\pi)^{n/2}\sqrt{\det(\boldsymbol{P}_w)}} e^{-\frac{1}{2}\underline{w}^T \boldsymbol{P}_w^{-1}\underline{w}} \\ p(\underline{v}) = \dfrac{1}{(2\pi)^{p/2}\sqrt{\det(\boldsymbol{P}_v)}} e^{-\frac{1}{2}\underline{v}^T \boldsymbol{P}_v^{-1}\underline{v}} \end{cases}$$

- Random vectors $\underline{w}(t)$ and $\underline{v}(t)$ are white noise (i.e. uncorrelated). The covariance matrices of $\underline{w}(t)$ and $\underline{v}(t)$ will be denoted $\boldsymbol{P}_w$ and $\boldsymbol{P}_v$ respectively:

$$\begin{cases} E\left[\underline{w}(t)\underline{w}^T(t+\tau)\right] = \boldsymbol{P}_w\,\delta(\tau) \text{ where } \boldsymbol{P}_w > 0 \\ E\left[\underline{v}(t)\underline{v}^T(t+\tau)\right] = \boldsymbol{P}_v\,\delta(\tau) \text{ where } \boldsymbol{P}_v > 0 \end{cases}$$

- The cross correlation between $\underline{w}(t)$ and $\underline{v}(t)$ is zero:

$$E\left[\underline{w}(t)\underline{v}^T(t+\tau)\right] = 0 \text{ and } E\left[\underline{v}(t)\underline{w}^T(t+\tau)\right] = 0$$

Problem to be solved

- The LQG control problem is to find the optimal control $\underline{u}(t)$ which minimizes the following performance index $J(\underline{u}(t))$ where $E()$ is the mathematical expectation, $\boldsymbol{Q} = \boldsymbol{Q}^T \geq 0$ and $\boldsymbol{R} = \boldsymbol{R}^T > 0$:

$$J(\underline{u}(t)) = E\left(\lim_{t_f \to \infty} \frac{1}{2\, t_f} \int_0^{t_f} \underline{x}^T(t)\boldsymbol{Q}\underline{x}(t) + \underline{u}^T(t)\boldsymbol{R}\underline{u}(t)\, dt\right)$$

Solution of the LQG problem

- The solution of the LQG problem is obtained by applying the separation principle
- First assume an exact measurement of the full state to solve the deterministic Linear Quadratic (LQ) control problem which minimizes the following cost functional $J(\underline{u}(t))$:

$$J(\underline{u}(t)) = \frac{1}{2} \int_0^{\infty} \underline{x}^T(t)\boldsymbol{Q}\underline{x}(t) + \underline{u}^T(t)\boldsymbol{R}\underline{u}(t)dt$$

- This leads to the following stabilizing control $\underline{u}(t)$:

$$\underline{u}(t) = -\boldsymbol{K}\underline{x}(t)$$

Solution of the LQG problem (cont.)

- Then obtain an optimal estimate of the state which minimizes the estimated error covariance

$$E\left[\underline{e}^T(t)\underline{e}(t)\right] = E\left[\left(\underline{x}(t) - \hat{\underline{x}}(t)\right)^T\left(\underline{x}(t) - \hat{\underline{x}}(t)\right)\right]$$

- This leads to the Kalman-Bucy filter:

$$\frac{d}{dt}\hat{\underline{x}}(t) = \boldsymbol{A}\hat{\underline{x}}(t) + \boldsymbol{B}\underline{u}(t) + \boldsymbol{L}\left(\underline{y}(t) - \boldsymbol{C}\hat{\underline{x}}(t)\right)$$

- It is worth noticing that the optimal state estimate is independent of $\boldsymbol{Q}$ and $\boldsymbol{R}$
- The stabilizing control $\underline{u}(t)$ now reads:

$$\underline{u}(t) = -\boldsymbol{K}\hat{\underline{x}}(t)$$

Observer gain and duality principle

- The observer gain $\boldsymbol{L}$ reads

$$\boldsymbol{L} = \boldsymbol{Y}\boldsymbol{C}^T\boldsymbol{P}_v^{-1}$$

- Matrix $\boldsymbol{Y} = \boldsymbol{Y}^T > 0$ is the solution of the following ARE

$$0 = \boldsymbol{A}\boldsymbol{Y} + \boldsymbol{Y}\boldsymbol{A}^T - \boldsymbol{Y}\boldsymbol{C}^T\boldsymbol{P}_v^{-1}\boldsymbol{C}\boldsymbol{Y} + \boldsymbol{P}_w$$

- The preceding ARE can be obtained through duality principle

Controller	Observer
$\boldsymbol{A}$	$\boldsymbol{A}^T$
$\boldsymbol{B}$	$\boldsymbol{C}^T$
$\boldsymbol{C}$	$\boldsymbol{B}^T$
$\boldsymbol{K}$	$\boldsymbol{L}^T$
$\boldsymbol{P} = \boldsymbol{P}^T \geq 0$	$\boldsymbol{Y} = \boldsymbol{Y}^T \geq 0$
$\boldsymbol{Q} = \boldsymbol{Q}^T \geq 0$	$\boldsymbol{P}_w = \boldsymbol{P}_w^T \geq 0$
$\boldsymbol{R} = \boldsymbol{R}^T > 0$	$\boldsymbol{P}_v = \boldsymbol{P}_v^T > 0$

Controller transfer function

- The controller transfer function $\boldsymbol{K}(s)$ reads:

$$\underline{U}(s) = -\boldsymbol{K}\,\widehat{\underline{X}}(s) = -\boldsymbol{K}(s)\,\underline{Y}(s)$$

where

$$\boldsymbol{K}(s) = \boldsymbol{K}\left(s\mathbb{I} - \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K} + \boldsymbol{L}\boldsymbol{C}\right)^{-1}\boldsymbol{L}$$

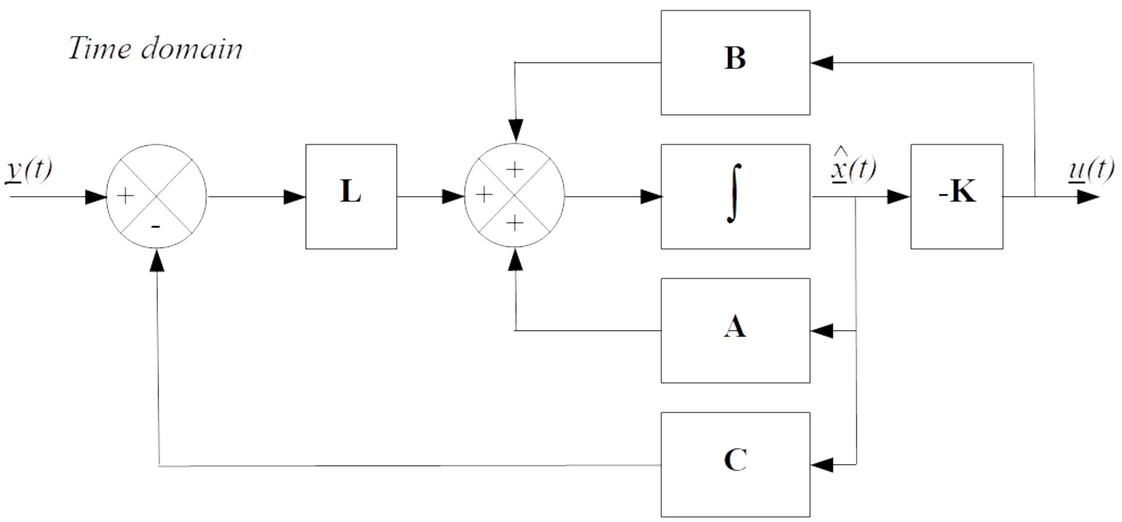
- State space representation of the controller

$$\begin{bmatrix} \dot{\hat{\underline{x}}}(t) \\ \underline{u}(t) \end{bmatrix} \left[\begin{array}{c|c} \boldsymbol{A}_K & \boldsymbol{B}_K \\ \hline \boldsymbol{C}_K & \boldsymbol{D}_K \end{array}\right] \begin{bmatrix} \hat{\underline{x}}(t) \\ \underline{y}(t) \end{bmatrix}$$

Where:

$$\left[\begin{array}{c|c} \boldsymbol{A}_K & \boldsymbol{B}_K \\ \hline \boldsymbol{C}_K & \boldsymbol{D}_K \end{array}\right] = \left[\begin{array}{c|c} \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K} - \boldsymbol{L}\boldsymbol{C} & \boldsymbol{L} \\ \hline \boldsymbol{K} & 0 \end{array}\right]$$

Controller block diagram



Matlab's hands-on: Velocity Aided Inertial Navigation

We consider a vehicle moving close to the surface of the earth. The 2D navigation equations linearized around $\theta(t) = 0$ read as follows:

$$\begin{cases} \dot{\widetilde{u}}(t) = -g\,\widetilde{\theta}(t) + \widetilde{a}_x(t) \\ \dot{\widetilde{\theta}}(t) = \frac{\widetilde{u}(t)}{R} + w_g(t) \end{cases}$$

where:

- $g = 9.81\ m/s^2$ is the acceleration of gravity;
- $R = 6\,371 \cdot 10^3\ m$ is the radius of the earth;
- $\widetilde{u}(t)$ is the *noisy* velocity of the vehicle;
- $\widetilde{\theta}(t)$ is the *noisy* pitch of the vehicle;
- $\widetilde{a}_x(t) = a_x(t) + w_a(t)$ is the output of the accelerometer, where $a_x(t)$ is the *actual* acceleration of the vehicle;

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control

Matlab's hands-on: Velocity Aided Inertial Navigation

Vectors $w_a(t)$ and $w_g(t)$ represent the accelerometer and gyroscope noise processes, respectively. They are assumed to be zero mean independent white noise with covariance P_{w_a} and P_{w_g}:

$$\begin{cases} E\left[w_a(t)w_a(t+\tau)\right] = P_{w_a}\,\delta(\tau) \text{ where } P_{w_a} > 0 \\ E\left[w_g(t)w_g(t+\tau)\right] = P_{w_g}\,\delta(\tau) \text{ where } P_{w_g} > 0 \end{cases}$$

We will assume that a velocity aid provides a noisy measurement $\widetilde{y}(t)$ of the *actual* velocity $u(t)$:

$$\widetilde{y}(t) = u(t) + v(t)$$

where $v(t)$ is assumed to be a zero mean white noise with covariance $\boldsymbol{P}_v$:

$$E\left[v(t)v(t+\tau)\right] = \boldsymbol{P}_v\,\delta(\tau) \text{ where } \boldsymbol{P}_v > 0$$

Matlab's hands-on: Velocity Aided Inertial Navigation

The state vector $\underline{x}(t)$ is chosen to be form with the *actual* values of velocity $u(t)$ and pitch angle $\theta(t)$:

$$\underline{x}(t) = \begin{bmatrix} u(t) \\ \theta(t) \end{bmatrix}$$

We are seeking the best estimate $\widehat{u}(t)$ of the actual velocity $u(t)$.

Matlab's hands-on: Velocity Aided Inertial Navigation

- Assuming no noise, give the expression of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ such that the state space equations corresponding to the *actual* values of velocity $u(t)$ and pitch angle $\theta(t)$ read:

$$\dot{\underline{x}}(t) = \boldsymbol{A}\,\underline{x}(t) + \boldsymbol{B}\,a_x(t)$$

- Let $\delta\underline{x}(t)$ be the error vector between the measured values (with $\tilde{}$) and the actual values (without $\tilde{}$):

$$\delta\underline{x}(t) = \widetilde{\underline{x}}(t) - \underline{x}(t) = \begin{bmatrix} \widetilde{u}(t) - u(t) \\ \widetilde{\theta}(t) - \theta(t) \end{bmatrix}$$

The linearized equation of navigation errors read:

$$\delta\dot{\underline{x}}(t) = \boldsymbol{A}\,\delta\underline{x}(t) + \underline{w}(t)$$

Give the expression of noise vector $\underline{w}(t)$

- Give the expression of the covariance matrix $\boldsymbol{P}_w$ of white noise $\underline{w}(t)$

Matlab's hands-on: Velocity Aided Inertial Navigation

- Let $\delta y(t)$ be defined as follows:

$$\delta y(t) = \widetilde{u}(t) - \widetilde{y}(t)$$

Give the expression of the output matrix $\boldsymbol{C}$ such that the output equation of the navigation errors reads:

$$\delta y(t) = \boldsymbol{C}\, \delta \underline{x}(t) - v(t)$$

- Let $\delta \hat{\underline{x}}(t)$ be defined as follows:

$$\delta \hat{\underline{x}}(t) = \widetilde{\underline{x}}(t) - \hat{\underline{x}}(t)$$

Give the expression of the Kalman-Bucy filter which will provide the optimum estimate $\hat{\underline{x}}(t)$ of the state vector as a function of the observer gain $\boldsymbol{L}$

Fundamentals of Linear Quadratic Regulator (LQR)
Robustness property of LQR
Design methods
Linear Quadratic Gaussian (LQG) control

Matlab's hands-on: Velocity Aided Inertial Navigation

- The observer gain $\boldsymbol{L}$ reads as follows:

$$\boldsymbol{L} = \boldsymbol{Y}\,\boldsymbol{C}^T\,\boldsymbol{P}_v^{-1}$$

where matrix $\boldsymbol{Y} = \boldsymbol{Y}^T \geq 0$ is the solution of the following Algebraic Riccati Equation (ARE):

$$0 = \boldsymbol{A}\boldsymbol{Y} + \boldsymbol{Y}\boldsymbol{A}^T - \boldsymbol{Y}\boldsymbol{C}^T\boldsymbol{P}_v^{-1}\boldsymbol{C}\boldsymbol{Y} + \boldsymbol{P}_w$$

Check that pair $(\boldsymbol{A},\ \boldsymbol{C})$ is observable and use Matlab to find the positive definite solution $\boldsymbol{Y}$ of this Algebraic Riccati Equation (ARE) with:

$$\begin{cases} P_{w_a} = 6 \cdot 10^{-4}\ (m/s^2)^2/Hertz \\ P_{w_g} = 3 \cdot 10^{-6}\ (rad/s)^2/Hertz \\ P_v = 2 \cdot 10^{-4}\ (m/s)^2/Hertz \end{cases}$$

Note: with Simulink, the `Sample time` of the `Band-Limited White Noise` box may be set to 0.01

Introduction to Optimal Control

Lecture notes

thierry.miquel@enac.fr

To cite this document:

T. Miquel, Introduction to Optimal Control. HAL Id: hal-02987731

December 1, 2022

Introduction

The application of optimal control theory to the practical design of multivariable control systems started in the 1960s[1]: in 1957 R. Bellman applied dynamic programming to the optimal control of discrete-time systems. His procedure resulted in nonlinear feedback schemes. By 1958, L.S. Pontryagin has developed the maximum principle relying on the calculus of variations developed by L. Euler (1707-1783). He solved the minimum-time problem, deriving in 1962 an on/off relay control law as an optimal control. In 1960 three major papers were published by R. Kalman and coworkers, working in the U.S. One of these publicized the vital work of Lyapunov (1857-1918) in the time-domain control of nonlinear systems. The next discussed the optimal control of systems, providing the design equations for the Linear Quadratic Regulator (LQR). The third paper has provided the design equations for the discrete Kalman filter. The continuous Kalman filter was developed by Kalman and Bucy in 1961.

In control theory, Kalman introduced linear algebra and matrices, so that systems with multiple inputs and outputs could easily be treated. He also formalized the notion of optimality in control theory by minimizing a very general quadratic generalized energy function. In the period of a year, the major limitations of classical control theory were overcome, important new theoretical tools were introduced, and a new era in control theory had begun; we call it the era of modern control. In the period since 1980 the theory has been further refined under the name of H_2 theory which is out of the scope of this survey.

This lecture focuses on LQ (linear quadratic) theory and is a compilation of a number of results in the context of control system design. This has been written thanks to the references put in bibliographical section. It starts with a reminder of the main results in optimization of non linear systems which will be used as a background for this lecture. Then linear quadratic regulator (LQR) for finite final time and for infinite final time where the solution to the LQ problem are discussed. The robustness properties of the linear quadratic regulator (LQR) are then presented where the asymptotic properties and the guaranteed gain and phase margins associated with the LQ solution are presented. The next section presents some design methods with a special emphasis on symmetric root locus. We conclude with a short section dedicated to the Linear Quadratic Tracker (LQT) where the usefulness of augmenting the plant with integrators is presented.

[1] https://lewisgroup.uta.edu/history.htm

Bibliography

[1] *Alazard D., Apkarian P. and Cumer C., Robustesse et commande optimale*, Cépaduès (2000)

[2] *Anderson B. and Moore J., Optimal Control: Linear Quadratic Methods*, Prentice Hall (1990)

[3] *Friedland B., Control System Design: An Introduction to State-Space Methods*, Dover Books on Electrical Engineering (2012)

[4] *Hespanha J. P., Linear Systems Theory*, Princeton University Press (2009)

[5] *Hull D. G., Optimal Control Theory for Applications*, Springer (2003)

[6] *Jeffrey B. Burl, Linear Optimal Control*, Prentice Hall (1998)

[7] *Lewis F., Vrabie D. and Syrmos L., Optimal Control*, John Wiley & Sons (2012)

[8] *Perry Y. Li, Linear Quadratic Optimal Control*, University of Minnesota (2012)

[9] *Sinha A., Linear Systems: Optimal and Robust Control*, CRC Press (2007)

[10] *Skogestad S., and Postlethwaite I., Multivariable Feedback Control: Analysis and Design*, John Wiley (2005)

Table of contents

1 Overview of Pontryagin's Minimum Principle **11**
1.1 Introduction . . . 11
1.2 Variation . . . 11
1.3 Example . . . 13
1.4 Lagrange multipliers . . . 14
1.5 Example . . . 15
1.6 Euler-Lagrange equation . . . 17
1.7 Fundamentals of optimal control theory . . . 20
1.7.1 Problem to be solved . . . 20
1.7.2 Bolza, Mayer and Lagrange problems . . . 20
1.7.3 First order necessary conditions . . . 21
1.8 Example: brachistochrone problem . . . 23
1.8.1 Problem overview . . . 23
1.8.2 System dynamics . . . 23
1.8.3 Euler-Lagrange approach . . . 25
1.8.4 Hamiltonian approach . . . 26
1.9 Hamilton-Jacobi-Bellman (HJB) equation . . . 28
1.9.1 Finite horizon control . . . 28
1.9.2 Principle of optimality, dynamic programming . . . 29
1.9.3 Infinite horizon control . . . 29
1.9.4 Application of HJB equation to linear time invariant systems 30
1.10 Pontryagin's principle . . . 31
1.11 Hamiltonian over time . . . 33
1.11.1 General result . . . 33
1.11.2 Autonomous system without constraint on input . . . 33
1.11.3 Free final time . . . 34
1.12 Bang-bang control . . . 35
1.12.1 Pontryagin's principle application . . . 35
1.12.2 Example 1 . . . 37
1.12.3 Example 2 . . . 40
1.13 Singular arc - Legendre-Clebsch condition . . . 42

2 Finite Horizon Linear Quadratic Regulator **43**
2.1 Problem to be solved . . . 43
2.2 Positive definite and positive semi-definite matrix . . . 44
2.3 Hamiltonian matrix . . . 45

2.4 Optimal control 46
2.4.1 State vector expression 46
2.4.2 Lagrange multipliers for imposed final state 47
2.4.3 Lagrange multipliers for weighted final state 47
2.4.4 Limit values when final state weighting matrix increases . 48
2.4.5 Closed-loop block diagram 49
2.5 Riccati differential equation 50
2.6 Examples 51
2.6.1 Example 1 51
2.6.2 Example 2 52
2.7 Second order necessary condition for optimality 53
2.8 Minimum cost achieved 53
2.9 Application to minimum energy control problem 54
2.9.1 Moving a linear system close to a final state with minimum energy 54
2.9.2 Moving a linear system exactly to a final state with minimum energy 56
2.9.3 Example 57
2.10 Finite horizon LQ regulator with cross-term in the performance index 59
2.11 Extension to nonlinear system affine in control 61

3 Infinite Horizon Linear Quadratic Regulator (LQR) 63
3.1 Problem to be solved 63
3.2 Stabilizability and detectability 64
3.3 Algebraic Riccati equation 64
3.4 Extension to nonlinear system affine in control 66
3.5 Solving the algebraic Riccati equation 68
3.5.1 Hamiltonian matrix based solution 68
3.5.2 Proof of the results on the Hamiltonian matrix 69
3.5.3 Solving general algebraic Riccati and Lyapunov equations 70
3.6 Application to the optimal control of any scalar LTI plant 72
3.7 Hamiltonian matrix properties 74
3.8 Discrete time LQ regulator 78
3.8.1 Finite horizon LQ regulator 78
3.8.2 Finite horizon LQ regulator with zero terminal state . . . 78
3.8.3 Infinite horizon LQ regulator 79
3.9 Robustness property 81
3.9.1 Hsu-Chen theorem 81
3.9.2 Generalized (MIMO) Nyquist stability criterion 83
3.9.3 Kalman equality 84
3.9.4 Robustness of Linear Quadratic Regulator 86

4 Design methods 91
4.1 Symmetric Root Locus 91
4.1.1 Characteristics polynomials 91
4.1.2 Root Locus reminder 92

4.1.3 Chang-Letov design procedure 94
4.1.4 Proof of the symmetric root locus result 96
4.2 Asymptotic properties of LQR applied to SISO plants 97
4.2.1 Closed-loop poles location 97
4.2.2 Shape of the magnitude of the loop gain 98
4.2.3 Weighting matrices selection 100
4.2.4 Poles assignment in optimal regulator using root locus . . 101
4.3 Poles shifting in optimal regulator 102
4.3.1 Mirror property 102
4.3.2 Reduced-order model 105
4.3.3 Shifting one real eigenvalue 105
4.3.4 Shifting a pair of complex conjugate eigenvalues 106
4.3.5 Sequential pole shifting via reduced-order models 107
4.4 Frequency domain approach 109
4.4.1 Non optimal pole assignment 109
4.4.2 Assignment of weighting matrices **Q** and **R** 110
4.5 Poles assignment in optimal regulator through matrix inequalities 113
4.6 Model matching 115
4.6.1 Cross-term in the performance index 115
4.6.2 Implicit reference model 116
4.7 Optimal output feedback 117
4.7.1 Reformulation of the state feedback optimal control problem 117
4.7.2 Output feedback optimal control problem 120
4.7.3 Solution of the output feedback optimal control problem . 121
4.7.4 Poles placement in a specified region 122
4.8 Frequency shaped LQ control 124
4.9 Optimal transient stabilization 126

5 Linear Quadratic Tracker (LQT) 129
5.1 Introduction 129
5.2 Control with feedforward 129
5.3 Finite horizon Linear Quadratic Tracker 130
5.4 Infinite horizon Linear Quadratic Tracker 133
5.4.1 General result 133
5.4.2 Asymptotically stable linear reference model 133
5.4.3 Constant reference tracking 134
5.5 Plant augmented with integrator 135
5.5.1 Integral augmentation 135
5.5.2 Proof of the cancellation of the steady-state error through integral augmentation 137
5.6 Tracking with prefilter 139
5.6.1 Tracking without integral augmentation 139
5.6.2 Tracking with integral augmentation 142

6 Linear Quadratic Gaussian (LQG) regulator 145
6.1 Introduction 145
6.2 Luenberger observer 146
6.3 White noise through Linear Time Invariant (LTI) system 147
6.3.1 Assumptions and definitions 147
6.3.2 Mean and covariance matrix of the state vector 148
6.3.3 Autocorrelation function of the stationary output vector . 149
6.3.4 Proof of the expression of the autocorrelation function . . 153
6.4 Kalman-Bucy filter 154
6.4.1 Linear Quadratic Estimator 154
6.4.2 Sketch of the proof 156
6.5 Duality principle 157
6.6 Separation principle 158
6.7 Controller transfer function 160
6.8 Loop Transfer Recovery 162
6.8.1 Lack of guaranteed robustness of LQG design 162
6.8.2 Doyle's seminal example 162
6.8.3 Closed-loop eigenvalues and eigenvectors 164
6.8.4 Asymptotic behavior of Riccati equation 165
6.8.5 Loop Transfer Recovery (LTR) design 166
6.9 Proof of the Loop Transfer Recovery condition 169
6.9.1 Loop transfer function with observer 169
6.9.2 Loop Transfer Recovery (LTR) condition 171
6.9.3 Setting the Loop Transfer Recovery design parameter . . 173
6.10 Robust control design 174
6.11 Sensor data fusion 176
6.11.1 Complementary filter 176
6.11.2 Kalman filter 178
6.11.3 Relation between complementary filter and Kalman filter 179
6.12 Euler angles estimation 181
6.12.1 One dimensional attitude estimation 181
6.12.2 One dimensional complementary filter 182
6.12.3 Direct Cosine Matrix (DCM) and kinematic relations . . . 184
6.12.4 Roll and pitch angles estimation from accelerometer measurements 185
6.12.5 Yaw angle estimation from magnetometer measurements . 187
6.12.6 Angular velocity from gyroscope measurements 187
6.12.7 Attitude and Heading Reference System (AHRS) based on complementary filter 187

Chapter 1

Overview of Pontryagin's Minimum Principle

1.1 Introduction

Pontryagin's Minimum (or Maximum) Principle was formulated in 1956 by the Russian mathematician Lev Pontryagin (1908 - 1988) and his students[1]. Its initial application was dedicated to the maximization of the terminal speed of a rocket. The result was derived using ideas from the classical calculus of variations.

This chapter is devoted to the main results of optimal control theory which leads to conditions for optimality.

1.2 Variation

Optimization can be accomplished by using a generalization of the differential called variation.

Let's consider the real scalar cost function $J(\underline{x})$ of a vector $\underline{x} \in \mathbb{R}^n$. Cost function $J(\underline{x})$ has a local minimum at $\underline{x}^*$ if and only if for all $\delta\underline{x}$ sufficiently small;

$$J(\underline{x}^* + \delta\underline{x}) \geq J(\underline{x}^*) \tag{1.1}$$

An equivalent statement statement is that:

$$\Delta J(\underline{x}^*, \delta\underline{x}) = J(\underline{x}^* + \delta\underline{x}) - J(\underline{x}^*) \geq 0 \tag{1.2}$$

c The term $\Delta J(\underline{x}^*, \delta\underline{x})$ is called the increment of $J(\underline{x})$. The optimality condition can be found by expanding $J(\underline{x}^* + \delta\underline{x})$ in a Taylor series around the extremun point $\underline{x}^*$. When $J(\underline{x})$ is a scalar function of multiple variables, the expansion of $J(\underline{x})$ in the Taylor series involves the gradient and the Hessian of the cost function $J(\underline{x})$:

– Assuming that $J(\underline{x})$ is a differentiable function, the term $\frac{dJ(\underline{x}^*)}{d\underline{x}}$ is the gradient of $J(\underline{x})$ at $\underline{x}^* \in \mathbb{R}^n$ which is the vector of $\mathbb{R}^n$ defined by:

[1] https://en.wikipedia.org/wiki/Pontryagin's_maximum_principle

$$\frac{dJ(\underline{x}^*)}{d\underline{x}} = \nabla J(\underline{x}^*) = \begin{bmatrix} \frac{dJ(\underline{x})}{dx_1} \\ \vdots \\ \frac{dJ(\underline{x})}{dx_n} \end{bmatrix}_{\underline{x}=\underline{x}^*} \tag{1.3}$$

– Assuming that $J(\underline{x})$ is a twice differentiable function, the term $\frac{d^2J(\underline{x}^*)}{d\underline{x}^2}$ is the Hessian of $J(\underline{x})$ at $\underline{x}^* \in \mathbb{R}^n$ which is the symmetric $n \times n$ matrix defined by:

$$\frac{d^2J(\underline{x}^*)}{d\underline{x}^2} = \nabla^2 J(\underline{x}^*) = \begin{bmatrix} \frac{\partial^2 J(\underline{x})}{\partial x_1 \partial x_1} & \cdots & \frac{\partial^2 J(\underline{x})}{\partial x_1 \partial x_n} \\ \vdots & & \\ \frac{\partial^2 J(\underline{x})}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 J(x)}{\partial x_n \partial x_n} \end{bmatrix}_{x=x^*} = \left[\frac{d^2J(x^*)}{dx_i dx_j}\right]_{1\leq i,j \leq n} \tag{1.4}$$

Expanding $J(\underline{x}^* + \delta\underline{x})$ in a Taylor series around the point $\underline{x}^*$ leads to the following expression, where HOT stands for *Higher-Order Terms*:

$$J(\underline{x}^* + \delta\underline{x}) = J(\underline{x}^*) + \delta\underline{x}^T \nabla J(\underline{x}^*) + \frac{1}{2}\delta\underline{x}^T \nabla^2 J(\underline{x}^*)\delta\underline{x} + HOT \tag{1.5}$$

Thus:

$$\begin{array}{rcl} \Delta J(\underline{x}^*, \delta\underline{x}) & = & J(\underline{x}^* + \delta\underline{x}) - J(\underline{x}^*) \\ & = & \delta\underline{x}^T \nabla J(\underline{x}^*) + \frac{1}{2}\delta\underline{x}^T \nabla^2 J(\underline{x}^*)\delta\underline{x} + HOT \end{array} \tag{1.6}$$

When dealing with a functional (a real scalar function of functions) $\delta\underline{x}$ is called the variation of $\underline{x}$ and the term in the increment $\Delta J(\underline{x}^*, \delta\underline{x})$ which is linear in $\delta\underline{x}^T$ is called the variation of J and is denoted $\delta J(\underline{x}^*)$. The variation of $J(\underline{x})$ is a generalization of the differential and can be applied to the optimization of a functional. Equation (1.6) can be used to develop necessary conditions for optimality. Indeed as $\delta\underline{x}$ approaches zero the terms $\delta\underline{x}^T\delta\underline{x}$ as well as HOT become arbitrarily small compared to $\delta\underline{x}$. As a consequence, *a necessary condition for $\underline{x}^*$ to be a local extremum of the cost function J is that the first variation of J (its gradient) at $\underline{x}^*$ is zero:*

$$\delta J(\underline{x}^*) = \nabla J(\underline{x}^*) = 0 \tag{1.7}$$

A critical (or stationary) point $\underline{x}^*$ is a point where $\delta J(\underline{x}^*) = \nabla J(\underline{x}^*) = 0$. Furthermore the sign of the Hessian provides sufficient condition for a local extremum. Let's write the Hessian $\nabla^2 J(\underline{x}^*)$ at the critical point $\underline{x}^*$ as follows:

$$\nabla^2 J(\underline{x}^*) = \begin{bmatrix} h_{11} & \cdots & h_{1n} \\ \vdots & & \vdots \\ h_{n1} & \cdots & h_{nn} \end{bmatrix} \tag{1.8}$$

– The sufficient condition for the critical point $\underline{x}^*$ to be a local minimum is that the Hessian is positive definite, that is that all the principal minor determinants are positive:

$$\Leftrightarrow \begin{cases} \forall\ 1 \le k \le n\ H_k > 0 \\ H_1 = h_{11} > 0 \\ H_2 = \begin{vmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{vmatrix} > 0 \\ H_3 = \begin{vmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{vmatrix} > 0 \\ \text{and so on...} \end{cases} \tag{1.9}$$

– The sufficient condition for the critical point $\underline{x}^*$ to be a local maximum is that the Hessian is negative definite, or equivalently that the opposite of the Hessian is positive definite:

$$\Leftrightarrow \begin{cases} \forall\ 1 \le k \le n\ (-1)^k H_k > 0 \\ H_1 = h_{11} < 0 \\ H_2 = \begin{vmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{vmatrix} > 0 \\ H_3 = \begin{vmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{vmatrix} < 0 \\ \text{and so on...} \end{cases} \tag{1.10}$$

– If the Hessian has both positive and negative eigenvalues then the critical point $\underline{x}^*$ is a saddle point for the cost function $J(\underline{x})$.

It should be emphasized that if the Hessian is positive semi-definite or negative semi-definite or has null eigenvalues at a critical point $\underline{x}^*$, then it cannot be concluded that the critical point is a minimizer or a maximizer or a saddle point of the cost function $J(\underline{x})$ and the test is inconclusive.

1.3 Example

Find the local maxima/minima for the following cost function:

$$J(\underline{x}) = 5 - (x_1 - 2)^2 - 2(x_2 - 1)^2 \tag{1.11}$$

First let's compute the first variation of J, or equivalently its gradient:

$$\frac{dJ(\underline{x})}{d\underline{x}} = \nabla J(\underline{x}) = \begin{bmatrix} \frac{dJ(\underline{x})}{dx_1} \\ \frac{dJ(\underline{x})}{dx_2} \end{bmatrix} = \begin{bmatrix} -2(x_1 - 2) \\ -4(x_2 - 1) \end{bmatrix} \tag{1.12}$$

A necessary condition for $\underline{x}^*$ to be a local extremum is that the first variation of J at $\underline{x}^*$ is zero for all $\delta\underline{x}$:

$$\delta J(\underline{x}^*) = \nabla J(\underline{x}^*) = 0 \tag{1.13}$$

As a consequence, the following point is a critical point:

$$\underline{x}^* = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \tag{1.14}$$

Now, we compute the Hessian to conclude on the nature of this critical point:

$$\nabla^2 J(\underline{x}^*) = \begin{bmatrix} -2 & 0 \\ 0 & -4 \end{bmatrix} \tag{1.15}$$

As far as the Hessian is negative definite we conclude that the critical point $\underline{x}^*$ is a local maximum.

1.4 Lagrange multipliers

Optimal control problems which will be tackled involve minimization of a cost function subject to constraints on the state vector and the control. The necessary condition given above is only applicable to unconstrained minimization problems; Lagrange multipliers provide a method of converting a constrained minimization problem into an unconstrained minimization problem of higher order. Optimization can then be performed using the above necessary condition. A constrained optimization problem is a problem of the form:

Maximize (or minimize) cost function $J(\underline{x})$ subject to the condition $g(\underline{x}) = 0$

The most popular technique to solve this constrained optimization problem is to use the Lagrange multiplier technique. Necessary condition for optimality of J at a point $\underline{x}^*$ are that $\underline{x}^*$ satisfies $g(\underline{x}) = 0$ and that the gradient of J is zero in all direction along the surface $g(\underline{x}) = 0$; this condition is satisfied if the gradient of J is normal to the surface at $\underline{x}^*$. As far as the gradient of $g(\underline{x})$ is normal to the surface, including $\underline{x}^*$, this condition is satisfied if the gradient of J is parallel (that is proportional) to the gradient of $g(\underline{x})$ at $\underline{x}^*$, or equivalently:

$$\left. \left(\frac{\partial J(\underline{x})}{\partial \underline{x}} + \underline{\lambda}^T \frac{\partial g(\underline{x})}{\partial \underline{x}} \right) \right|_{\underline{x}=\underline{x}^*} = 0 \tag{1.16}$$

It is worth noticing that the following relations hold:

$$\begin{cases} J = a + \underline{b}^T \underline{x} = a + \underline{x}^T \underline{b} \Rightarrow \dfrac{\partial J}{\partial \underline{x}} = \begin{bmatrix} \frac{\partial J}{\partial x_1} \\ \frac{\partial J}{\partial x_2} \\ \vdots \\ \frac{\partial J}{\partial x_n} \end{bmatrix} = \underline{b} \\ J = a + \dfrac{\underline{b}^T \underline{x}}{\|\underline{x}\|^n} \Rightarrow \dfrac{\partial J}{\partial \underline{x}} = \begin{bmatrix} \frac{\partial J}{\partial x_1} \\ \frac{\partial J}{\partial x_2} \\ \vdots \\ \frac{\partial J}{\partial x_n} \end{bmatrix} = \dfrac{\underline{b}}{\|\underline{x}\|^n} - \dfrac{n}{\|\underline{x}\|^{n+2}} \left(\underline{x}^T \underline{b} \right) \underline{x} \end{cases} \tag{1.17}$$

As an illustration, consider the cost function $J(\underline{x}) = (x_1 - 1)^2 + (x_2 - 2)^2$: this is the equation of a circle of center $(1, 2)$ with radius $J(\underline{x})$. It is clear that $J(\underline{x})$ is minimal when (x_1, x_2) is situated on the center of the circle. In this case $J(\underline{x})^* = 0$. Nevertheless if we impose on (x_1, x_2) to belong to the straight line defined by $x_2 - 2x_1 - 6 = 0$ then $J(\underline{x})$ will be minimized as soon as the circle of radius $J(\underline{x})$ tangent the straight line, that is if the gradient of $J(\underline{x})$ is normal to the surface at $\underline{x}^*$. Parameter $\underline{\lambda}$ is called the Lagrange multiplier and has the dimension of the number of constraints expressed through $g(\underline{x})$. The necessary condition for optimality can be obtained as the solution of the following unconstrained optimization problem where $L(\underline{x}, \underline{\lambda})$ is the Lagrange function:

$$L(\underline{x}, \underline{\lambda}) = J(\underline{x}) + \underline{\lambda}^T g(\underline{x}) \tag{1.18}$$

Setting to zero the gradient of the Lagrange function with respect to $\underline{x}$ leads to (1.16) whereas setting to zero the derivative of the Lagrange function with respect to the Lagrange multiplier $\underline{\lambda}$ leads to the constraint $g(\underline{x}) = 0$. As a consequence, *a necessary condition for $\underline{x}^*$ to be a local extremum of the cost function J subject to the constraint $g(\underline{x}) = 0$ is that the first variation of Lagrange function (its gradient) at $\underline{x}^*$ is zero*:

$$\left.\frac{\partial L(\underline{x}, \underline{\lambda})}{\partial \underline{x}}\right|_{\underline{x}=\underline{x}^*} = 0 \Leftrightarrow \left.\left(\frac{\partial J(\underline{x})}{\partial \underline{x}} + \underline{\lambda}^T \frac{\partial g(\underline{x})}{\partial \underline{x}}\right)\right|_{\underline{x}=\underline{x}^*} = 0 \tag{1.19}$$

The bordered Hessian is the $(n+m) \times (n+m)$ symmetric matrix which is used for the second-derivative test. If there are m constraints represented by $g(\underline{x}) = 0$, then there are m border rows at the top-right and m border columns at the bottom-left (the transpose of the top-right matrix) and the zero in the south-east corner of the bordered Hessian is an $m \times m$ block of zeros, represented by $\mathbf{0}_{m\times m}$. The bordered Hessian $\mathbf{H}_b(p)$ is defined by:

$$\mathbf{H}_b(p) = \left.\begin{bmatrix} \frac{\partial^2 L(\underline{x})}{\partial x_1 \partial x_1} - p & \frac{\partial^2 L(\underline{x})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 L(\underline{x})}{\partial x_1 \partial x_n} & \frac{\partial g_1(\underline{x})}{\partial x_1} & \cdots & \frac{\partial g_m(\underline{x})}{\partial x_1} \\ \frac{\partial^2 L(\underline{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 L(\underline{x})}{\partial x_2 \partial x_2} - p & \cdots & \frac{\partial^2 L(\underline{x})}{\partial x_2 \partial x_n} & \frac{\partial g_1(\underline{x})}{\partial x_2} & \cdots & \frac{\partial g_m(\underline{x})}{\partial x_2} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 L(\underline{x})}{\partial x_n \partial x_1} & \frac{\partial^2 L(\underline{x})}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 L(\underline{x})}{\partial x_n \partial x_n} - p & \frac{\partial g_1(\underline{x})}{\partial x_n} & \cdots & \frac{\partial g_m(\underline{x})}{\partial x_n} \\ \frac{\partial g_1(\underline{x})}{\partial x_1} & \cdots & \cdots & \frac{\partial g_1(\underline{x})}{\partial x_n} & & & \\ \cdots & \cdots & \cdots & & & 0_{m\times m} & \\ \frac{\partial g_m(\underline{x})}{\partial x_1} & \cdots & \cdots & \frac{\partial g_m(\underline{x})}{\partial x_n} & & & \end{bmatrix}\right|_{\underline{x}=\underline{x}^*} \tag{1.20}$$

The sufficient condition for the critical point $\underline{x}^*$ to be an extrema is that the values of p obtained from $\det(\mathbf{H}_b(p)) = 0$ must be of the same sign.

- If all the values of p are strictly negative, then it is a maxima
- If all the values of p are strictly positive, then it is a minima
- However if some values of p are zero or of a different sign, then the critical point $\underline{x}^*$ is a saddle point.

1.5 Example

Find the local maxima/minima for the following cost function:

$$J(\underline{x}) = x_1 + 3x_2 \tag{1.21}$$

Subject to the constraint:

$$g(\underline{x}) = x_1^2 + x_2^2 - 10 = 0 \tag{1.22}$$

First let's compute the Lagrange function of this problem:

$$L(\underline{x}, \underline{\lambda}) = J(\underline{x}) + \underline{\lambda}^T g(\underline{x}) = x_1 + 3x_2 + \lambda(x_1^2 + x_2^2 - 10) \tag{1.23}$$

A necessary condition for $\underline{x}^*$ to be a local extremum is that the first variation of J at $\underline{x}^*$ is zero for all $\delta\underline{x}$:

$$\frac{\partial L(\underline{x}^*)}{\partial \underline{x}} = \begin{bmatrix} 1 + 2\lambda x_1 \\ 3 + 2\lambda x_2 \end{bmatrix} = 0 \text{ s.t. } x_1^2 + x_2^2 - 10 = 0 \tag{1.24}$$

As a consequence, the Lagrange multiplier λ shall be chosen as follows:

$$\begin{bmatrix} x_1 = -\frac{1}{2\lambda} \\ x_2 = -\frac{3}{2\lambda} \end{bmatrix} \Rightarrow x_1^2 + x_2^2 - 10 = \frac{1}{4\lambda^2} + \frac{9}{4\lambda^2} - 10 = 0 \Leftrightarrow 10 - 40\lambda^2 = 0 \Leftrightarrow \lambda = \pm\frac{1}{2} \tag{1.25}$$

Using the values of the Lagrange multiplier within (1.24) we then obtain 2 critical points:

$$\lambda = \frac{1}{2} \Rightarrow x_1^* = \begin{bmatrix} -1 \\ -3 \end{bmatrix} \text{ and } \lambda = -\frac{1}{2} \Rightarrow x_2^* = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \tag{1.26}$$

– For $\lambda = \frac{1}{2}$ the bordered Hessian is:

$$\mathbf{H}_b(p) = \begin{bmatrix} 2\lambda - p & 0 & 2x_1 \\ 0 & 2\lambda - p & 2x_2 \\ 2x_1 & 2x_2 & 0 \end{bmatrix}_{\underline{x}=\underline{x}^*} = \begin{bmatrix} 1-p & 0 & -2 \\ 0 & 1-p & -6 \\ -2 & -6 & 0 \end{bmatrix} \tag{1.27}$$

Thus:

$$\det(\mathbf{H}_b(p)) = -40 + 40p \tag{1.28}$$

We conclude that the critical point $(-1; -3)$ is a local minima because $\det(\mathbf{H}_b(p)) = 0$ for $p = +1$ which is strictly positive.

– For $\lambda = -\frac{1}{2}$ the bordered Hessian is:

$$\mathbf{H}_b(p) = \begin{bmatrix} 2\lambda - p & 0 & 2x_1 \\ 0 & 2\lambda - p & 2x_2 \\ 2x_1 & 2x_2 & 0 \end{bmatrix}_{\underline{x}=\underline{x}^*} = \begin{bmatrix} -1-p & 0 & 2 \\ 0 & -1-p & 6 \\ 2 & 6 & 0 \end{bmatrix} \tag{1.29}$$

Thus:

$$\det(\mathbf{H}_b(p)) = 40 + 40p \tag{1.30}$$

We conclude that the critical point $(+1; +3)$ is a local maxima because $\det(\mathbf{H}_b(p)) = 0$ for $p = -1$ which is strictly negative.

1.6 Euler-Lagrange equation

Historically, Euler-Lagrange equation came with the study of the tautochrone (or isochrone curve) problem. Lagrange solved this problem in 1755 and sent the solution to Euler. Their correspondence ultimately led to the calculus of variations [2].

The problem considered was to find the expression of $\underline{x}(t)$ which minimizes the following performance index $J(\underline{x}(t))$ where $F(\underline{x}(t), \dot{\underline{x}}(t))$ is a real-valued twice continuous function:

$$J(\underline{x}(t)) = \int_0^{t_f} F(\underline{x}(t), \dot{\underline{x}}(t))\, dt \tag{1.31}$$

Furthermore the initial and final values of $\underline{x}(t)$ are imposed:

$$\begin{cases} \underline{x}(0) = \underline{x}_0 \\ \underline{x}(t_f) = \underline{x}_f \end{cases} \tag{1.32}$$

Let $\underline{x}^*(t)$ be a candidate for the minimization of $J(\underline{x}(t))$. In order to see whether $\underline{x}^*(t)$ is indeed an optimal solution, this candidate optimal input is perturbed by a small amount $\delta\underline{x}$ which leads to a perturbation $\delta\underline{x}$ in the optimal state vector $\underline{x}^*(t)$:

$$\begin{cases} \underline{x}(t) = \underline{x}^*(t) + \delta\underline{x}(t) \\ \dot{\underline{x}}(t) = \dot{\underline{x}}^*(t) + \delta\dot{\underline{x}}(t) \end{cases} \tag{1.33}$$

The change δJ in the value of the performance index is obtained thanks to the calculus of variation:

$$\delta J = \int_0^{t_f} \delta F(\underline{x}(t), \dot{\underline{x}}(t))\, dt = \int_0^{t_f} \left(\frac{\partial F}{\partial \underline{x}}^T \delta\underline{x} + \frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\dot{\underline{x}} \right) dt \tag{1.34}$$

Integrating $\frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\dot{\underline{x}}$ by parts leads to the following expression:

$$\begin{array}{l} \frac{d}{dt}\left(\frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\underline{x} \right) = \frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\underline{x} + \frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\dot{\underline{x}} \\ \Rightarrow \delta J = \int_0^{t_f} \left(\frac{\partial F}{\partial \underline{x}}^T \delta\underline{x} - \frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\underline{x} \right) dt + \left. \frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\underline{x} \right|_0^{t_f} \end{array} \tag{1.35}$$

Because $\delta\underline{x}$ is a perturbation around the optimal state vector $\underline{x}^*(t)$ we shall set to zero the first variation δJ whatever the value of the variation $\delta\underline{x}$:

$$\delta J = 0 \ \forall\ \delta\underline{x} \tag{1.36}$$

This leads to the following necessary conditions for optimality:

$$\begin{cases} \frac{\partial F}{\partial \underline{x}}^T \delta\underline{x} - \frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\underline{x} = 0 \\ \left. \frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\underline{x} \right|_0^{t_f} = 0 \end{cases} \tag{1.37}$$

[2] https://en.wikipedia.org/wiki/Euler-Lagrange_equation

As far as the initial and final values of $\underline{x}(t)$ are imposed no variation are permitted on $\delta\underline{x}$:

$$\left\{ \begin{array}{l} \underline{x}(0) = \underline{x}_0 \\ \underline{x}(t_f) = \underline{x}_f \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \delta\underline{x}(0) = 0 \\ \delta\underline{x}(t_f) = 0 \end{array} \right. \tag{1.38}$$

On the other hand it is worth noticing that if the final value was not imposed we shall have $\left.\frac{\partial F}{\partial \dot{\underline{x}}}\right|_{t=t_f} = 0$.

Thus the first variation δJ of the functional cost reads:

$$\delta J = \int_0^{t_f} \left(\frac{\partial F}{\partial \underline{x}}^T \delta\underline{x} - \frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}}^T \delta\underline{x} \right) dt \tag{1.39}$$

In order to set to zero the first variation δJ whatever the value of the variation $\delta\underline{x}$ the following second-order partial differential equation has to be solved:

$$\frac{\partial F}{\partial \underline{x}}^T - \frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}}^T = 0 \tag{1.40}$$

Or by taking the transpose:

$$\frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}} - \frac{\partial F}{\partial \underline{x}} = 0 \tag{1.41}$$

We retrieve the well known Euler-Lagrange equation of classical mechanics.

Euler-Lagrange equation is a second order Ordinary Differential Equations (ODE) that $\underline{x}$ shall satisfy to minimize $\int_0^{t_f} F(\underline{x}(t), \dot{\underline{x}}(t))\, dt$. Euler-Lagrange equation is usually quite difficult to solve.

Nevertheless, because $F(\underline{x}(t), \dot{\underline{x}}(t))$ does not depends explicitly on time t, Beltrami identity[3] provides a first integral of the Euler-Lagrange equation. Denoting by C a constant, the first integral of the Euler-Lagrange equation reads as follows:

$$\frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}} - \frac{\partial F}{\partial \underline{x}} = \underline{0} \Leftrightarrow F - \frac{\partial F}{\partial \dot{\underline{x}}}^T \dot{\underline{x}} = C \tag{1.42}$$

Indeed, multiplying both sides of the Euler-Lagrange equation by $\dot{\underline{x}}^T$ we get:

$$\frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}} - \frac{\partial F}{\partial \underline{x}} = \underline{0} \Rightarrow \dot{\underline{x}}^T \left(\frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}} \right) - \dot{\underline{x}}^T \frac{\partial F}{\partial \underline{x}} = 0 \tag{1.43}$$

Since $F(\underline{x}(t), \dot{\underline{x}}(t))$ does not depend explicitly on time t, we have:

$$\begin{array}{rl} \frac{dF(\underline{x}(t),\dot{\underline{x}}(t))}{dt} & = \frac{\partial F}{\partial \underline{x}}^T \frac{\partial \underline{x}}{\partial t} + \frac{\partial F}{\partial \dot{\underline{x}}}^T \frac{\partial \dot{\underline{x}}}{\partial t} \\ & = \dot{\underline{x}}^T \frac{\partial F}{\partial \underline{x}} + \frac{\partial F}{\partial \dot{\underline{x}}}^T \frac{\partial \dot{\underline{x}}}{\partial t} \\ \Rightarrow \dot{\underline{x}}^T \frac{\partial F}{\partial \underline{x}} & = \frac{dF}{dt} - \frac{\partial F}{\partial \dot{\underline{x}}}^T \frac{\partial \dot{\underline{x}}}{\partial t} \end{array} \tag{1.44}$$

Using this expression of $\dot{\underline{x}}^T \frac{\partial F}{\partial \underline{x}}$ in (1.43) reads:

$$\begin{array}{l} \dot{\underline{x}}^T \left(\frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}} \right) - \left(\frac{dF}{dt} - \frac{\partial F}{\partial \dot{\underline{x}}}^T \frac{\partial \dot{\underline{x}}}{\partial t} \right) = 0 \\ \Leftrightarrow \left(\frac{d}{dt}\frac{\partial F}{\partial \dot{\underline{x}}} \right)^T \dot{\underline{x}} + \frac{\partial F}{\partial \dot{\underline{x}}}^T \frac{\partial \dot{\underline{x}}}{\partial t} - \frac{dF}{dt} = 0 \\ \Leftrightarrow \frac{d}{dt} \left(\frac{\partial F}{\partial \dot{\underline{x}}}^T \dot{\underline{x}} - F \right) = 0 \end{array} \tag{1.45}$$

[3]https://en.wikipedia.org/wiki/Beltrami_identity

Denoting by C a constant, the first integral of the Euler-Lagrange equation finally reads as the Beltrami identity:

$$F - \frac{\partial F}{\partial \underline{\dot{x}}}^T \underline{\dot{x}} = C \tag{1.46}$$

Alternatively, Euler-Lagrange equation could be transformed into a set of first order Ordinary Differential Equations, which may be more convenient to manipulate, by introducing a control $\underline{u}(t)$ defined by $\underline{\dot{x}}(t) = \underline{u}(t)$ and by using the Hamiltonian function $\mathcal{H}$ as it will be seen in the next sections.

Example 1.1. *Let's find the shortest distance between two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ in the euclidean plane.*

The length of the path between the two points is defined by:

$$J(y(x)) = \int_{P_1}^{P_2} \sqrt{dx^2 + dy^2} = \int_{x_1}^{x_2} \sqrt{1 + (y'(x))^2}\, dx \tag{1.47}$$

For that example $F(y(x), y'(x))$ reads:

$$F\left(y(x), y'(x)\right) = \sqrt{1 + \left(\frac{dy(x)}{dx}\right)^2} = \sqrt{1 + (y'(x))^2} \tag{1.48}$$

The initial and final values on $y(x)$ are imposed as follows:

$$\begin{cases} y(x_1) = y_1 \\ y(x_2) = y_2 \end{cases} \tag{1.49}$$

The Euler-Lagrange equation for this example reads:

$$\frac{d}{dx}\frac{\partial F}{\partial y'} - \frac{\partial F}{\partial y} = 0 \Leftrightarrow \frac{d}{dx}\frac{y'(x)}{\sqrt{1 + (y'(x))^2}} = 0 \tag{1.50}$$

From the preceding relation it is clear that, denoting by c_1 a constant, $y'(x)$ shall satisfy the following first order differential equation:

$$\begin{array}{l} \frac{y'(x)}{\sqrt{1+(y'(x))^2}} = c_1 \Rightarrow (y'(x))^2 = c_1^2 \left(1 + (y'(x))^2\right) \\ \Rightarrow (y'(x))^2 = \frac{c_1^2}{1-c_1^2} \Rightarrow y'(x) = a = constant \end{array} \tag{1.51}$$

Alternatively, Beltrami identity reads as follows:

$$\begin{cases} F(y, y') = \sqrt{1 + (y')^2} \\ F - \frac{\partial F}{\partial y'} y' = C \end{cases} \Rightarrow \begin{cases} \sqrt{1 + (y')^2} - \frac{(y')^2}{\sqrt{1+(y')^2}} = C \\ \Rightarrow 1 = C\sqrt{1 + (y')^2} \\ \Rightarrow y'(x) = a = constant \end{cases} \tag{1.52}$$

Thus, the shortest distance between two fixed points in the euclidean plane is a curve with constant slope, that is a straight-line:

$$y(x) = a\,x + b \tag{1.53}$$

With initial and final values imposed on $y(x)$ we finally get for $y(x)$ the Lagrange polynomial of degree 1:

$$\begin{cases} y(x_1) = y_1 \\ y(x_2) = y_2 \end{cases} \Rightarrow y(x) = y_1 \frac{x - x_2}{x_1 - x_2} + y_2 \frac{x - x_1}{x_2 - x_1} \tag{1.54}$$

■

1.7 Fundamentals of optimal control theory

1.7.1 Problem to be solved

We first consider optimal control problems for general nonlinear *time invariant* systems of the form:

$$\begin{cases} \underline{\dot{x}} = f(\underline{x}, \underline{u}) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{1.55}$$

Where $\underline{x} \in \mathbb{R}^n$ and $\underline{u} \in \mathbb{R}^m$ are the state variable and control inputs, respectively, and $f(\underline{x}, \underline{u})$ is a continuous nonlinear function and $\underline{x}_0$ the initial conditions. The goal is to find a control $\underline{u}$ that *minimizes* the following performance index:

$$J(\underline{u}(t)) = G\left(\underline{x}(t_f)\right) + \int_0^{t_f} F(\underline{x}(t), \underline{u}(t))\, dt \tag{1.56}$$

Where:

- t is the current time and t_f the final time;
- $J(\underline{u}(t))$ is the integral cost function;
- $F(\underline{x}(t), \underline{u}(t))$ is the scalar running cost function;
- $G\left(\underline{x}(t_f)\right)$ is the scalar terminal cost function.

Note that the state equation serves as constraints for the optimization of the performance index $J(\underline{u}(t))$. In addition, notice that the use of function $G\left(\underline{x}(t_f)\right)$ is optional; indeed, if the final state $\underline{x}(t_f)$ is imposed then there is no need to insert the expression $G\left(\underline{x}(t_f)\right)$ in the cost to be minimized.

1.7.2 Bolza, Mayer and Lagrange problems

The problem defined above is known as the Bolza problem. In the special case where $F(\underline{x}(t), \underline{u}(t)) = 0$ then the problem is known as the Mayer problem; on the other hand if $G\left(\underline{x}(t_f)\right) = 0$ the problem is known as the Lagrange problem.

The Bolza problem is equivalent to the Lagrange problem and in fact leads to it with the following change of variable:

$$\begin{cases} J_1(\underline{u}(t)) = \int_0^{t_f} \left(F(\underline{x}(t), \underline{u}(t)) + x_{n+1}(t)\right)\, dt \\ \dot{x}_{n+1}(t) = 0 \\ x_{n+1} = \frac{G\left(\underline{x}(t_f)\right)}{t_f}\ \forall t \end{cases} \tag{1.57}$$

It also leads to the Mayer problem if one sets:

$$\left\{\begin{array}{l} J_2(\underline{u}(t)) = G\left(\underline{x}(t_f)\right) + x_0(t_f) \\ \dot{x}_0(t) = F(\underline{x}(t), \underline{u}(t)) \\ x_0(0) = 0 \end{array}\right. \tag{1.58}$$

1.7.3 First order necessary conditions

The optimal control problem is then a constrained optimization problem, with cost being a functional of $\underline{u}(t)$ and the state equation providing the constraint equations. This optimal control problem can be converted to an unconstrained optimization problem of higher dimension by the use of Lagrange multipliers. An augmented performance index is then constructed by adding a vector of Lagrange multipliers $\underline{\lambda}$ times each constraint imposed by the differential equations driving the dynamics of the plant; these constraints are added to the performance index by the addition of an integral to form the augmented performance index J_a:

$$J_a(\underline{u}(t)) = G\left(\underline{x}(t_f)\right) + \int_0^{t_f} \left(F(\underline{x}(t), \underline{u}(t)) + \underline{\lambda}^T(t)(f(\underline{x}, \underline{u}) - \dot{\underline{x}})\right) dt \tag{1.59}$$

Let $\underline{u}^*(t)$ be a candidate for the optimal input vector and let the corresponding state vector be $\underline{x}^*(t)$:

$$\dot{\underline{x}}^*(t) = f(\underline{x}^*(t), \underline{u}^*(t)) \tag{1.60}$$

In order to see whether $\underline{u}^*(t)$ is indeed an optimal solution, this candidate optimal input is perturbed by a small amount $\delta\underline{u}$ which leads to a perturbation $\delta\underline{x}$ in the optimal state vector $\underline{x}^*(t)$:

$$\left\{\begin{array}{l} \underline{u}(t) = \underline{u}^*(t) + \delta\underline{u}(t) \\ \underline{x}(t) = \underline{x}^*(t) + \delta\underline{x}(t) \end{array}\right. \tag{1.61}$$

Assuming that the final time t_f is known, the change δJ_a in the value of the augmented performance index is obtained thanks to the calculus of variation [4]:

$$\begin{array}{rl} \delta J_a & = \frac{\partial G\left(\underline{x}(t_f)\right)}{\partial \underline{x}(t_f)}^T \delta\underline{x}(t_f) + \\ & \int_0^{t_f} \left(\frac{\partial F}{\partial \underline{x}}^T \delta\underline{x} + \frac{\partial F}{\partial \underline{u}}^T \delta\underline{u} + \underline{\lambda}^T(t)\left(\frac{\partial f}{\partial \underline{x}}\delta\underline{x} + \frac{\partial f}{\partial \underline{u}}\delta\underline{u} - \frac{d\delta\underline{x}}{dt}\right)\right) dt \\ & = \frac{\partial G\left(\underline{x}(t_f)\right)}{\partial \underline{x}(t_f)}^T \delta\underline{x}(t_f) + \\ & \int_0^{t_f} \left(\left(\frac{\partial F}{\partial \underline{x}}^T + \underline{\lambda}^T(t)\frac{\partial f}{\partial \underline{x}}\right)\delta\underline{x} + \left(\frac{\partial F}{\partial \underline{u}}^T + \underline{\lambda}^T(t)\frac{\partial f}{\partial \underline{u}}\right)\delta\underline{u} - \underline{\lambda}^T(t)\frac{d\delta\underline{x}}{dt}\right) dt \end{array} \tag{1.62}$$

In the preceding equation:

- $\frac{\partial G\left(\underline{x}(t_f)\right)}{\partial \underline{x}(t_f)}$, $\frac{\partial F}{\partial \underline{u}}^T$ and $\frac{\partial F}{\partial \underline{x}}^T$ are row vectors;

[4] Ferguson J., Brief Survey of the History of the Calculus of Variations and its Applications (2004) arXiv:math/0402357

- $\frac{\partial f}{\partial \underline{x}}$ and $\frac{\partial f}{\partial \underline{u}}$ are matrices;
- $\frac{\partial f}{\partial \underline{x}}\delta\underline{x}$, $\frac{\partial f}{\partial \underline{u}}\delta\underline{u}$ and $\frac{d\delta\underline{x}}{dt}$ are column vectors.

Then we introduce the functional $\mathcal{H}$, known as the *Hamiltonian function*, which is defined as follows:

$$\boxed{\mathcal{H}(\underline{x}, \underline{u}, \underline{\lambda}) = F(\underline{x}, \underline{u}) + \underline{\lambda}^T(t) f(\underline{x}, \underline{u})} \tag{1.63}$$

Then:

$$\begin{cases} \frac{\partial \mathcal{H}}{\partial \underline{x}}^T = \frac{\partial F}{\partial \underline{x}}^T + \underline{\lambda}^T(t) \frac{\partial f}{\partial \underline{x}} \\ \frac{\partial \mathcal{H}}{\partial \underline{u}}^T = \frac{\partial F}{\partial \underline{u}}^T + \underline{\lambda}^T(t) \frac{\partial f}{\partial \underline{u}} \end{cases} \tag{1.64}$$

Equation (1.62) becomes:

$$\delta J_a = \frac{\partial G\left(\underline{x}(t_f)\right)}{\partial \underline{x}(t_f)}^T \delta\underline{x}(t_f) + \int_0^{t_f} \left(\frac{\partial \mathcal{H}}{\partial \underline{x}}^T \delta\underline{x} + \frac{\partial \mathcal{H}}{\partial \underline{u}}^T \delta\underline{u} - \underline{\lambda}^T(t) \frac{d\delta\underline{x}}{dt} \right) dt \tag{1.65}$$

Let's concentrate on the last term within the integral that we integrate by parts:

$$\begin{array}{l} \int_0^{t_f} \underline{\lambda}^T(t) \frac{d\delta\underline{x}}{dt}\, dt = \underline{\lambda}^T(t)\delta\underline{x}\big|_0^{t_f} - \int_0^{t_f} \dot{\underline{\lambda}}^T(t)\delta\underline{x}\, dt \\ \Leftrightarrow \int_0^{t_f} \underline{\lambda}^T(t) \frac{d\delta\underline{x}}{dt}\, dt = \underline{\lambda}^T(t_f)\delta\underline{x}(t_f) - \underline{\lambda}^T(0)\delta\underline{x}(0) - \int_0^{t_f} \dot{\underline{\lambda}}^T(t)\delta\underline{x}\, dt \end{array} \tag{1.66}$$

As far as the initial state is imposed, the variation of the initial condition is null; consequently we have $\delta\underline{x}(0) = 0$ and:

$$\int_0^{t_f} \underline{\lambda}^T(t) \frac{d\delta\underline{x}}{dt}\, dt = \underline{\lambda}^T(t_f)\delta\underline{x}(t_f) - \int_0^{t_f} \dot{\underline{\lambda}}^T(t)\delta\underline{x}\, dt \tag{1.67}$$

Using (1.67) within (1.65) leads to the following expression for the first variation of the augmented functional cost:

$$\begin{aligned} \delta J_a = & \left(\frac{\partial G\left(\underline{x}(t_f)\right)}{\partial \underline{x}(t_f)}^T - \underline{\lambda}^T(t_f) \right) \delta\underline{x}(t_f) + \\ & \int_0^{t_f} \left(\frac{\partial \mathcal{H}}{\partial \underline{u}}^T \delta\underline{u} + \left(\frac{\partial \mathcal{H}}{\partial \underline{x}}^T + \dot{\underline{\lambda}}^T(t) \right) \delta\underline{x} \right) dt \end{aligned} \tag{1.68}$$

In order to set the first variation of the augmented functional cost δJ_a to zero the time dependent Lagrange multipliers $\underline{\lambda}(t)$, which are also called costate functions, are chosen as follows:

$$\boxed{\dot{\underline{\lambda}}^T(t) + \frac{\partial \mathcal{H}}{\partial \underline{x}}^T = 0 \Leftrightarrow \dot{\underline{\lambda}}(t) = -\frac{\partial \mathcal{H}}{\partial \underline{x}}} \tag{1.69}$$

This equation is called the adjoint equation. As far as it is a differential equation we need to know the value of $\underline{\lambda}(t)$ at a specific value of time t to be able to compute its solution (also called its *trajectory*) :

- Assuming that final value $\underline{x}(t_f)$ is specified to be $\underline{x}_f$ then the variation $\delta\underline{x}(t_f)$ in (1.68) is zero and $\underline{\lambda}(t_f)$ is set such that $\underline{x}(t_f) = \underline{x}_f$.
- Assuming that final value $\underline{x}(t_f)$ is *not* specified then the variation $\delta\underline{x}(t_f)$ in (1.68) is not equal to zero and the value of $\underline{\lambda}(t_f)$ is set by imposing that the following difference vanishes at final time t_f:

$$\boxed{\frac{\partial G\left(\underline{x}(t_f)\right)^T}{\partial \underline{x}(t_f)} - \underline{\lambda}^T(t_f) = 0 \Leftrightarrow \underline{\lambda}(t_f) = \frac{\partial G\left(\underline{x}(t_f)\right)}{\partial \underline{x}(t_f)}} \tag{1.70}$$

This is the boundary condition, also known as transversality condition, which set the final value of the Lagrange multipliers.

Hence in both situations the first variation of the augmented functional cost (1.68) can be written as:

$$\delta J_a = \int_0^{t_f} \left(\frac{\partial \mathcal{H}}{\partial \underline{u}}^T \delta \underline{u} \right) dt \tag{1.71}$$

Moreover if there is *no constraint on input* $\underline{u}(t)$, then $\delta\underline{u}$ is free and the first variation of the augmented functional cost δJ_a in (1.71) is set to zero through the following necessary condition for optimality:

$$\boxed{\delta J_a = 0 \Rightarrow \frac{\partial \mathcal{H}}{\partial \underline{u}}^T = 0 \Leftrightarrow \frac{\partial \mathcal{H}}{\partial \underline{u}} = 0} \tag{1.72}$$

1.8 Example: brachistochrone problem

1.8.1 Problem overview

A classic optimal control problem is the brachistochrone problem: it consists in computing the curve of fastest descent for a point of mass m which slides without friction and with constant gravitational acceleration g to a fixed end point in the shortest time[5].

The control parameter is the slope $\gamma(t)$ of the curve. Variable $y(t)$ is the horizontal position of the point, $z(t)$ its vertical position in the down direction and $v(t)$ its velocity.

1.8.2 System dynamics

First let's focus on the dynamics of the system using Lagrangian Mechanics. Let $\underline{q}$ be the vector of generalized coordinates. We choose:

$$\underline{q} := \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} = \begin{bmatrix} y(t) \\ z(t) \end{bmatrix} \tag{1.73}$$

[5]https://apmonitor.com/wiki/index.php/Apps/BrachistochroneProblem

The kinetic energy $T(\underline{q}, \underline{\dot{q}})$ and potential energy $V(\underline{q})$ read as follows (remember that the vertical position is oriented downward):

$$\left\{ \begin{array}{l} T(\underline{q}, \underline{\dot{q}}) = \frac{1}{2} m \left(\dot{y}(t)^2 + \dot{z}(t)^2 \right) \\ V(\underline{q}) = -mgz(t) \end{array} \right. \tag{1.74}$$

The Lagrangian $\mathcal{L}$ (for classical mechanics) is defined as the difference between kinetic and potential energy:

$$\mathcal{L} = T(\underline{q}, \underline{\dot{q}}) - V(\underline{q}) = \frac{1}{2} m \left(\dot{y}(t)^2 + \dot{z}(t)^2 \right) + mgz(t) \tag{1.75}$$

The dynamics of the system is then obtained by applying Euler-Lagrange equation:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \frac{\partial \mathcal{L}}{\partial q_i} = \underline{0} \Rightarrow \left\{ \begin{array}{l} \ddot{y}(t) = 0 \\ \ddot{z}(t) = g \end{array} \right. \tag{1.76}$$

Now we introduce the following kinematic relations related to the velocity $v(t)$ of the point and to the slope $\gamma(t)$ of the curve on which the point slides on:

$$\left\{ \begin{array}{l} \dot{y}(t) = v(t) \cos(\gamma(t)) \\ \dot{z}(t) = v(t) \sin(\gamma(t)) \end{array} \right. \tag{1.77}$$

When taking the time derivative of the square of the velocity $v(t)$ and using relations (1.76) we get the following expression of the time derivative of velocity $v(t)$:

$$\begin{array}{rl} v(t)^2 & = \dot{y}(t)^2 + \dot{z}(t)^2 \\ \Rightarrow v(t)\dot{v}(t) & = \dot{y}(t)\ddot{y}(t) + \dot{z}(t)\ddot{z}(t) \\ \Rightarrow \dot{v}(t) & = \frac{v(t) \sin(\gamma(t)) g}{v(t)} \end{array} \tag{1.78}$$

Finally the dynamics of the system is of dimension 3 and reads as follows:

$$\left\{ \begin{array}{l} \dot{y}(t) = v(t) \cos(\gamma(t)) \\ \dot{z}(t) = v(t) \sin(\gamma(t)) \\ \dot{v}(t) = g \sin(\gamma(t)) \end{array} \right. \tag{1.79}$$

In order to reduce the size of the system, it is worth noticing that $\dot{z}(t)$ and $\dot{v}(t)$ depend on $\sin(\gamma(t))$. So we can write:

$$\dot{v}(t) = g \frac{\dot{z}(t)}{v(t)} \Leftrightarrow v(t)\dot{v}(t) = g\dot{z}(t) \tag{1.80}$$

That is, after integration:

$$\begin{array}{l} \frac{1}{2} v(t)^2 - \frac{1}{2} v(0)^2 = g\, z(t) - g\, z(0) \\ \Leftrightarrow v(t) = \sqrt{2g \left(z(t) - z(0) \right) + v(0)^2} \end{array} \tag{1.81}$$

Then the dynamics of the system is reduced to dimension 2:

$$\left\{ \begin{array}{l} \dot{y}(t) = \cos(\gamma(t)) \sqrt{2g\, z + l_0} \\ \dot{z}(t) = \sin(\gamma(t)) \sqrt{2g\, z + l_0} \\ l_0 = v(0)^2 - 2g\, z(0) \end{array} \right. \tag{1.82}$$

Constant l_0 depends on initial conditions $v(0)$ and $z(0)$

The dynamics of the system is reduced one step further through the use of infinitesimal element of curvilinear abscissa ds. Indeed, on one side we have:

$$ds = \sqrt{dy^2 + dz^2} = \sqrt{1 + (z')^2}\, dy \tag{1.83}$$

where:

$$z' := \frac{dz}{dy} \tag{1.84}$$

On the other side, from (1.82) we have:

$$ds = \sqrt{dy^2 + dz^2} = \sqrt{\dot{y}^2 + \dot{z}^2}\, dt = \sqrt{2g\, z + l_0}\, dt \tag{1.85}$$

Thus by equating (1.83) and (1.85) we get the following relation between dy and dt:

$$\sqrt{2g\, z + l_0}\, dt = \sqrt{1 + (z')^2}\, dy \Rightarrow dt = \sqrt{\frac{1 + (z')^2}{2g\, z + l_0}} dy \tag{1.86}$$

Finally the system is reduced to dimension 1 through the following relation:

$$t' := \frac{dt}{dy} = \sqrt{\frac{1 + (z')^2}{2g\, z + l_0}} \tag{1.87}$$

1.8.3 Euler-Lagrange approach

Using Euler-Lagrange formalism, the optimal control problem can be formulated as follows:

$$\begin{array}{c} \text{find } z(y) \\ \text{which minimizes } t_f = \int_0^{t_f} dt = \int_0^{y_f} F\left(z, z'\right) dy \end{array} \tag{1.88}$$

According to (1.87) the functional $F\left(z, z'\right)$ to be minimized reads:

$$F\left(z, z'\right) = \sqrt{\frac{1 + (z')^2}{2g\, z + l_0}} \tag{1.89}$$

Then we have to find $z(y)$ which solves the Euler-Lagrange equation:

$$\frac{d}{dt}\frac{\partial F}{\partial \underline{\dot{x}}} - \frac{\partial F}{\partial \underline{x}} = 0 \tag{1.90}$$

Using Beltrami identity, the first integral of the Euler-Lagrange equation reads as follows where C is a constant:

$$F - \frac{\partial F}{\partial z'} z' = C \Leftrightarrow \sqrt{\frac{1 + (z')^2}{2g\, z + l_0}} - \frac{1}{\sqrt{2g\, z + l_0}} \frac{(z')^2}{\sqrt{1 + (z')^2}} = C \tag{1.91}$$

Multiplying both members by $\sqrt{(2g\, z + l_0)\left(1 + (z')^2\right)}$ and simplifying, we get:

$$C = \frac{1}{\sqrt{(2g\, z + l_0)\left(1 + (z')^2\right)}} \tag{1.92}$$

We thus obtain the following differential equation:

$$\begin{array}{rl} (2g\,z + l_0)\left(1 + (z')^2\right) & = \frac{1}{C^2} \\ \Leftrightarrow 2g\left(z + \frac{l_0}{2g}\right)\left(1 + (z')^2\right) & = \frac{1}{C^2} \end{array} \tag{1.93}$$

The following reduced height $z_r(t)$ is introduced in order to normalize the solution:

$$z_r := z + \frac{l_0}{2g} \Rightarrow dz_r = dz \tag{1.94}$$

We finally get:

$$z_r\left(1 + \left(z_r'\right)^2\right) = \frac{1}{2g\,C^2} \quad \text{where} \quad z_r' = \frac{dz_r}{dy} \tag{1.95}$$

The solution of this differential equation is the cycloid curve. The parametric expression of the cycloid curve is the following where parameter θ varies from 0 to θ_f:

$$\left\{ \begin{array}{l} y = R(C)\,(\theta - \sin(\theta)) \\ z_r = R(C)\,(1 - \cos(\theta)) \end{array} \right. \quad \text{where} \quad R(C) := \frac{1}{4g\,C^2} \tag{1.96}$$

The cycloid curve corresponds to the trajectory of a point on a circle of radius $R(C)$ rolling along a straight line.

The values of C and θ_f shall then be chosen such that the final conditions on y and z_r are fulfilled:

$$\left\{ \begin{array}{l} y(t_f) = R(C)\,(\theta_f - \sin(\theta_f)) \\ z_r(t_f) = R(C)\,(1 - \cos(\theta_f)) \end{array} \right. \tag{1.97}$$

1.8.4 Hamiltonian approach

We will use $z' := u$ as the control variable. The optimal control problem can be formulated as follows:

$$\begin{array}{c} \text{find } u(y) \\ \text{which minimizes } J(u) = \int_0^{t_f} dt = \int_0^{y_f} \sqrt{\frac{1+u^2}{2g\,z(y)+l_0}}dy \\ \text{under the following constraint :} \\ z' = u \end{array} \tag{1.98}$$

The *Hamiltonian function* H reads:

$$H(\underline{x}, \underline{u}, \underline{\lambda}) = F(\underline{x}, \underline{u}) + \underline{\lambda}^T(t) f(\underline{x}, \underline{u}) = \sqrt{\frac{1 + u^2}{2g\,z + l_0}} + \lambda_z u \tag{1.99}$$

Because there is no constraint on control u, the necessary conditions for optimality read as follows:

$$\left\{ \begin{array}{l} \frac{\partial H}{\partial u} = 0 \\ \frac{\partial H}{\partial z} = -\lambda_z' \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \frac{1}{\sqrt{2g\,z+l_0}} \frac{u}{\sqrt{1+u^2}} + \lambda_z = 0 \\ -g\sqrt{1+u^2}\,(2g\,z + l_0)^{-3/2} = -\lambda_z' \end{array} \right. \tag{1.100}$$

Then we have to find the expression of control u as a function of z and λ_z and solve the differential equations involving z and λ_z:

$$\left\{ \begin{array}{l} z' = u(z, \lambda_z) \\ \lambda_z' = g\sqrt{1+u(z,\lambda_z)^2}\,(2g\,z+l_0)^{-3/2} \end{array} \right. \tag{1.101}$$

This could be a tricky task be let's try it ! First from the first equation of (1.100) we get the expression of $1+u^2$ as a function of z and λ_z:

$$\frac{u}{\sqrt{1+u^2}} = -\lambda_z\sqrt{2g\,z+l_0} \Rightarrow 1+u^2 = \frac{1}{1-\lambda_z^2\,(2g\,z+l_0)} \tag{1.102}$$

Using this expression in the second equation of (1.100), we get the following expression of λ_z':

$$\begin{array}{rl} \lambda_z' & = g\sqrt{1+u^2}\,(2g\,z+l_0)^{-3/2} \\ & = g\sqrt{\frac{1}{1-\lambda_z^2(2g\,z+l_0)}}\,(2g\,z+l_0)^{-3/2} \end{array} \tag{1.103}$$

As far as the differential equation involving z is concerned, we use the expression of u^2 to get the following expression of $(z')^2$:

$$z' = u \Rightarrow \left(z'\right)^2 = u^2 = \frac{1}{1-\lambda_z^2\,(2g\,z+l_0)} - 1 = \frac{\lambda_z^2\,(2g\,z+l_0)}{1-\lambda_z^2\,(2g\,z+l_0)} \tag{1.104}$$

From the first equation of (1.100), that is $\frac{1}{\sqrt{2g\,z+l_0}}\frac{u}{\sqrt{1+u^2}}+\lambda_z = 0$, it is clear that $u = z'$ and λ_z have opposite sign. Thus we get:

$$z' = -\lambda_z\sqrt{\frac{(2g\,z+l_0)}{1-\lambda_z^2\,(2g\,z+l_0)}} = -\lambda_z\sqrt{\frac{1}{1-\lambda_z^2\,(2g\,z+l_0)}}\sqrt{2g\,z+l_0} \tag{1.105}$$

Then we get the expression of $\sqrt{\frac{1}{1-\lambda_z^2(2g\,z+l_0)}}$ that we insert in (1.103). We get:

$$\begin{array}{rl} \sqrt{\frac{1}{1-\lambda_z^2(2g\,z+l_0)}} & = -\frac{z'}{\lambda_z\sqrt{2g\,z+l_0}} \\ \Rightarrow \lambda_z' & = -g\frac{z'}{\lambda_z\sqrt{2g\,z+l_0}}\,(2g\,z+l_0)^{-3/2} \\ & = -\frac{g}{\lambda_z}\frac{z'}{(2g\,z+l_0)^2} \end{array} \tag{1.106}$$

We finally get:

$$\lambda_z'\lambda_z = -\frac{g\,z'}{(2g\,z+l_0)^2} \tag{1.107}$$

Thus the first integral of this differential equation is the following where C_1 denotes a constant:

$$\lambda_z^2 = \frac{1}{2g\,z+l_0} + C_1 \tag{1.108}$$

Or, equivalently:

$$\lambda_z^2\,(2g\,z+l_0) = 1 + C_1\,(2g\,z+l_0) \tag{1.109}$$

Then this result is used in (1.102) to get the following relation:

$$\begin{array}{rl} 1+u^2 & = \frac{1}{1-\lambda_z^2(2g\,z+l_0)} \\ & = \frac{-1}{C_1(2g\,z+l_0)} \\ \Rightarrow (2g\,z+l_0)\left(1+u^2\right) & = -\frac{1}{C_1} \end{array} \tag{1.110}$$

Having in mind that $z' = u$, we retrieve the first integral (1.93) which has been obtained through Baltrami identity. Then the resolution process is similar to what has been done in the previous section.

1.9 Hamilton-Jacobi-Bellman (HJB) equation

1.9.1 Finite horizon control

Let $J^*(\underline{x}, t)$ be the optimal cost-to-go function between t and t_f:

$$J^*(\underline{x}, t) = min_{\underline{u}(t)\in\mathcal{U}} \int_t^{t_f} F(\underline{x}(t), \underline{u}(t))dt \tag{1.111}$$

The Hamilton-Jacobi-Bellman equation related to the optimal control problem (1.56) under the constraint (1.55) is the following first order partial derivative equation[6]:

$$\boxed{-\frac{\partial J^*(\underline{x}, t)}{\partial t} = min_{\underline{u}(t)\in\mathcal{U}} \left(F(\underline{x}, \underline{u}) + \left(\frac{\partial J^*(\underline{x}, t)}{\partial \underline{x}}\right)^T f(\underline{x}, \underline{u}) \right)} \tag{1.112}$$

or, equivalently:

$$-\frac{\partial J^*(\underline{x}, t)}{\partial t} = \mathcal{H}^*\left(\frac{\partial J^*(\underline{x}, t)}{\partial \underline{x}}, \underline{x}(t)\right) \tag{1.113}$$

where

$$\mathcal{H}^*(\underline{\lambda}(t), \underline{x}(t)) = min_{\underline{u}(t)\in\mathcal{U}} \left(F(\underline{x}, \underline{u}) + \left(\frac{\partial J^*(\underline{x}, t)}{\partial \underline{x}}\right)^T f(\underline{x}, \underline{u}) \right) \tag{1.114}$$

For the time-dependent case, the terminal condition on the optimal cost-to-go function solution of (1.112) reads:

$$J^*(\underline{x}, t_f) = G\left(\underline{x}(t_f)\right) \tag{1.115}$$

It is worth noticing that the Lagrange multiplier $\underline{\lambda}(t)$ represents the partial derivative with respect to the state of the optimal cost-to-go function[7]:

$$\boxed{\underline{\lambda}(t) = \frac{\partial J^*(\underline{x}, t)}{\partial \underline{x}}} \tag{1.116}$$

[6] da Silva J., de Sousa J., Dynamic Programming Techniques for Feedback Control, Proceedings of the 18th World Congress, Milano (Italy) August 28 - September 2, 2011

[7] Alazard D., Optimal Control & Guidance: From Dynamic Programming to Pontryagin's Minimum Principle, lecture notes

1.9.2 Principle of optimality, dynamic programming

The preceding results lead to the so-called *dynamic programming* approach which has been introduced by Bellman[8] in 1957. This is a very powerful approach which encompasses both necessary and sufficient conditions for optimality. Contrary to the Lagrange multipliers approach, the *dynamic programming* solves the constrained optimal problem directly. Behind the *dynamic programming* is the *principle of optimality*, which states that from any point on an optimal state space trajectory, the remaining trajectory is optimal for the corresponding problem initiated at that point. This result can be quite easily applied for discrete time system but for continuous time system it involves to find the solution of a partial derivative equation which may be difficult in practice.

1.9.3 Infinite horizon control

For *infinite horizon* control problem, the problem is to find the control $\underline{u}(t)$ which minimizes the following cost-to-go function:

$$J(\underline{x}) = \int_0^\infty F(\underline{x}(t), \underline{u}(t))dt \tag{1.117}$$

under the following nonlinear *time invariant* dynamics of the form:

$$\begin{cases} \dot{\underline{x}} = f(\underline{x}, \underline{u}) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{1.118}$$

Then, denoting by $J^*(\underline{x})$ the optimal cost function (which now no more depends on time t), the Hamilton-Jacobi-Bellman equation related to this optimal control problem reads:

$$0 = min_{\underline{u}(t) \in \mathcal{U}} \left(F(\underline{x}, \underline{u}) + \left(\frac{\partial J^*(\underline{x})}{\partial \underline{x}} \right)^T f(\underline{x}, \underline{u}) \right) \tag{1.119}$$

Lower bounds on the optimal cost are obtained by integrating the corresponding inequality:

$$0 \leq F(\underline{x}, \underline{u}) + \left(\frac{\partial J(\underline{x})}{\partial \underline{x}} \right)^T f(\underline{x}, \underline{u}) \qquad \forall \, \underline{x}, \, \underline{u} \tag{1.120}$$

It it worth noticing that:

$$\begin{array}{rl} \int_0^\infty \left(\frac{\partial J(\underline{x})}{\partial \underline{x}} \right)^T f(\underline{x}, \underline{u})dt & = \int_0^\infty \left(\frac{\partial J(\underline{x})}{\partial \underline{x}} \right)^T \dot{\underline{x}}(t)dt \\ & = \int_{\underline{x}(0)}^{\underline{x}(\infty)} \left(\frac{\partial J(\underline{x})}{\partial \underline{x}} \right)^T d\underline{x} \\ & = J(\underline{x}(\infty)) - J(\underline{x}(0)) \end{array} \tag{1.121}$$

Thus, assuming that $\underline{x}(\infty) = \underline{0}$, we get:

$$J(\underline{x}(0)) - J(\underline{0}) = - \int_0^\infty \left(\frac{\partial J(\underline{x})}{\partial \underline{x}} \right)^T f(\underline{x}, \underline{u})dt \leq \int_0^\infty F(\underline{x}, \underline{u})dt \tag{1.122}$$

[8] Bellman R., Dynamic programming, Princeton University Press, 1957

Moreover, the optimal cost $J^*(\underline{x})$ has a decay rate given by $-F(\underline{x}, \underline{u}^*)$, which is negative. Thus $J^*(\underline{x})$ may serve as a Lyapunov function to prove that the optimal control law is stabilizing[9].

1.9.4 Application of HJB equation to linear time invariant systems

We consider in this section linear time invariant systems, where $\underline{x}(t)$ is the state vector and $\underline{u}(t)$ is the control vector of dimension m. Furthermore we assume that the cost-to-go function $J(\underline{x}, t)$ to be minimized is quadratic:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ J(\underline{x}, t) = \int_t^{t_f} \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right) \, dt \end{cases} \tag{1.123}$$

where:

$$\begin{cases} \mathbf{Q} = \mathbf{Q}^T \geq 0 \\ \mathbf{R} = \mathbf{R}^T > 0 \end{cases} \tag{1.124}$$

Assuming that the final state at $t = t_f$ is set to zero, a candidate solution $J^*(\underline{x}, t)$ of the Hamilton-Jacobi-Bellman (HJB) partial differential equation is the following quadratic function:

$$J^*(\underline{x}, t) := \underline{x}^T\mathbf{P}(t)\underline{x} \text{ where } \mathbf{P}(t) = \mathbf{P}^T(t) \geq 0 \tag{1.125}$$

Thus:

$$\begin{cases} \frac{\partial J^*(\underline{x},t)}{\partial t} = \underline{x}^T\dot{\mathbf{P}}(t)\underline{x} \\ \frac{\partial J^*(\underline{x},t)}{\partial \underline{x}} = 2\mathbf{P}(t)\underline{x}(t) \end{cases} \tag{1.126}$$

Finally, assuming unconstrained control, that is $\underline{u}(t) \in \mathbb{R}^m$, the Hamilton-Jacobi-Bellman (HJB) equation (1.112) reads as follows:

$$\begin{aligned} -\underline{x}^T\dot{\mathbf{P}}(t)\underline{x} &= min_{\underline{u}(t) \in \mathbb{R}^m} \left(F(\underline{x}, \underline{u}) + \left(\frac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)^T f(\underline{x}, \underline{u})\right) \\ &= min_{\underline{u}(t) \in \mathbb{R}^m} \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t) \right. \\ &\qquad \left. + 2\underline{x}^T(t)\mathbf{P}(t)\left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right)\right) \end{aligned} \tag{1.127}$$

To get $min_{\underline{u}(t) \in \mathbb{R}^m}$, we set the derivative of its argument with respect to $\underline{u}$ to zero:

$$\begin{aligned} &\frac{\partial}{\partial \underline{u}} \left(\underline{x}^T\mathbf{Q}\underline{x} + \underline{u}^T\mathbf{R}\underline{u} + 2\underline{x}^T\mathbf{P}\left(\mathbf{A}\underline{x} + \mathbf{B}\underline{u}\right)\right) = \underline{0} \\ &\Rightarrow 2\left(\mathbf{R}\underline{u} + \mathbf{B}^T\mathbf{P}\underline{x}\right) = \underline{0} \end{aligned} \tag{1.128}$$

We finally get:

$$\underline{u}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)\underline{x}(t) \tag{1.129}$$

Thus the Hamilton-Jacobi-Bellman (HJB) partial differential equation reads:

$$-\underline{x}^T\dot{\mathbf{P}}(t)\underline{x} = \underline{x}^T\mathbf{Q}\underline{x} - \underline{x}^T\mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)\underline{x} + 2\underline{x}^T\mathbf{P}(t)\mathbf{A}\underline{x} \tag{1.130}$$

[9] Anders Rantzer and Mikael Johansson, Piecewise Linear Quadratic Optimal Control, IEEE Transactions On Automatic Control, Vol. 45, No. 4, April 2000

Then, using the fact that $\mathbf{P}(t) = \mathbf{P}^T(t)$, we can write:

$$2\underline{x}^T\mathbf{P}(t)\mathbf{A}\underline{x} = 2\underline{x}^T\mathbf{A}^T\mathbf{P}(t)\underline{x} = \underline{x}^T\left(\mathbf{P}(t)\mathbf{A} + \mathbf{A}^T\mathbf{P}(t)\right)\underline{x} \tag{1.131}$$

Then the Hamilton-Jacobi-Bellman (HJB) partial differential equation becomes:

$$-\underline{x}^T\dot{\mathbf{P}}(t)\underline{x} = \underline{x}^T\mathbf{Q}\underline{x} - \underline{x}^T\mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)\underline{x} + \underline{x}^T\left(\mathbf{P}(t)\mathbf{A} + \mathbf{A}^T\mathbf{P}(t)\right)\underline{x} \tag{1.132}$$

Because this equation must be true $\forall\, \underline{x}$, we conclude that $\mathbf{P}(t) = \mathbf{P}^T(t)$ shall solve the following differential Riccati differential equation:

$$-\dot{\mathbf{P}}(t) = \mathbf{A}^T\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A} - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t) + \mathbf{Q} \tag{1.133}$$

For time invariant systems with infinite horizon ($t_f \to \infty$), the optimal cost-to-go function $J^*(\underline{x}, t)$ is independent of time t: $J^*(\underline{x}, t) = J^*(\underline{x})$. Thus matrix $\mathbf{P}(t)$ becomes a constant matrix:

$$t_f \to \infty \Rightarrow J^*(\underline{x}, t) = J^*(\underline{x}) \Rightarrow \mathbf{P}(t) = \mathbf{P} \tag{1.134}$$

1.10 Pontryagin's principle

In this section we consider the optimal control problem with possibly control-state constraints. More specifically we consider the problem of finding a control $\underline{u}$ that *minimizes* the following performance index:

$$J(\underline{u}(t)) = G\left(\underline{x}(t_f)\right) + \int_0^{t_f} F(\underline{x}(t), \underline{u}(t))\, dt \tag{1.135}$$

Under the following constraints:

– Dynamics and boundary conditions:

$$\begin{cases} \dot{\underline{x}} = f(\underline{x}, \underline{u}) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{1.136}$$

– Mixed control-state constraints:

$$c(\underline{x}, \underline{u}) \leq 0, \ \textit{where}\ c(\underline{x}, \underline{u}) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R} \tag{1.137}$$

Usually a slack variable $\alpha(t)$, which is actually a new control variable, is introduced in order to convert the preceding inequality constraint into an equality constraint:

$$c(\underline{x}, \underline{u}) + \alpha^2(t) = 0, \ \textit{where}\ \alpha(t) : \mathbb{R} \to \mathbb{R} \tag{1.138}$$

To solve this problem we introduce the *augmented Hamiltonian function* $\mathcal{H}_a(\underline{x}, \underline{u}, \underline{\lambda}, \underline{\mu})$ which is defined as follows [10]:

$$\begin{aligned} \mathcal{H}_a(\underline{x}, \underline{u}, \underline{\lambda}, \underline{\mu}, \alpha) &= \mathcal{H}(\underline{x}, \underline{u}, \underline{\lambda}) + \underline{\mu}\left(c(\underline{x}, \underline{u}) + \alpha^2\right) \\ &= F(\underline{x}, \underline{u}) + \underline{\lambda}^T(t) f(\underline{x}, \underline{u}) + \underline{\mu}(t)\left(c(\underline{x}, \underline{u}) + \alpha^2\right) \end{aligned} \tag{1.139}$$

Then the Pontryagin's principle states that the optimal control $\underline{u}^*$ must satisfy the following conditions:

[10] Hull D. G., Optimal Control Theory for Applications, Springer (2003)

– Adjoint equation and transversality condition:

$$\begin{cases} \dot{\underline{\lambda}}(t) = -\frac{\partial \mathcal{H}_a}{\partial \underline{x}} \\ \underline{\lambda}(t_f) = \frac{\partial G\left(\underline{x}(t_f)\right)}{\partial \underline{x}(t_f)} \end{cases} \tag{1.140}$$

– Local minimum condition for augmented Hamiltonian:

$$\begin{cases} \frac{\partial \mathcal{H}_a}{\partial \underline{u}} = 0 \\ \frac{\partial \mathcal{H}_a}{\partial \alpha} = 0 \Rightarrow 2\underline{\mu}\alpha = 0 \end{cases} \tag{1.141}$$

– Sign of multiplier $\underline{\mu}(t)$ and complementarity condition: the equation $\frac{\partial \mathcal{H}_a}{\partial \alpha} = 0$ implies $2\underline{\mu}\alpha = 0$. Thus either $\underline{\mu} = 0$, which is an off-boundary arc, or $\alpha = 0$ which is an on-boundary arc:

  – For the off-boundary arc where $\underline{\mu} = 0$ control $\underline{u}$ is obtained from $\frac{\partial \mathcal{H}_a}{\partial \underline{u}} = 0$ and α from the equality constraint $c(\underline{x}, \underline{u}) + \alpha^2 = 0$;
  – For the on-boundary arc where $\alpha = 0$ control $\underline{u}$ is obtained from equality constraint $c(\underline{x}, \underline{u}) = 0$. Indeed there always exists a smooth function $\underline{u}_b(\underline{x})$ called boundary control which satisfies:

$$c\left(\underline{x}, \underline{u}_b(\underline{x})\right) = 0 \tag{1.142}$$

  Then multiplier $\underline{\mu}$ is obtained from $\frac{\partial \mathcal{H}_a}{\partial \underline{u}} = 0$:

$$0 = \frac{\partial \mathcal{H}_a}{\partial \underline{u}} = \frac{\partial \mathcal{H}}{\partial \underline{u}} + \underline{\mu}\frac{\partial c(\underline{x}, \underline{u})}{\partial \underline{u}} \Rightarrow \underline{\mu} = -\left.\frac{\frac{\partial \mathcal{H}_a}{\partial \underline{u}}}{\frac{\partial c(\underline{x}, \underline{u})}{\partial \underline{u}}}\right|_{\underline{u}=\underline{u}_b(\underline{x})} \tag{1.143}$$

Weierstrass conditions (proposed in 1879) for a variational extremum states that optimal control $\underline{u}^*$ and α^* within the *augmented Hamiltonian function* $\mathcal{H}_a$ must satisfy the following condition for a minimum at every point of the optimal path:

$$\mathcal{H}_a(\underline{x}^*, \underline{u}^*, \underline{\lambda}^*, \underline{\mu}^*, \alpha^*) - \mathcal{H}_a(\underline{x}^*, \underline{u}, \underline{\lambda}^*, \underline{\mu}^*, \alpha) < 0 \tag{1.144}$$

Since $c(\underline{x}, \underline{u}) + \alpha^2(t) = 0$, the Weierstrass conditions for a variational extremum can be rewritten as a function of the *Hamiltonian function* $\mathcal{H}$ and the inequality constraint:

$$\begin{cases} \mathcal{H}(\underline{x}^*, \underline{u}^*, \underline{\lambda}^*) - \mathcal{H}(\underline{x}^*, \underline{u}, \underline{\lambda}^*) < 0 \\ c(\underline{x}^*, \underline{u}^*) \leq 0 \end{cases} \tag{1.145}$$

or, equivalently:

$$\boxed{\underline{u}^* = min_{\underline{u}(t) \in \mathcal{U}} \mathcal{H}(\underline{x}^*, \underline{u}, \underline{\lambda}^*)} \tag{1.146}$$

where $\mathcal{U}$ denotes the set of admissible values for the control $\underline{u}$ (here $\underline{u}(t) \in \mathcal{U}$ as soon as $c(\underline{x}^*, \underline{u}) \leq 0$). The last relation is the so-called Pontryagin's principle.

1.11 Hamiltonian over time

1.11.1 General result

From Pontryagin's principle, special conditions for the Hamiltonian can be derived[11]. When the final time t_f is fixed and the Hamiltonian $\mathcal{H}$ does not depend explicitly on time, that is when $\frac{\partial \mathcal{H}}{\partial t} = 0$, then the Hamiltonian functional $\mathcal{H}$ remains constant along an optimal trajectory:

$$\boxed{\mathcal{H}(\underline{x}^*, \underline{u}^*, \underline{\lambda}^*) = \text{constant}} \tag{1.147}$$

Moreover, if the terminal time t_f is free, then along the optimal trajectory we have:

$$\boxed{\mathcal{H}(\underline{x}^*, \underline{u}^*, \underline{\lambda}^*) = 0 \text{ when } t_f \text{ is free}} \tag{1.148}$$

1.11.2 Autonomous system without constraint on input

We will show that the Hamiltonian $\mathcal{H}$ is constant along the optimal trajectory in the particular case of autonomous system assuming no constraint on input $\underline{u}(t)$. For an autonomous system, the function $f()$ is not an explicit function of time. From (1.63) we get:

$$\frac{d\mathcal{H}}{dt} = \frac{\partial \mathcal{H}}{\partial \underline{x}}^T \frac{d\underline{x}}{dt} + \frac{\partial \mathcal{H}}{\partial \underline{u}}^T \frac{d\underline{u}}{dt} + \frac{\partial \mathcal{H}}{\partial \underline{\lambda}}^T \frac{d\underline{\lambda}}{dt} \tag{1.149}$$

According to (1.55), (1.63) and (1.69) we have

$$\left\{ \begin{array}{l} \dot{\underline{\lambda}}^T(t) = -\frac{\partial \mathcal{H}}{\partial \underline{x}}^T \\ \frac{\partial \mathcal{H}}{\partial \underline{\lambda}}^T = f^T = \dot{\underline{x}}^T \end{array} \right. \Rightarrow \frac{d\mathcal{H}}{dt} = -\dot{\underline{\lambda}}^T(t) \frac{d\underline{x}}{dt} + \frac{\partial \mathcal{H}}{\partial \underline{u}}^T \frac{d\underline{u}}{dt} + \dot{\underline{x}}^T \frac{d\underline{\lambda}}{dt} \tag{1.150}$$

Having in mind that the Hamiltonian $\mathcal{H}$ is a scalar functional we get:

$$\dot{\underline{\lambda}}^T(t) \frac{d\underline{x}}{dt} = \dot{\underline{x}}^T(t) \frac{d\underline{\lambda}}{dt} \Rightarrow \frac{d\mathcal{H}}{dt} = \frac{\partial \mathcal{H}}{\partial \underline{u}}^T \frac{d\underline{u}}{dt} \tag{1.151}$$

Finally, assuming no constraint on input $\underline{u}(t)$, we use (1.72) to obtain relation (1.147):

$$\frac{\partial \mathcal{H}}{\partial \underline{u}} = 0 \Rightarrow \frac{d\mathcal{H}}{dt} = 0 \Rightarrow \mathcal{H}(\underline{x}^*, \underline{u}^*, \underline{\lambda}^*) = \text{constant} \tag{1.152}$$

Example 1.2. *As in example 1.1 we consider again the problem of finding the shortest distance between two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ in the euclidean plane.*

Setting $u(x) = y'(x)$ the length of the path between the two points is defined by:

$$J(u(x)) = \int_{P_1}^{P_2} \sqrt{dx^2 + dy^2} = \int_{x_1}^{x_2} \sqrt{1 + u(x)^2}\, dx \tag{1.153}$$

[11] https://en.wikipedia.org/wiki/Hamiltonian_(control_theory)

Here $J(u(x))$ is the performance index to be minimized under the following constraints:

$$\left\{ \begin{array}{l} y'(x) = u(x) \\ y(x_1) = y_1 \\ y(x_2) = y_2 \end{array} \right. \tag{1.154}$$

Let $\lambda(x)$ be the Lagrange multiplier, which is here a scalar. The Hamiltonian $\mathcal{H}$ reads:

$$\mathcal{H} = \sqrt{1 + u^2(x)} + \lambda(x)u(x) \tag{1.155}$$

The necessary conditions for optimality are the following:

$$\left\{ \begin{array}{l} \frac{\partial \mathcal{H}}{\partial y} = -\lambda'(x) \Leftrightarrow \lambda'(x) = 0 \\ \frac{\partial \mathcal{H}}{\partial u} = 0 \Leftrightarrow \frac{u(x)}{\sqrt{1+u^2(x)}} + \lambda(x) = 0 \end{array} \right. \tag{1.156}$$

Denoting by c a constant we get from the first equation of (1.156)*:*

$$\lambda(x) = c \tag{1.157}$$

Using this relation in the second equation of (1.156) *leads to the following expression of $u(x)$ where constant a is introduced:*

$$\frac{u(x)}{\sqrt{1+u^2(x)}} + c = 0 \Rightarrow u^2(x) = \frac{c^2}{1-c^2} \Rightarrow u(x) = \sqrt{\frac{c^2}{1-c^2}} := a = constant \tag{1.158}$$

Thus, the shortest distance between two fixed points in the euclidean plane is a curve with constant slope, that is a straight-line:

$$y(x) = a\,x + b \tag{1.159}$$

We obviously retrieve the result of example 1.1.

Moreover, we can check that over the optimal trajectory the Hamiltonian $\mathcal{H}$ is constant (not null because here the final value of x is set to x_2). Indeed:

$$\left\{ \begin{array}{l} \lambda(x) = c \\ u(x) = \sqrt{\frac{c^2}{1-c^2}} \end{array} \right. \Rightarrow \mathcal{H} = \sqrt{1 + u^2(x)} + \lambda(x)u(x) = constant \tag{1.160}$$

■

1.11.3 Free final time

It is worth noticing that if final time t_f is not specified, and after having noticed that $f(t_f) - \underline{\dot{x}}(t_f) = 0$, the following term shall be added to δJ_a in (1.62):

$$\left(F(t_f) + \frac{\partial G\left(\underline{x}(t_f)\right)^T}{\partial \underline{x}} f(t_f) \right) \delta t_f \tag{1.161}$$

In this case the first variation of the augmented performance index with respect to δt_f is zero as soon as:

$$F(t_f) + \frac{\partial G\left(\underline{x}(t_f)\right)^T}{\partial \underline{x}} f(t_f) = 0 \tag{1.162}$$

As far as boundary conditions (1.70) apply we get:

$$\underline{\lambda}(t_f) = \frac{\partial G\left(\underline{x}(t_f)\right)}{\partial \underline{x}(t_f)} \Rightarrow F(t_f) + \underline{\lambda}(t_f) f(t_f) = 0 \tag{1.163}$$

The preceding equation is called transversality condition. We recognize in $F(t_f) + \underline{\lambda}(t_f) f(t_f)$ the value of the *Hamiltonian function* $\mathcal{H}(t)$ defined in (1.63) at final time t_f. Because the Hamiltonian $\mathcal{H}(t)$ is constant along an optimal trajectory for an autonomous system (see (1.147)) it is concluded that $\mathcal{H}(t) = 0$ along an optimal trajectory for an autonomous system when final time t_f is free.

Alternatively we can introduce a new variable, denoted s for example, which is related to time t as follows:

$$t(s) = t_0 + (t_f - t_0)s \;\forall\; s \in [0, 1] \tag{1.164}$$

From the preceding equation we get:

$$dt = (t_f - t_0)\, ds \tag{1.165}$$

Then the optimal control problem with respect to time t where the final time t_f is free is changed into an optimal control problem with respect to new variable s and an additional state $t_f(s)$ which is constant with respect to s. The optimal control problem reads:

Minimize:

$$J(\underline{u}(s)) = G\left(\underline{x}(1)\right) + \int_0^1 \left(t_f(s) - t_0\right) F(\underline{x}(s), \underline{u}(s))\, ds \tag{1.166}$$

Under the following constraints:

– Dynamics and boundary conditions:

$$\begin{cases} \frac{d}{ds}\underline{x}(s) = \frac{d\underline{x}(t)}{dt}\frac{dt}{ds} = (t_f(s) - t_0) f(\underline{x}(s), \underline{u}(s)) \\ \frac{d}{ds} t_f(s) = 0 \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{1.167}$$

– Mixed control-state constraints:

$$c(\underline{x}(s), \underline{u}(s)) \leq 0, \;\textit{where}\; c(\underline{x}(s), \underline{u}(s)) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R} \tag{1.168}$$

1.12 Bang-bang control

1.12.1 Pontryagin's principle application

Bang–bang control is a term used to indicate that the control $\underline{u}$ switches abruptly between two values. It appears when the control $\underline{u}$ is restricted to be between a lower and an upper bound. We apply Pontryagin's principle in the following cases:

- For a problem where the *Hamiltonian function* $\mathcal{H}$ is linear in the *scalar* control u we can write:

$$\mathcal{H} = a + \sigma\, u \tag{1.169}$$

When *scalar* control u is limited between u_{min} and u_{max}, Pontryagin's principle provides the following necessary condition for optimality:

$$u_{min} \leq u(t) \leq u_{max} \Rightarrow u(t) = \begin{cases} u_{max} \text{ if } \sigma(t) = \dfrac{\partial \mathcal{H}}{\partial u} < 0 \\ u_{min} \text{ if } \sigma(t) = \dfrac{\partial \mathcal{H}}{\partial u} > 0 \\ \in [u_{min}, u_{max}] \text{ if } \sigma(t) = \dfrac{\partial \mathcal{H}}{\partial u} = 0 \end{cases} \tag{1.170}$$

- For multi-inputs systems, suppose that the *Hamiltonian function* $\mathcal{H}$ is related to the control *vector* $\underline{u}(t)$ as follows:

$$\mathcal{H} = a + \underline{b}^T\, \underline{u} + \|\underline{u}\| \text{ where } \underline{b} \neq \underline{0} \tag{1.171}$$

Cauchy–Schwartz inequality may be applied to get[12]:

$$\underline{b}^T\, \underline{u} \geq -\|\underline{b}\|\|\underline{u}\| \Rightarrow \mathcal{H} \geq a + \|\underline{u}\|\,(1 - \|\underline{b}\|) \tag{1.172}$$

and the equality is obtained when $\underline{u}$ is proportionnal to $\underline{b}$:

$$\underline{u} = -\alpha\, \frac{\underline{b}}{\|\underline{b}\|} \qquad \text{where } \alpha \geq 0 \tag{1.173}$$

Moreover Pontryagin's principle provides the following necessary condition for optimality[12], assuming that $\|\underline{u}\| \leq u_{max}$:

$$\|\underline{u}\| \leq u_{max} \Rightarrow \alpha = \begin{cases} u_{max} \text{ if } \sigma < 0 \\ 0 \text{ if } \sigma > 0 \\ \in [0, u_{max}] \text{ if } \sigma = 0 \end{cases} \qquad \text{where } \sigma = 1 - \|\underline{b}\| \tag{1.174}$$

Function σ is usually called the switching function. Thus optimal control $u(t)$ switches at times when switching function σ switches from negative to positive (or vice-versa). This type of control where the control is always set to boundary values is called *bang-bang control*.

In addition, and as the unconstrained control case, the *Hamiltonian functional* $\mathcal{H}$ *remains constant along an optimal trajectory for an autonomous system* when there are constraints on input $\underline{u}(t)$. Indeed in that situation control $\underline{u}(t)$ is constant (it is set either to its minimum or maximum value) and consequently $\frac{du}{dt}$ is zero. From (1.151) we get $\frac{d\mathcal{H}}{dt} = 0$.

[12] Bertrand R., Epenoy R., New smoothing techniques for solving bang-bang optimal control problems - Numerical results and statistical interpretation, Optimal Control Applications and Methods 23(4):171 - 197, July 2002, DOI:10.1002/oca.709

Last but not least, assume that the performance index to be minimized reads as follows where $\lambda_0 > 0$:

$$J(\underline{u}(t)) = \frac{\lambda_0}{2} \int_0^{t_f} \|\underline{u}(t)\| \, dt \tag{1.175}$$

Then, as suggested by Bertrand & Epenoy[12], it could be valuable to deduce the solution of the initial problem from the successive solutions of an auxiliary problem through an homotopic approach by defining the following perturbed performance index:

$$J_\epsilon(\underline{u}(t)) = \frac{\lambda_0}{2} \int_0^{t_f} \left(\|\underline{u}(t)\| - \epsilon \, h\left(\|\underline{u}(t)\|\right)\right) \, dt \tag{1.176}$$

Parameter ϵ is assumed to be in the interval $]0, 1]$ and function h is a continuous function satisfying $h\,(w) \geq 0 \;\forall\, w \in [0, 1]$. For example, one could choose $h(w) = w - w^2$; with this choice $\|\underline{u}(t)\| - \epsilon\, h\left(\|\underline{u}(t)\|\right) = \|\underline{u}(t)\|^2$ for $\epsilon = 1$ and $\|\underline{u}(t)\| - \epsilon\, h\left(\|\underline{u}(t)\|\right) = \|\underline{u}(t)\|$ for $\epsilon = 0$.

If $h(w) \to \infty$ as w approaches 1 or 0, then h is called a barrier function, otherwise it is a penalty function.

The homotopic (or continuation) approach[12] consists in solving the perturbed problem with $\epsilon = 1$. Then, after defining a decreasing sequence of ϵ values ($\epsilon_1 = 1 > \epsilon_2 > \cdots > \epsilon_n > 0$), the current optimal control problem associated with $\epsilon = \epsilon_k$ where $k = 2, \cdots, n$ is solved with the solution of the previous one as a starting point.

1.12.2 Example 1

Consider a simple mass m which moves on the x-axis and is subject to a force $f(t)$[13]. Equation of motion reads:

$$m\ddot{y}(t) = f(t) \tag{1.177}$$

We set control $u(t)$ as:

$$u(t) = \frac{f(t)}{m} \tag{1.178}$$

Consequently the equation of motion reduces to:

$$\ddot{y}(t) = u(t) \tag{1.179}$$

The state space realization of this system is the following:

$$\left\{ \begin{array}{l} x_1(t) = y(t) \\ x_2(t) = \dot{y}(t) \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \dot{x}_1(t) = x_2(t) \\ \dot{\underline{x}}_2(t) = u(t) \end{array} \right. \Leftrightarrow f(x, u) = \begin{bmatrix} x_2(t) \\ u(t) \end{bmatrix} \tag{1.180}$$

We will assume that the initial position of the mass is zero and that the movement starts from rest:

$$\left\{ \begin{array}{l} y(0) = 0 \\ \dot{y}(0) = 0 \end{array} \right. \tag{1.181}$$

[13] Linear Systems: Optimal and Robust Control 1st Edition, by Alok Sinha, CRC Press

We will assume that control $u(t)$ is subject to the following constraint:

$$u_{min} \leq u(t) \leq u_{max} \tag{1.182}$$

First we are looking for the optimal control $u(t)$ which enables the mass to cover the maximum distance in a fixed time t_f:

The objective of the problem is to maximize $y(t_f)$. This corresponds to minimize the opposite of $y(t_f)$; consequently the cost $J(u(t))$ reads as follows where $F(x, u) = 0$ when compared to (1.56):

$$J(u(t)) = G\left(x(t_f)\right) = -y(t_f) := -x_1(t_f) \tag{1.183}$$

As $F(x, u) = 0$ the Hamiltonian for this problem reads:

$$\begin{aligned} \mathcal{H}(\underline{x}, u, \underline{\lambda})) &= \lambda(t)^T f(x, u) \\ &= \begin{bmatrix} \lambda_1(t) & \lambda_2(t) \end{bmatrix} \begin{bmatrix} x_2(t) \\ u(t) \end{bmatrix} \\ &= \lambda_1(t) x_2(t) + \lambda_2(t) u(t) \end{aligned} \tag{1.184}$$

Adjoint equations read:

$$\dot{\lambda}(t) = -\frac{\partial \mathcal{H}}{\partial x} \Leftrightarrow \left\{ \begin{array}{l} \dot{\lambda}_1(t) = -\frac{\partial \mathcal{H}}{\partial x_1} = 0 \\ \dot{\lambda}_2(t) = -\frac{\partial \mathcal{H}}{\partial x_2} = -\lambda_1(t) \end{array} \right. \tag{1.185}$$

Solutions of adjoint equations are the following where c and d are constants:

$$\left\{ \begin{array}{l} \lambda_1(t) = c \\ \lambda_2(t) = -ct + d \end{array} \right. \tag{1.186}$$

As far as final time t_f is not specified values of constants c and d are determined by transversality condition (1.70):

$$\lambda(t_f) = \frac{\partial G\left(x(t_f)\right)}{\partial x(t_f)} \Rightarrow \left\{ \begin{array}{l} \lambda_1(t_f) = \frac{\partial\left(-x_1(t_f)\right)}{\partial x_1(t_f)} = -1 \\ \lambda_2(t_f) = \frac{\partial\left(-x_1(t_f)\right)}{\partial x_2(t_f)} = 0 \end{array} \right. \tag{1.187}$$

Consequently:

$$\left\{ \begin{array}{l} c = -1 \\ d = -t_f \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \lambda_1(t) = -1 \\ \lambda_2(t) = t - t_f \end{array} \right. \tag{1.188}$$

Thus the Hamiltonian $\mathcal{H}$ reads as follows:

$$\mathcal{H}(\underline{x}, u, \underline{\lambda})) = \lambda_1(t) x_2(t) + \lambda_2(t) u(t) = -x_2(t) + (t - t_f) u(t) \tag{1.189}$$

Then $\frac{\partial \mathcal{H}}{\partial u} = t - t_f \leq 0 \; \forall \; 0 \leq t \leq t_f$. Applying (1.170) leads to the expression of control $u(t)$:

$$\frac{\partial \mathcal{H}}{\partial u} \leq 0 \Rightarrow u(t) = u_{max} \; \forall \; 0 \leq t \leq t_f \tag{1.190}$$

This is of common sense when the objective is to cover the maximum distance in a fixed time without any constraint on the vehicle velocity at the final time. The optimal state trajectory can be easily obtained by solving the state equations with given initial conditions:

$$\begin{cases} \underline{\dot{x}}_1 = x_2 \\ \underline{\dot{x}}_2 = u_{max} \end{cases} \Rightarrow \begin{cases} x_1(t) = \frac{1}{2} u_{max} t^2 \\ x_2(t) = u_{max} t \end{cases} \tag{1.191}$$

The Hamiltonian along the optimal trajectory has the following value:

$$\mathcal{H}(\underline{x}, u, \underline{\lambda})) = \lambda_1(t) x_2(t) + \lambda_2(t) u(t) = -u_{max} t + (t - t_f) u_{max} = -u_{max} t_f \tag{1.192}$$

As expected the Hamiltonian along the optimal trajectory is constant. The minimum value of the performance index is:

$$J(u(t)) = -x_1(t_f) = -\frac{1}{2} u_{max} t_f^2 \tag{1.193}$$

Alternatively, we can write $J(u(t))$ as follows:

$$J(u(t)) = -y(t_f) = \int_0^{tf} \left(-\frac{dy}{dt}\right) dt = \int_0^{tf} (-x_2(t))\, dt \tag{1.194}$$

The Hamiltonian for this equivalent $J(u(t))$ now reads:

$$\begin{aligned} \mathcal{H}(\underline{x}, u, \underline{\lambda})) &= -x_2(t) + \lambda(t)^T f(x, u) \\ &= -x_2(t) + \lambda_1(t) x_2(t) + \lambda_2(t) u(t) \end{aligned} \tag{1.195}$$

Adjoint equations become:

$$\dot{\lambda}(t) = -\frac{\partial \mathcal{H}}{\partial x} \Leftrightarrow \begin{cases} \dot{\lambda}_1(t) = -\frac{\partial \mathcal{H}}{\partial x_1} = 0 \\ \dot{\lambda}_2(t) = -\frac{\partial \mathcal{H}}{\partial x_2} = 1 - \lambda_1(t) \end{cases} \tag{1.196}$$

Solutions of adjoint equations are the following where c and d are constants:

$$\begin{cases} \lambda_1(t) = c \\ \lambda_2(t) = (1 - c)\, t + d \end{cases} \tag{1.197}$$

Because the final value of $\underline{x}(t_f)$ is no specified, we have $G\left(x(t_f)\right) = 0$ and the transversality condition (1.70) now reads:

$$G\left(x(t_f)\right) = 0 \Rightarrow \lambda(t_f) = \frac{\partial G\left(x(t_f)\right)}{\partial x(t_f)} = 0 \Rightarrow \begin{cases} \lambda_1(t_f) = \frac{\partial\left(-x_1(t_f)\right)}{\partial x_1(t_f)} = 0 \\ \lambda_2(t_f) = \frac{\partial\left(-x_1(t_f)\right)}{\partial x_2(t_f)} = 0 \end{cases} \tag{1.198}$$

Consequently:

$$\begin{cases} c = 0 \\ d = -t_f \end{cases} \Rightarrow \begin{cases} \lambda_1(t) = 0 \\ \lambda_2(t) = t - t_f \end{cases} \tag{1.199}$$

Obviously, we retrieve the same expressions for $\lambda_1(t)$ and $\lambda_2(t)$ than those obtained previously, and we finally get the same bang-bang optimal control.

1.12.3 Example 2

We re-use the preceding example but now we are looking for the optimal control $u(t)$ which enables the mass to cover the maximum distance in a fixed time t_f with the additional constraint that the final velocity is equal to zero:

$$x_2(t_f) = 0 \tag{1.200}$$

The solution of this problem starts as in the previous case and leads to the solution of adjoint equations where c and d are constants:

$$\begin{cases} \lambda_1(t) = c \\ \lambda_2(t) = -ct + d \end{cases} \tag{1.201}$$

The difference when compared with the previous case is that now the final velocity is equal to zero, that is $x_2(t_f) = 0$. Consequently transversality condition (1.70) involves only state x_1 and reads as follows:

$$\lambda(t_f) = \frac{\partial G\left(x(t_f)\right)}{\partial x(t_f)} \Leftrightarrow \lambda_1(t_f) = \frac{\partial\left(-x_1(t_f)\right)}{\partial x_1(t_f)} = -1 \tag{1.202}$$

Taking into account (1.202) into (1.201) leads to:

$$\begin{cases} \lambda_1(t) = -1 \\ \lambda_2(t) = t + d \end{cases} \tag{1.203}$$

The Hamiltonian $\mathcal{H}$ reads as follows:

$$\mathcal{H}(\underline{x}, u, \underline{\lambda})) = \lambda_1(t)x_2(t) + \lambda_2(t)u(t) = -x_2(t) + (t+d)u(t) \tag{1.204}$$

Thus $\frac{\partial \mathcal{H}}{\partial u} = t + d = \lambda_2(t)\ \forall\ 0 \le t \le t_f$ where the value of constant d is not known: it can be either $d < -t_f$, $d \in [-t_f, 0]$ or $d > 0$. Figure 1.1 plots the three possibilities.

- The possibility $d < -t_f$ leads to $u(t) = u_{min}\, \forall t \in [0, t_f]$ according to (1.170), that is $y(t) := x_1(t) = 0.5 u_{min} t^2$ when taking into account initial conditions (1.181). Thus there is no way to achieve the constraint that the velocity is zero at instant t_f and the possibility $d < -t_f$ is ruled out;
- Similarly, the possibility $d > 0$ leads to $u(t) = u_{max}\, \forall t \in [0, t_f]$, that is $y(t) := x_1(t) = 0.5 u_{max} t^2$ when taking into account initial conditions (1.181). Thus the possibility $d > 0$ is also ruled out.

Hence d shall be chosen between $-t_f$ and 0. According to (1.170) and Figure 1.1 we have:

$$u(t) = \begin{cases} u_{max}\ \forall\ 0 \le t \le t_s \\ u_{min}\ \forall\ t_s < t \le t_f \end{cases} \tag{1.205}$$

Instant t_s is the switching instant, that is time at which $\frac{\partial \mathcal{H}}{\partial u} = \lambda_2(t)$ changes in sign. Solving the state equations with initial velocity set to zero yields the

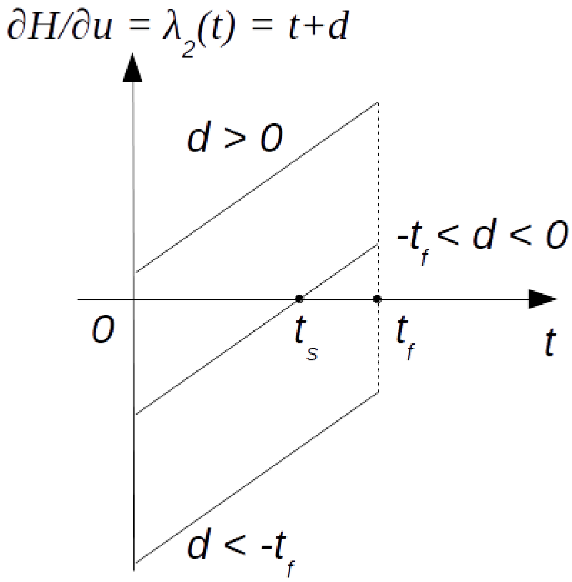


Figure 1.1: Three possibilities for the values of $\partial\mathcal{H}/\partial u = \lambda_2(t)$

expression of $x_2(t)\ \forall\ t_s < t \le t_f$:

$$\begin{array}{l}\left\{\begin{array}{l}\dot{\underline{x}}_2 = u_{max}\ \forall\ 0 \le t \le t_s \\ \dot{\underline{x}}_2 = u_{min}\ \forall\ t_s < t \le t_f \\ x_2(0) = 0\end{array}\right. \\ \Rightarrow \left\{\begin{array}{l}x_2(t_s) = u_{max}t_s \\ x_2(t) = u_{max}t_s + u_{min}(t - t_s)\ \forall\ t_s < t \le t_f\end{array}\right.\end{array} \tag{1.206}$$

Imposing $x_2(t_f) = 0$ leads to the value of the switching instant t_s:

$$\begin{array}{c}x_2(t_f) = 0 \Rightarrow u_{max}t_s + u_{min}(t_f - t_s) = 0 \\ \Rightarrow t_s = \frac{u_{min}t_f}{u_{min}-u_{max}} = -\frac{u_{min}t_f}{u_{max}-u_{min}}\end{array} \tag{1.207}$$

From Figure 1.1 it is clear that at $t = t_s$ we have $\lambda_2(t_s) = 0$. Using the fact that $\lambda_2(t) = t + d$ we finally get the value of constant d:

$$\left\{\begin{array}{l}\lambda_2(t) = t + d \\ \lambda_2(t_s) = 0\end{array}\right. \Rightarrow d = -t_s \tag{1.208}$$

Furthermore the Hamiltonian along the optimal trajectory has the following value:

$$\left\{\begin{array}{rl}\forall\ 0 \le t \le t_s\ \mathcal{H}(\underline{x}, u, \underline{\lambda})) & = \lambda_1(t)x_2(t) + \lambda_2(t)u(t) \\ & = -u_{max}t + (t + d)u_{max} = -t_s u_{max} \\ \forall\ t_s < t \le t_f\ \mathcal{H}(\underline{x}, u, \underline{\lambda})) & = -u_{max}t_s - u_{min}(t - t_s) + (t - t_s)u_{max} \\ & = -t_s u_{max}\end{array}\right. \tag{1.209}$$

As expected the Hamiltonian along the optimal trajectory is constant.

1.13 Singular arc - Legendre-Clebsch condition

The case where $\partial\mathcal{H}/\partial\underline{u}$ does not yield to a definite value for the control $\underline{u}(t)$ is called *singular* control. Usually singular control arises when a multiplier $\sigma(t)$ of the control $\underline{u}(t)$ (which is called the switching function) in the Hamiltonian $\mathcal{H}$ *vanishes over a finite length of time* $t_1 \leq t \leq t_2$:

$$\sigma(t) := \frac{\partial\mathcal{H}}{\partial\underline{u}} = 0 \quad \forall\, t_1 \leq t \leq t_2 \tag{1.210}$$

The *singular* control can be determined by the condition that the switching function $\sigma(t)$ and its time derivatives vanish along the so-called singular arc. Hence over a singular arc we have:

$$\frac{d^k}{dt^k}\sigma(t) = 0 \quad \forall\, t_1 \leq t \leq t_2, \quad \forall\, k \in \mathbb{N} \tag{1.211}$$

At some derivative order the control $\underline{u}(t)$ does appear explicitly and its value is thereby determined. Furthermore it can be shown that the control $\underline{u}(t)$ appears at an even derivative order. So the derivative order at which the control $\underline{u}(t)$ does appear explicitly will be denoted $2q$. Thus:

$$k := 2q \Rightarrow \frac{d^{2q}\,\sigma(t)}{dt^{2q}} := A(t, \underline{x}, \underline{\lambda}) + B(t, \underline{x}, \underline{\lambda})\underline{u} = 0 \tag{1.212}$$

The previous equation gives an explicit equation for the singular control, once the Lagrange multiplier $\underline{\lambda}$ have been obtained through the relation $\dot{\underline{\lambda}}(t) = -\frac{\partial\mathcal{H}}{\partial\underline{x}}$.

The singular arc will be optimal if it satisfies the following generalized Legendre-Clebsch condition, which is also known as the Kelley condition[14], where $2q$ is the (always even) value of k at which the control $\underline{u}(t)$ explicitly appears in $\frac{d^k}{dt^k}\sigma(t)$ for the first time:

$$(-1)^q \frac{\partial}{\partial\underline{u}}\left[\frac{d^{2q}\,\sigma(t)}{dt^{2q}}\right] \geq 0 \tag{1.213}$$

Note that for the *regular* arc the second order necessary condition for optimality to achieve a minimum cost is the positive semi-definiteness of the Hessian matrix of the Hamiltonian along an optimal trajectory. This condition is obtained by setting $q = 0$ in the generalized Legendre-Clebsch condition (1.213):

$$q = 0 \Rightarrow \frac{\partial}{\partial\underline{u}}\sigma(t) = \frac{\partial^2\mathcal{H}}{\partial\underline{u}^2} = \mathcal{H}_{uu} \geq 0 \tag{1.214}$$

This inequality is also termed *regular* Legendre-Clebsch condition.

[14]Douglas M. Pargett & Douglas Mark D. Ardema, Flight Path Optimization at Constant Altitude, Journal of Guidance Control and Dynamics, July 2007, 30(4):1197-1201, DOI: 10.2514/1.28954

Chapter 2

Finite Horizon Linear Quadratic Regulator

2.1 Problem to be solved

The Linear Quadratic Regulator (LQR) is an optimal control problem where the state equation of the plant is linear, the performance index is quadratic and the initial conditions are known. We discuss in this chapter linear quadratic regulation in the case where the final time which appears in the cost to be minimized is finite whereas the next chapter will focus on the infinite horizon case. The optimal control problem to be solved is the following: assume a plant driven by a linear dynamical equation of the form:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{2.1}$$

Where:

- $\mathbf{A}$ is the state (or system) matrix
- $\mathbf{B}$ is the input matrix
- $\underline{x}(t)$ is the state vector of dimension n
- $\underline{u}(t)$ is the control vector of dimension m

Then we have to find the control $\underline{u}(t)$ which minimizes the following quadratic performance index:

$$J(\underline{u}(t)) = \frac{1}{2}\left(\underline{x}(t_f) - \underline{x}_f\right)^T \mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right) + \frac{1}{2}\int_0^{t_f} \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\,dt \tag{2.2}$$

where the final time t_f is set and $\underline{x}_f$ is the final state to be reached. The performance index relates to the fact that a trade-off has been done between the rate of variation of $\underline{x}(t)$ and the magnitude of the control input $\underline{u}(t)$. Matrices

$\mathbf{S}$ and $\mathbf{Q}$ shall be chosen to be symmetric positive semi-definite and matrix $\mathbf{R}$ symmetric positive definite.

$$\begin{cases} \mathbf{S} = \mathbf{S}^T \geq 0 \\ \mathbf{Q} = \mathbf{Q}^T \geq 0 \\ \mathbf{R} = \mathbf{R}^T > 0 \end{cases} \tag{2.3}$$

Notice that the use of matrix $\mathbf{S}$ is optional; indeed, if the final state $\underline{x}_f$ is imposed then there is no need the insert the expression $\frac{1}{2}\left(\underline{x}(t_f) - \underline{x}_f\right)^T \mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right)$ in the cost to be minimized.

2.2 Positive definite and positive semi-definite matrix

A positive definite matrix $\mathbf{M}$ is denoted $\mathbf{M} > 0$. We remind that a real $n \times n$ symmetric matrix $\mathbf{M} = \mathbf{M}^T$ is called positive definite if and only if we have either:

- $\underline{x}^T \mathbf{M} \underline{x} > 0$ for all $\underline{x} \neq 0$;
- All eigenvalues of $\mathbf{M}$ are strictly positive;
- All of the leading principal minors are strictly positive (the leading principal minor of order k is the minor of order k obtained by deleting the last $n - k$ rows and columns);
- Matrix $\mathbf{M}$ can be written as follows where matrix $\mathbf{M}^{0.5}$ is square, symmetric and invertible:

  $$\mathbf{M} = \mathbf{M}^{0.5}\mathbf{M}^{0.5} \text{ where } \left(\mathbf{M}^{0.5}\right)^T = \mathbf{M}^{0.5} \tag{2.4}$$

  Matrix $\mathbf{M}^{0.5}$ is called the root square of matrix $\mathbf{M}$. By getting the modal decomposition of matrix $\mathbf{M}$, that is $\mathbf{M} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$ where $\mathbf{V}$ is the matrix whose columns are the eigenvectors of $\mathbf{M}$ and $\mathbf{D}$ is the diagonal matrix whose diagonal elements are the corresponding positive eigenvalues, the square root $\mathbf{M}^{0.5}$ of $\mathbf{M}$ is given by $\mathbf{M}^{0.5} = \mathbf{V}\mathbf{D}^{0.5}\mathbf{V}^{-1}$, where $\mathbf{D}^{0.5}$ is any diagonal matrix whose elements are the square root of the diagonal elements of $\mathbf{D}$[1].

Similarly a semi-definite positive matrix $\mathbf{M}$ is denoted $\mathbf{M} \geq 0$. We remind that a $n \times n$ real symmetric matrix $\mathbf{M} = \mathbf{M}^T$ is called positive semi-definite if and only if we have either:

- $\underline{x}^T \mathbf{M} \underline{x} \geq 0$ for all $\underline{x} \neq 0$;
- All eigenvalues of $\mathbf{M}$ are non-negative;
- All of the principal (not only leading) minors are non-negative (the principal minor of order k is the minor of order k obtained by deleting $n - k$ rows and the $n - k$ columns with the same position than the rows. For instance, in a principal minor where you have deleted rows 1 and 3, you should also delete columns 1 and 3);

[1] https://en.wikipedia.org/wiki/Square_root_of_a_matrix

- Matrix $\mathbf{M}$ can be written as $\left(\mathbf{M}^{0.5}\right)^T \mathbf{M}^{0.5}$ where matrix $\mathbf{M}^{0.5}$ is full row rank.

Furthermore a real symmetric matrix $\mathbf{M}$ is called negative (semi-)definite if $-\mathbf{M}$ is positive (semi-)definite.

Example 2.1. *Check that* $\mathbf{M}_1 = \mathbf{M}_1^T = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$ *is not positive definite and that* $\mathbf{M}_2 = \mathbf{M}_2^T = \begin{bmatrix} 1 & -2 \\ -2 & 5 \end{bmatrix}$ *is positive definite.* ■

2.3 Hamiltonian matrix

For this optimal control problem, the Hamiltonian (1.63) reads:

$$\mathcal{H}(\underline{x}, \underline{u}, \underline{\lambda}) = \frac{1}{2}\left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right) + \underline{\lambda}^T(t)\left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right) \tag{2.5}$$

The necessary condition for optimality (1.72) yields:

$$\frac{\partial \mathcal{H}}{\partial \underline{u}} = \mathbf{R}\underline{u}(t) + \mathbf{B}^T \underline{\lambda}(t) = \underline{0} \tag{2.6}$$

Taking into account that $\mathbf{R}$ is a symmetric matrix, we get:

$$\underline{u}(t) = -\mathbf{R}^{-1}\mathbf{B}^T \underline{\lambda}(t) \tag{2.7}$$

Eliminating $\underline{u}(t)$ in equation (2.1) reads:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \underline{\lambda}(t) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{2.8}$$

The dynamics of Lagrange multipliers $\underline{\lambda}(t)$ is given by (see (1.69)):

$$\dot{\underline{\lambda}}(t) = -\frac{\partial \mathcal{H}}{\partial \underline{x}} = -\mathbf{Q}\underline{x}(t) - \mathbf{A}^T \underline{\lambda}(t) \tag{2.9}$$

The final values of the Lagrange multipliers are given by (1.70). Using the fact that $\mathbf{S}$ is a symmetric matrix we get:

$$\underline{\lambda}(t_f) = \frac{\partial}{\partial \underline{x}(t_f)}\left(\frac{1}{2}\left(\underline{x}(t_f) - \underline{x}_f\right)^T \mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right)\right) = \mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right) \tag{2.10}$$

Taking into account that matrices $\mathbf{Q}$ and $\mathbf{S}$ are symmetric matrices, equations (2.9) and (2.10) are written as follows:

$$\begin{cases} \dot{\underline{\lambda}}(t) = -\mathbf{Q}\underline{x}(t) - \mathbf{A}^T \underline{\lambda}(t) \\ \underline{\lambda}(t_f) = \mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right) \end{cases} \tag{2.11}$$

Equations (2.8) and (2.11) represent a two-point boundary value problem. Combining (2.8) and (2.11) into a single state equation yields:

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\underline{\lambda}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} = \mathbf{H} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} \tag{2.12}$$

where we have introduced the Hamiltonian matrix $\mathbf{H}$ defined by:

$$\boxed{\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix}} \tag{2.13}$$

By definition, a matrix $\mathbf{H}$ is said to be an Hamiltonian matrix as soon as the following property holds:

$$(\mathbf{J}\mathbf{H})^T = \mathbf{J}\mathbf{H} \tag{2.14}$$

where $\mathbf{J}$ is the following skew-symmetric matrix:

$$\mathbf{J} = -\mathbf{J}^T = \begin{bmatrix} \mathbf{0} & \mathbb{I} \\ -\mathbb{I} & \mathbf{0} \end{bmatrix} \tag{2.15}$$

2.4 Optimal control

2.4.1 State vector expression

Solving (2.12) yields:

$$\begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} = e^{\mathbf{H}t} \begin{bmatrix} \underline{x}(0) \\ \underline{\lambda}(0) \end{bmatrix} \tag{2.16}$$

In the previous equation, the value of $\underline{\lambda}(0)$ is not known. On the other hand, $\underline{x}(t_f)$ or $\underline{\lambda}(t_f)$ is known, depending on whether the final state is imposed or weighted. Thus by replacing t by $t - t_f$ in the previous equation we obtain:

$$\begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} = e^{\mathbf{H}t} \begin{bmatrix} \underline{x}(0) \\ \underline{\lambda}(0) \end{bmatrix} = e^{\mathbf{H}(t-t_f)} \begin{bmatrix} \underline{x}(t_f) \\ \underline{\lambda}(t_f) \end{bmatrix} \tag{2.17}$$

Then exponential matrix $e^{\mathbf{H}(t-t_f)}$ is partitioned as follows:

$$\boxed{e^{\mathbf{H}(t-t_f)} := \begin{bmatrix} \mathbf{Y}_1(t) & \mathbf{X}_1(t) \\ \mathbf{Y}_2(t) & \mathbf{X}_2(t) \end{bmatrix}} \tag{2.18}$$

Then (2.17) yields:

$$\begin{cases} \underline{x}(t) = \mathbf{Y}_1(t)\, \underline{x}(t_f) + \mathbf{X}_1(t)\, \underline{\lambda}(t_f) \\ \underline{\lambda}(t) = \mathbf{Y}_2(t)\, \underline{x}(t_f) + \mathbf{X}_2(t)\, \underline{\lambda}(t_f) \end{cases} \tag{2.19}$$

Furthermore notice the following relations obtained when $t = t_f$:

$$e^{\mathbf{H}(t-t_f)}\Big|_{t=tf} = \mathbb{I} := \begin{bmatrix} \mathbf{Y}_1(t_f) & \mathbf{X}_1(t_f) \\ \mathbf{Y}_2(t_f) & \mathbf{X}_2(t_f) \end{bmatrix} \Rightarrow \begin{cases} \mathbf{Y}_1(t_f) = \mathbf{X}_2(t_f) = \mathbb{I} \\ \mathbf{X}_1(t_f) = \mathbf{Y}_2(t_f) = \mathbf{0} \end{cases} \tag{2.20}$$

2.4.2 Lagrange multipliers for imposed final state

Assume that the final state $\underline{x}(t_f)$ is imposed:

$$\underline{x}(t_f) := \underline{x}_f \tag{2.21}$$

Then (2.19) can be manipulated to get rid of the unknown vector $\underline{\lambda}(t_f)$:

$$\begin{cases} \underline{\lambda}(t_f) = \mathbf{X}_1^{-1}(t)\,\left(\underline{x}(t) - \mathbf{Y}_1(t)\,\underline{x}_f\right) \\ \underline{\lambda}(t_f) = \mathbf{X}_2^{-1}(t)\,\left(\underline{\lambda}(t) - \mathbf{Y}_2(t)\,\underline{x}_f\right) \end{cases} \tag{2.22}$$

Then equating $\underline{\lambda}(t_f) = \underline{\lambda}(t_f)$ we get:

$$\begin{array}{l} \mathbf{X}_2^{-1}(t)\,\left(\underline{\lambda}(t) - \mathbf{Y}_2(t)\,\underline{x}_f\right) = \mathbf{X}_1^{-1}(t)\,\left(\underline{x}(t) - \mathbf{Y}_1(t)\,\underline{x}_f\right) \\ \Leftrightarrow \underline{\lambda}(t) = \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\,\left(\underline{x}(t) - \mathbf{Y}_1(t)\,\underline{x}_f\right) + \mathbf{Y}_2(t)\,\underline{x}_f \\ \Leftrightarrow \underline{\lambda}(t) = \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\,\underline{x}(t) - \left(\mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\,\mathbf{Y}_1(t) - \mathbf{Y}_2(t)\right)\,\underline{x}_f \end{array} \tag{2.23}$$

In order to factor $\underline{x}(t)$ and $\underline{x}_f$, let $\mathbf{P}(t)$ and $\mathbf{F}(t)$ be the following matrices:

$$\begin{cases} \mathbf{P}(t) := \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t) \\ \mathbf{F}(t) := \mathbf{P}(t)\,\mathbf{Y}_1(t) - \mathbf{Y}_2(t) \end{cases} \tag{2.24}$$

We finally get:

$$\boxed{\underline{\lambda}(t) = \mathbf{P}(t)\,\underline{x}(t) - \mathbf{F}(t)\,\underline{x}_f} \tag{2.25}$$

2.4.3 Lagrange multipliers for weighted final state

In the case where final state $\underline{x}(t_f)$ is expected to be close to the final value $\underline{x}_f$ then the final condition $\underline{\lambda}(t_f)$ is given by (2.10):

$$\underline{\lambda}(t_f) = \mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right) \tag{2.26}$$

Then (2.19) can be manipulated to get rid of the unknown vector $\underline{x}(t_f)$:

$$\begin{array}{rl} \underline{x}(t) & = \mathbf{Y}_1(t)\,\underline{x}(t_f) + \mathbf{X}_1(t)\,\underline{\lambda}(t_f) \\ & = \mathbf{Y}_1(t)\,\underline{x}(t_f) + \mathbf{X}_1(t)\,\mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right) \\ & = \left(\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S}\right)\,\underline{x}(t_f) - \mathbf{X}_1(t)\,\mathbf{S}\,\underline{x}_f \\ \Rightarrow \underline{x}(t_f) & = \left(\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S}\right)^{-1}\,\left(\underline{x}(t) + \mathbf{X}_1(t)\,\mathbf{S}\,\underline{x}_f\right) \\ \text{and} \quad \underline{\lambda}(t) & = \mathbf{Y}_2(t)\,\underline{x}(t_f) + \mathbf{X}_2(t)\,\underline{\lambda}(t_f) \\ & = \mathbf{Y}_2(t)\,\underline{x}(t_f) + \mathbf{X}_2(t)\,\mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right) \\ & = \left(\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S}\right)\,\underline{x}(t_f) - \mathbf{X}_2(t)\,\mathbf{S}\,\underline{x}_f \\ \Rightarrow \underline{x}(t_f) & = \left(\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S}\right)^{-1}\,\left(\underline{\lambda}(t) + \mathbf{X}_2(t)\,\mathbf{S}\,\underline{x}_f\right) \end{array} \tag{2.27}$$

Then equating $\underline{x}(t_f) = \underline{x}(t_f)$:

$$\begin{array}{l} \left(\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S}\right)^{-1}\,\left(\underline{\lambda}(t) + \mathbf{X}_2(t)\,\mathbf{S}\,\underline{x}_f\right) \\ \qquad = \left(\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S}\right)^{-1}\,\left(\underline{x}(t) + \mathbf{X}_1(t)\,\mathbf{S}\,\underline{x}_f\right) \end{array} \tag{2.28}$$

Thus:

$$\underline{\lambda}(t) = (\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S})\left((\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S})^{-1}\left(\underline{x}(t) + \mathbf{X}_1(t)\,\mathbf{S}\,\underline{x}_f\right)\right) - \mathbf{X}_2(t)\,\mathbf{S}\,\underline{x}_f \tag{2.29}$$

In order to factor $\underline{x}(t)$ and $\underline{x}_f$, let $\mathbf{P}_S(t)$ and $\mathbf{F}_S(t)$ be the following matrices:

$$\left\{\begin{array}{l} \mathbf{P}_S(t) := (\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S})\;(\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S})^{-1} \\ \mathbf{F}_S(t) := (\mathbf{X}_2(t) - \mathbf{P}_S(t)\,\mathbf{X}_1(t))\;\mathbf{S} \end{array}\right. \tag{2.30}$$

We finally get:

$$\boxed{\underline{\lambda}(t) = \mathbf{P}_S(t)\,\underline{x}(t) - \mathbf{F}_S(t)\,\underline{x}_f} \tag{2.31}$$

2.4.4 Limit values when final state weighting matrix increases

In order to assess what happen when the final state weighting matrix $\|\mathbf{S}\| \to \infty$, we first recall the Neumann series:

$$(\mathbb{I} - \mathbf{T})^{-1} = \sum_{k=0}^{\infty} \mathbf{T}^k \tag{2.32}$$

Applying this result to the right term of $\mathbf{P}_S(t)$ reads:

$$\begin{array}{rl} (\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S})^{-1} & = \left(\mathbb{I} + \left(\mathbf{Y}_1(t)\,(\mathbf{X}_1(t)\,\mathbf{S})^{-1}\right)\mathbf{X}_1(t)\,\mathbf{S}\right)^{-1} \\ & = (\mathbf{X}_1(t)\,\mathbf{S})^{-1}\left(\mathbb{I} + \mathbf{Y}_1(t)\,(\mathbf{X}_1(t)\,\mathbf{S})^{-1}\right)^{-1} \\ & = (\mathbf{X}_1(t)\,\mathbf{S})^{-1}\,\sum_{k=0}^{\infty}\left(-\mathbf{Y}_1(t)\,(\mathbf{X}_1(t)\,\mathbf{S})^{-1}\right)^k \\ & \approx (\mathbf{X}_1(t)\,\mathbf{S})^{-1}\left(\mathbb{I} - \mathbf{Y}_1(t)\,(\mathbf{X}_1(t)\,\mathbf{S})^{-1}\right) \\ & = \mathbf{S}^{-1}\,\mathbf{X}_1^{-1}(t)\left(\mathbb{I} - \mathbf{Y}_1(t)\,(\mathbf{X}_1(t)\,\mathbf{S})^{-1}\right) \end{array} \tag{2.33}$$

Thus $\mathbf{P}_S(t)$ can be approximated as follows when $\|\mathbf{S}\| \to \infty$:

$$\begin{array}{rl} \mathbf{P}_S(t) & \approx (\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S})\;\mathbf{S}^{-1}\,\mathbf{X}_1^{-1}(t)\left(\mathbb{I} - \mathbf{Y}_1(t)\,(\mathbf{X}_1(t)\,\mathbf{S})^{-1}\right) \\ & \approx \left(\mathbf{Y}_2(t)\,\mathbf{S}^{-1}\,\mathbf{X}_1^{-1}(t) + \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\right)\left(\mathbb{I} - \mathbf{Y}_1(t)\,(\mathbf{X}_1(t)\,\mathbf{S})^{-1}\right) \\ & \approx \left(\mathbf{Y}_2(t)\,\mathbf{S}^{-1}\,\mathbf{X}_1^{-1}(t) + \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\right)\left(\mathbb{I} - \mathbf{Y}_1(t)\,\mathbf{S}^{-1}\,\mathbf{X}_1^{-1}(t)\right) \end{array} \tag{2.34}$$

When $\|\mathbf{S}\| \to \infty$ we retrieve the expression of $\mathbf{P}(t)$ in (2.24) by using the order 0 approximation of $\mathbf{P}_S(t)$. Indeed:

$$\|\mathbf{S}\| \to \infty \Rightarrow \mathbf{P}_S(t) \approx \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\,\mathbb{I} = \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t) = \mathbf{P}(t) \tag{2.35}$$

As far as $\mathbf{F}_S(t)$ is concerned, we also retrieve the expression of $\mathbf{F}(t)$ in (2.24)

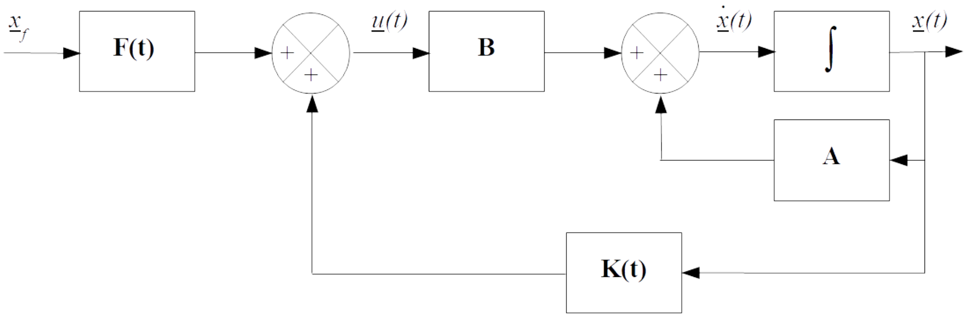


Figure 2.1: Finite horizon closed-loop optimal control

when $\|\mathbf{S}\| \to \infty$ by using the order 1 approximation of $\mathbf{P}_S(t)$. Indeed:

$$\begin{array}{rl} \|\mathbf{S}\| \to \infty \Rightarrow \mathbf{P}_S(t)\,\mathbf{X}_1(t) & \approx \left(\mathbf{Y}_2(t)\,\mathbf{S}^{-1}\,\mathbf{X}_1^{-1}(t) + \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\right)\left(\mathbf{X}_1(t) - \mathbf{Y}_1(t)\,\mathbf{S}^{-1}\right) \\ & \approx \mathbf{Y}_2(t)\,\mathbf{S}^{-1} + \mathbf{X}_2(t) - \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\,\mathbf{Y}_1(t)\,\mathbf{S}^{-1} \\ \Rightarrow \mathbf{X}_2(t) - \mathbf{P}_S(t)\,\mathbf{X}_1(t) & \approx \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\,\mathbf{Y}_1(t)\,\mathbf{S}^{-1} - \mathbf{Y}_2(t)\,\mathbf{S}^{-1} \\ \Rightarrow \mathbf{F}_S(t) & := (\mathbf{X}_2(t) - \mathbf{P}_S(t)\,\mathbf{X}_1(t))\ \mathbf{S} \\ & \approx \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)\,\mathbf{Y}_1(t) - \mathbf{Y}_2(t) \\ & = \mathbf{P}(t)\,\mathbf{Y}_1(t) - \mathbf{Y}_2(t) \\ & = \mathbf{F}(t) \end{array} \tag{2.36}$$

2.4.5 Closed-loop block diagram

Finally, using (2.7), optimal control $\underline{u}(t)$ reads as follows when the final state is imposed (when the final state is weighted, $\mathbf{P}(t)$ and $\mathbf{F}(t)$ have to be replaced by $\mathbf{P}_S(t)$ and $\mathbf{F}_S(t)$, respectively):

$$\begin{array}{rl} \underline{u}(t) & = -\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \\ & = -\mathbf{R}^{-1}\mathbf{B}^T\left(\mathbf{P}(t)\,\underline{x}(t) - \mathbf{F}(t)\,\underline{x}_f\right) \\ & := -\mathbf{K}(t)\underline{x}(t) + \mathbf{F}(t)\,\underline{x}_f \end{array} \tag{2.37}$$

where:

$$\mathbf{K}(t) = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t) \tag{2.38}$$

The preceding expression leads to the closed-loop block diagram shown in Figure 2.1.

It is worth noticing that $\mathbf{P}(t_f) = \mathbf{X}_2(t_f)\mathbf{X}_1^{-1}(t_f) \to \infty$ because $\mathbf{X}_1(t_f) = \mathbf{0}$ when the final value $\underline{x}_f$ of $\underline{x}(t_f)$ is imposed, as indicated by (2.20). This is in line with the final value of $\mathbf{P}(t)$ as indicated by (2.10) when the final state is close to zero:

$$\underline{x}_f = \underline{0} \Rightarrow \underline{\lambda}(t_f) = \mathbf{P}(t_f)\underline{x}(t_f) = \mathbf{S}\underline{x}(t_f) \Rightarrow \mathbf{P}(t_f) = \mathbf{S} \tag{2.39}$$

Consequently, when it is desired that the final value $\underline{x}(t_f)$ tends towards $\underline{x}_f$, then $\mathbf{S} \to \infty$. Thus $\mathbf{S} = \mathbf{P}(t_f)$ is singular when the final value $\underline{x}(t_f)$ is set to $\underline{x}_f$. In that case, and to avoid the numerical difficulty when $t = t_f$, we shall set $\underline{u}(t_f) = \underline{0}$. Thus the optimal control reads:

$$\underline{u}(t) = \begin{cases} -\mathbf{K}(t)\,\underline{x}(t) + \mathbf{F}(t)\,\underline{x}_f & \forall\ 0 \le t < t_f \\ \underline{0}\ \text{ for } t = t_f \end{cases} \tag{2.40}$$

2.5 Riccati differential equation

When the final value $\underline{x}_f$ is set to zero, we have seen that that Lagrange multipliers $\underline{\lambda}(t)$ linearly depend on the state vector $\underline{x}(t)$ through the time dependent matrix $\mathbf{P}(t)$:

$$\underline{x}_f = \underline{0} \Rightarrow \underline{\lambda}(t) = \mathbf{P}(t)\underline{x}(t) \tag{2.41}$$

Using (2.9) and (2.10), we can compute the time derivative of the Lagrange multipliers $\underline{\lambda}(t) = \mathbf{P}(t)\underline{x}(t)$ as follows:

$$\begin{cases} \dot{\underline{\lambda}}(t) = \dot{\mathbf{P}}(t)\underline{x}(t) + \mathbf{P}(t)\dot{\underline{x}}(t) = -\mathbf{Q}\underline{x}(t) - \mathbf{A}^T\underline{\lambda}(t) \\ \underline{\lambda}(t_f) = \mathbf{P}(t_f)\underline{x}(t_f) = \mathbf{S}\underline{x}(t_f) \end{cases} \tag{2.42}$$

Then substituting (2.1), (2.7) and (2.41) within (2.42) we get:

$$\begin{array}{l} \begin{cases} \dot{\underline{x}} = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{u}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \\ \underline{\lambda}(t) = \mathbf{P}(t)\underline{x}(t) \end{cases} \\ \Rightarrow \begin{cases} \dot{\mathbf{P}}(t)\underline{x}(t) + \mathbf{P}(t)\left(\mathbf{A}\underline{x}(t) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)\underline{x}(t)\right) = -\mathbf{Q}\underline{x}(t) - \mathbf{A}^T\mathbf{P}(t)\underline{x}(t) \\ \mathbf{P}(t_f)\underline{x}(t_f) = \mathbf{S}\underline{x}(t_f) \end{cases} \end{array} \tag{2.43}$$

Because the previous equation is true for all $\underline{x}(t)$ and $\underline{x}(t_f)$ we obtain the following equation, which is known as the *Riccati differential equation*:

$$\boxed{\begin{cases} \mathbf{A}^T\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A} - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t) + \mathbf{Q} = -\dot{\mathbf{P}}(t) \\ \mathbf{P}(t_f) = \mathbf{S} \end{cases}} \tag{2.44}$$

From a computational point of view, the *Riccati differential equation* (2.44) may be integrated backward. The kernel $\mathbf{P}(t)$ is stored for each values of t and then is used to compute $\mathbf{K}(t)$ and $\underline{u}(t)$.

Alternatively, the analytic solution of the *Riccati differential equation* (2.44) is given either by $\mathbf{P}(t) := \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t)$ in (2.24) when the final state $\underline{x}_f = \underline{0}$ is imposed or by $\mathbf{P}_S(t) := (\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S})\,(\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S})^{-1}$ in (2.30) when the final state $\underline{x}_f = \underline{0}$ is weighted by matrix $\mathbf{S} = \mathbf{S}^T \geq 0$. The key point to solve the *Riccati differential equation* is the partition of matrix $e^{\mathbf{H}(t-t_f)}$ shown in (2.18):

$$e^{\mathbf{H}(t-t_f)} := \begin{bmatrix} \mathbf{Y}_1(t) & \mathbf{X}_1(t) \\ \mathbf{Y}_2(t) & \mathbf{X}_2(t) \end{bmatrix} \tag{2.45}$$

It is worth noticing that the Riccati differential equation can be written in a compact form as follows where $\mathbf{H}$ denotes the Hamiltonian matrix defined in (2.13):

$$\begin{array}{r} -\dot{\mathbf{P}}(t) = \mathbf{A}^T\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A} - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t) + \mathbf{Q} \\ \Leftrightarrow -\dot{\mathbf{P}}(t) = \begin{bmatrix} \mathbf{P}(t) & -\mathbb{I}_n \end{bmatrix} \mathbf{H} \begin{bmatrix} \mathbb{I}_n \\ \mathbf{P}(t) \end{bmatrix} \end{array} \tag{2.46}$$

2.6 Examples

2.6.1 Example 1

Given the following scalar plant:

$$\left\{ \begin{array}{l} \dot{x}(t) = ax(t) + bu(t) \\ x(0) = x_0 \end{array} \right. \tag{2.47}$$

Find control $u(t)$ which minimizes the following performance index where $x_f = 0$, $S \geq 0$ and $\rho > 0$:

$$J(u(t)) = \frac{1}{2} x^T(t_f) S x(t_f) + \frac{1}{2} \int_0^{t_f} \rho u^2(t)\, dt \tag{2.48}$$

Hamiltonian matrix $\mathbf{H}$ defined in (2.13) reads:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} = \begin{bmatrix} a & \frac{-b^2}{\rho} \\ 0 & -a \end{bmatrix} \tag{2.49}$$

Denoting by s the Laplace variable, the exponential of matrix $e^{\mathbf{H}t}$ is obtained thanks to the inverse of the Laplace transform, which is denoted $\mathcal{L}^{-1}$:

$$\begin{array}{rl} e^{\mathbf{H}t} & = \mathcal{L}^{-1}\left((sI - \mathbf{H})^{-1}\right) = \mathcal{L}^{-1}\left(\begin{bmatrix} s-a & b^2/\rho \\ 0 & s+a \end{bmatrix}^{-1}\right) \\ & = \mathcal{L}^{-1}\left(\frac{1}{(s-a)(s+a)} \begin{bmatrix} s+a & -b^2/\rho \\ 0 & s-a \end{bmatrix}\right) \\ & = \mathcal{L}^{-1}\left(\begin{bmatrix} \frac{1}{s-a} & \frac{-b^2}{\rho(s^2-a^2)} \\ 0 & \frac{1}{s+a} \end{bmatrix}\right) \\ & = \begin{bmatrix} e^{at} & \frac{-b^2\left(e^{at}-e^{-at}\right)}{2\rho a} \\ 0 & e^{-at} \end{bmatrix} \end{array} \tag{2.50}$$

Following (2.18), the partition of $e^{\mathbf{H}(t-t_f)}$ reads:

$$e^{\mathbf{H}(t-t_f)} = \begin{bmatrix} e^{a(t-t_f)} & \frac{-b^2\left(e^{a(t-t_f)}-e^{-a(t-t_f)}\right)}{2\rho a} \\ 0 & e^{-a(t-t_f)} \end{bmatrix} := \begin{bmatrix} \mathbf{Y}_1(t) & \mathbf{X}_1(t) \\ \mathbf{Y}_2(t) & \mathbf{X}_2(t) \end{bmatrix} \tag{2.51}$$

Because the final state state $\underline{x}_f = \underline{0}$ is weighted by matrix $S \geq 0$, we finally get the solution of the Riccati differential equation thanks to $\mathbf{P}_S(t)$ in (2.30):

$$\begin{array}{rl} \mathbf{P}_S(t) & := \left(\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S}\right) \left(\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S}\right)^{-1} \\ & = e^{-a(t-t_f)}\, S \left(e^{a(t-t_f)} + \frac{-b^2\left(e^{a(t-t_f)}-e^{-a(t-t_f)}\right)}{2\rho a} S\right)^{-1} \\ & = \frac{Se^{-a\left(t-t_f\right)}}{e^{a\left(t-t_f\right)} - \frac{Sb^2\left(e^{a\left(t-t_f\right)}-e^{-a\left(t-t_f\right)}\right)}{2\rho a}} \\ & = \frac{S}{e^{2a\left(t-t_f\right)} + \frac{Sb^2\left(1-e^{2a\left(t-t_f\right)}\right)}{2\rho a}} \end{array} \tag{2.52}$$

Finally the optimal control reads:

$$\begin{aligned} u(t) &= -\mathbf{K}(t)x(t) \\ &= -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_S(t)x(t) \\ &= -\tfrac{b}{\rho}\mathbf{P}_S(t)x(t) \\ &= \frac{-bS}{\rho e^{2a\left(t-t_f\right)}+\frac{Sb^2\left(1-e^{2a\left(t-t_f\right)}\right)}{2a}}x(t) \end{aligned} \tag{2.53}$$

If we want to ensure that the optimal control drives $x(t_f)$ exactly to $x_f = 0$, we let $S \to \infty$ to weight heavily $x(t_f)$ in the performance index $J(u(t))$. Then:

$$\mathbf{P}_S(t) \underbrace{\rightarrow}_{S\to\infty} \mathbf{P}(t) = \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(t) = \frac{2\rho a}{b^2\left(1-e^{2a\left(t-t_f\right)}\right)} \tag{2.54}$$

and:

$$u(t) = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)x(t) = \frac{-2a}{b\left(1-e^{2a\left(t-t_f\right)}\right)}x(t) \tag{2.55}$$

2.6.2 Example 2

Given the following plant, which actually represents a double integrator:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \underline{u}(t) \tag{2.56}$$

Find control $\underline{u}(t)$ which minimizes the following performance index where $\underline{x}_f = \underline{0}$ and $\mathbf{S} = \mathbf{S}^T \geq 0$:

$$J(\underline{u}(t)) = \frac{1}{2}\underline{x}^T(t_f)\mathbf{S}\underline{x}(t_f) + \frac{1}{2}\int_0^{t_f} \underline{u}^2(t)dt \tag{2.57}$$

Weighting matrix $\mathbf{S}$ reads as follows:

$$\mathbf{S} = \mathbf{S}^T = \begin{bmatrix} s_p & 0 \\ 0 & s_v \end{bmatrix} \geq 0 \tag{2.58}$$

Hamiltonian matrix $\mathbf{H}$ defined in (2.13) reads:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} \tag{2.59}$$

In order to compute $e^{\mathbf{H}t}$ we use the following relation where $\mathcal{L}^{-1}$ stands for the inverse Laplace transform:

$$e^{\mathbf{H}t} = \mathcal{L}^{-1}\left[(s\mathbb{I} - \mathbf{H})^{-1}\right] \tag{2.60}$$

We get:

$$
\begin{array}{l}
s\mathbb{I} - \mathbf{H} = \begin{bmatrix} s & -1 & 0 & 0 \\ 0 & s & 0 & 1 \\ 0 & 0 & s & 0 \\ 0 & 0 & 1 & s \end{bmatrix} \Rightarrow (s\mathbb{I} - \mathbf{H})^{-1} = \begin{bmatrix} \frac{1}{s} & \frac{1}{s^2} & \frac{1}{s^4} & -\frac{1}{s^3} \\ 0 & \frac{1}{s} & \frac{1}{s^3} & -\frac{1}{s^2} \\ 0 & 0 & \frac{1}{s} & 0 \\ 0 & 0 & -\frac{1}{s^2} & \frac{1}{s} \end{bmatrix} \\
\Rightarrow e^{\mathbf{H}t} = \mathcal{L}^{-1}\left[(s\mathbb{I} - \mathbf{H})^{-1} \right] = \begin{bmatrix} 1 & t & \frac{t^3}{6} & -\frac{t^2}{2} \\ 0 & 1 & \frac{t^2}{2} & -t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -t & 1 \end{bmatrix}
\end{array}
\tag{2.61}
$$

Following (2.18), the partition of $e^{\mathbf{H}(t-t_f)}$ reads:

$$
e^{\mathbf{H}(t-t_f)} = \left[\begin{array}{cc|cc} 1 & (t-t_f) & \frac{(t-t_f)^3}{6} & -\frac{(t-t_f)^2}{2} \\ 0 & 1 & \frac{(t-t_f)^2}{2} & -(t-t_f) \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & -(t-t_f) & 1 \end{array} \right] := \begin{bmatrix} \mathbf{Y}_1(t) & \mathbf{X}_1(t) \\ \mathbf{Y}_2(t) & \mathbf{X}_2(t) \end{bmatrix} \tag{2.62}
$$

Because the final state state $\underline{x}_f = \underline{0}$ is weighted by matrix $\mathbf{S} \geq 0$, we finally get the solution of the Riccati differential equation thanks to $\mathbf{P}_S(t)$ in (2.30):

$$
\begin{aligned}
\mathbf{P}_S(t) &:= (\mathbf{Y}_2(t) + \mathbf{X}_2(t)\,\mathbf{S})\ (\mathbf{Y}_1(t) + \mathbf{X}_1(t)\,\mathbf{S})^{-1} \\
&= \begin{bmatrix} s_p & 0 \\ -s_p(t-t_f) & s_v \end{bmatrix} \begin{bmatrix} 1 + s_p\frac{(t-t_f)^3}{6} & t - t_f - s_v\frac{(t-t_f)^2}{2} \\ s_p\frac{(t-t_f)^2}{2} & 1 - s_v(t-t_f) \end{bmatrix}^{-1} \\
&= \frac{1}{\Delta} \begin{bmatrix} s_p & 0 \\ -s_p(t-t_f) & s_v \end{bmatrix} \begin{bmatrix} 1 - s_v(t-t_f) & t_f - t + s_v\frac{(t-t_f)^2}{2} \\ -s_p\frac{(t-t_f)^2}{2} & 1 + s_p\frac{(t-t_f)^3}{6} \end{bmatrix}
\end{aligned}
\tag{2.63}
$$

where:

$$
\Delta = \left(1 + s_p\frac{(t-t_f)^3}{6}\right)(1 - s_v(t-t_f)) - s_p\frac{(t-t_f)^2}{2}\left(t - t_f - s_v\frac{(t-t_f)^2}{2}\right) \tag{2.64}
$$

2.7 Second order necessary condition for optimality

It is worth noticing that the second order necessary condition for optimality to achieve a minimum cost is the positive semi-definiteness of the Hessian matrix of the Hamiltonian along an optimal trajectory (see (1.214)). This condition is always satisfied as soon as $\mathbf{R} > 0$. Indeed we get from (2.6):

$$
\frac{\partial^2 H}{\partial \underline{u}^2} = \mathbf{H}_{uu} = \mathbf{R} > 0 \tag{2.65}
$$

2.8 Minimum cost achieved

The minimum cost achieved is given by:

$$
J^* = J(\underline{u}^*(t)) = \frac{1}{2}\underline{x}^T(0)\mathbf{P}(0)\underline{x}(0) \tag{2.66}
$$

Indeed, from the Riccati equation (2.44), we deduce that:

$$\begin{array}{l} \underline{x}^T \left(\dot{\mathbf{P}} + \mathbf{PA} + \mathbf{A}^T\mathbf{P} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q}\right) \underline{x} = 0 \\ \Leftrightarrow \underline{x}^T\dot{\mathbf{P}}\underline{x} + \underline{x}^T\mathbf{PA}\underline{x} + \underline{x}^T\mathbf{A}^T\mathbf{P}\underline{x} - \underline{x}^T\mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P}\underline{x} + \underline{x}^T\mathbf{Q}\underline{x} = 0 \\ \Leftrightarrow \underline{x}^T\dot{\mathbf{P}}\underline{x} + \underline{x}^T\mathbf{PA}\underline{x} + \left(\underline{x}^T\mathbf{PA}\underline{x}\right)^T - \underline{x}^T\mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P}\underline{x} + \underline{x}^T\mathbf{Q}\underline{x} = 0 \end{array} \tag{2.67}$$

Taking into account the fact that $\mathbf{P} = \mathbf{P}^T > 0$, $\mathbf{R} = \mathbf{R}^T > 0$ as well as (2.1), (2.37) with $\underline{x}_f = \underline{0}$ and (2.38) it can be shown that:

$$\begin{array}{rl} \left\{ \begin{array}{rl} \underline{x}^T\mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P}\underline{x} & = -\underline{x}^T\mathbf{PB}\underline{u}^* \\ & = -\underline{x}^T\mathbf{PBR}^{-1}\mathbf{R}\underline{u}^* \\ & = \underline{u}^{*T}\mathbf{R}\underline{u}^* \\ \underline{x}^T\mathbf{PA}\underline{x} & = \underline{x}^T\mathbf{P}\left(\mathbf{A}\underline{x} + \mathbf{B}\underline{u}^* - \mathbf{B}\underline{u}^*\right) \\ & = \underline{x}^T\mathbf{P}\dot{\underline{x}} - \underline{x}^T\mathbf{PB}\underline{u}^* \\ & = \underline{x}^T\mathbf{P}\dot{\underline{x}} + \underline{u}^{*T}\mathbf{R}\underline{u}^* \end{array} \right. & \\ \Rightarrow \underline{x}^T\dot{\mathbf{P}}\underline{x} + \underline{x}^T\mathbf{PA}\underline{x} & + \left(\underline{x}^T\mathbf{PA}\underline{x}\right)^T - \underline{x}^T\mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P}\underline{x} \\ & = \underline{x}^T\dot{\mathbf{P}}\underline{x} + \underline{x}^T\mathbf{P}\dot{\underline{x}} + \dot{\underline{x}}^T\mathbf{P}\underline{x} + \underline{u}^{*T}\mathbf{R}\underline{u}^* \\ & = \frac{d}{dt}\left(\underline{x}^T\mathbf{P}\underline{x}\right) + \underline{u}^{*T}\mathbf{R}\underline{u}^* \end{array} \tag{2.68}$$

As a consequence equation (2.67) can be written as follows:

$$\frac{d}{dt}\left(\underline{x}^T(t)\mathbf{P}(t)\underline{x}(t)\right) + \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^{*T}(t)\mathbf{R}\underline{u}^*(t) = 0 \tag{2.69}$$

And the performance index (2.2) to be minimized can be re-written as:

$$\begin{array}{r} J(\underline{u}^*(t)) = \frac{1}{2}\underline{x}^T(t_f)\mathbf{S}\underline{x}(t_f) + \frac{1}{2}\int_0^{t_f} \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^{*T}(t)\mathbf{R}\underline{u}^*(t)\, dt \\ \Leftrightarrow J(\underline{u}^*(t)) = \frac{1}{2}\left(\underline{x}^T(t_f)\mathbf{S}\underline{x}(t_f) - \int_0^{t_f} \frac{d}{dt}\left(\underline{x}^T(t)\mathbf{P}(t)\underline{x}(t)\right)\, dt\right) \\ \Leftrightarrow J(\underline{u}^*(t)) = \frac{1}{2}\left(\underline{x}^T(t_f)\mathbf{S}\underline{x}(t_f) - \underline{x}^T(t_f)\mathbf{P}(t_f)\underline{x}(t_f) + \underline{x}^T(0)\mathbf{P}(0)\underline{x}(0)\right) \end{array} \tag{2.70}$$

Then taking into account the boundary conditions $\mathbf{P}(t_f) = \mathbf{S}$ we finally get (2.66).

2.9 Application to minimum energy control problem

Minimum energy control problem appears when $\mathbf{Q} := \mathbf{0}$.

2.9.1 Moving a linear system close to a final state with minimum energy

Let's consider the following dynamical system:

$$\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \tag{2.71}$$

We are looking for the control $\underline{u}(t)$ which moves the system from the initial state $\underline{x}(0) = \underline{x}_0$ to a final state which should be close to a given value $\underline{x}(t_f) = \underline{x}_f$ at final time $t = t_f$. We will assume that the performance index to be minimized is the following quadratic performance index where $\mathbf{R}$ is a symmetric positive definite matrix:

$$J(\underline{u}(t)) = \frac{1}{2}\left(\underline{x}(t_f) - \underline{x}_f\right)^T \mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right) + \frac{1}{2}\int_0^{t_f} \underline{u}^T(t)\mathbf{R}\underline{u}(t)\,dt \tag{2.72}$$

For this optimal control problem, the Hamiltonian (2.5) is:

$$H(\underline{x}, \underline{u}, \underline{\lambda}) = \frac{1}{2}\underline{u}^T(t)\mathbf{R}\underline{u}(t) + \underline{\lambda}^T(t)\left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right) \tag{2.73}$$

The necessary condition for optimality (2.6) yields:

$$\frac{\partial H}{\partial \underline{u}} = \mathbf{R}\underline{u}(t) + \mathbf{B}^T\underline{\lambda}(t) = 0 \tag{2.74}$$

We get:

$$\underline{u}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \tag{2.75}$$

Eliminating $\underline{u}(t)$ in equation (2.72) reads:

$$\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \tag{2.76}$$

The dynamics of Lagrange multipliers $\underline{\lambda}(t)$ is given by (2.9):

$$\dot{\underline{\lambda}}(t) = -\frac{\partial H}{\partial \underline{x}} = -\mathbf{A}^T\underline{\lambda}(t) \tag{2.77}$$

We get from the preceding equation:

$$\underline{\lambda}(t) = e^{-\mathbf{A}^T t}\underline{\lambda}(0) \tag{2.78}$$

The value of $\underline{\lambda}(0)$ will influence the final value of the state vector $\underline{x}(t)$. Indeed let's integrate the linear differential equation:

$$\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T e^{-\mathbf{A}^T t}\underline{\lambda}(0) \tag{2.79}$$

This leads to the following expression of the state vector $\underline{x}(t)$:

$$\begin{array}{ll} \underline{x}(t) & = e^{\mathbf{A}t}\underline{x}_0 + e^{\mathbf{A}t}\int_0^t e^{-\mathbf{A}\tau}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T e^{-\mathbf{A}^T\tau}\underline{\lambda}(0)d\tau \\ & = e^{\mathbf{A}t}\underline{x}_0 + e^{\mathbf{A}t}\left(\int_0^t e^{-\mathbf{A}\tau}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T e^{-\mathbf{A}^T\tau}d\tau\right)\underline{\lambda}(0) \end{array} \tag{2.80}$$

Or:

$$\underline{x}(t) = e^{\mathbf{A}t}\underline{x}_0 + e^{\mathbf{A}t}\,\mathbf{W}_c(t)\,\underline{\lambda}(0) \tag{2.81}$$

where matrix $\mathbf{W}_c(t)$ is defined as follows:

$$\mathbf{W}_c(t) = \int_0^t e^{-\mathbf{A}\tau}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T e^{-\mathbf{A}^T\tau}d\tau \tag{2.82}$$

Now using (2.10) we set $\underline{\lambda}(t_f)$ as follows:

$$\underline{\lambda}(t_f) = \mathbf{S}\left(\underline{x}(t_f) - \underline{x}_f\right) \tag{2.83}$$

Using (2.78) and (2.81) we get:

$$\begin{cases} \underline{\lambda}(t_f) = e^{-\mathbf{A}^T t_f} \underline{\lambda}(0) \\ \underline{x}(t_f) = e^{\mathbf{A} t_f} \underline{x}_0 + e^{\mathbf{A} t_f} \, \mathbf{W}_c(t_f) \, \underline{\lambda}(0) \end{cases} \tag{2.84}$$

And the transversality condition (2.83) is rewritten as follows:

$$\begin{aligned} \underline{\lambda}(t_f) &= \mathbf{S} \left(\underline{x}(t_f) - \underline{x}_f \right) \\ \Leftrightarrow e^{-\mathbf{A}^T t_f} \underline{\lambda}(0) &= \mathbf{S} \left(e^{\mathbf{A} t_f} \underline{x}_0 + e^{\mathbf{A} t_f} \, \mathbf{W}_c(t_f) \, \underline{\lambda}(0) - \underline{x}_f \right) \end{aligned} \tag{2.85}$$

Solving the preceding linear equation in $\underline{\lambda}(0)$ gives the following expression:

$$\begin{aligned} & \left(e^{-\mathbf{A}^T t_f} - \mathbf{S} e^{\mathbf{A} t_f} \mathbf{W}_c(t_f) \right) \underline{\lambda}(0) = \mathbf{S} \left(e^{\mathbf{A} t_f} \underline{x}_0 - \underline{x}_f \right) \\ \Leftrightarrow \; & \underline{\lambda}(0) = \left(e^{-\mathbf{A}^T t_f} - \mathbf{S} e^{\mathbf{A} t_f} \mathbf{W}_c(t_f) \right)^{-1} \mathbf{S} \left(e^{\mathbf{A} t_f} \underline{x}_0 - \underline{x}_f \right) \end{aligned} \tag{2.86}$$

Using the expression of $\underline{\lambda}(0)$ in (2.78) leads to the expression of the Lagrange multiplier $\underline{\lambda}(t)$:

$$\underline{\lambda}(t) = e^{-\mathbf{A}^T t} \left(e^{-\mathbf{A}^T t_f} - \mathbf{S} e^{\mathbf{A} t_f} \mathbf{W}_c(t_f) \right)^{-1} \mathbf{S} \left(e^{\mathbf{A} t_f} \underline{x}_0 - \underline{x}_f \right) \tag{2.87}$$

Finally control $\underline{u}(t)$ is obtained thanks equation (2.75):

$$\underline{u}(t) = -\mathbf{R}^{-1} \mathbf{B}^T \underline{\lambda}(t) \tag{2.88}$$

It is clear from the expression of $\underline{\lambda}(t)$ that the control $\underline{u}(t)$ explicitly depends on the initial state $\underline{x}_0$.

2.9.2 Moving a linear system exactly to a final state with minimum energy

We are now looking for the control $\underline{u}(t)$ which moves the system from the initial state $\underline{x}(0) = \underline{x}_0$ to a given final state $\underline{x}(t_f) = \underline{x}_f$ at final time $t = t_f$. We will assume that the performance index to be minimized is the following quadratic performance index where $\mathbf{R}$ is a symmetric positive definite matrix:

$$J = \frac{1}{2} \int_0^{t_f} \underline{u}^T(t) \mathbf{R} \underline{u}(t) \, dt \tag{2.89}$$

To solve this problem the same reasoning applies than in the previous example. As far as control $\underline{u}(t)$ is concerned this leads to equation (2.75). The change is that now the final value of the state vector $\underline{x}(t)$ is imposed to be $\underline{x}(t_f) = \underline{x}_f$. So there is no final value for the Lagrange multipliers. Indeed $\underline{\lambda}(t_f)$, or equivalently $\underline{\lambda}(0)$, has to be set such that $\underline{x}(t_f) = \underline{x}_f$. We have seen in (2.81) that the state vector $\underline{x}(t)$ has the following expression:

$$\underline{x}(t) = e^{\mathbf{A} t} \underline{x}_0 + e^{\mathbf{A} t} \, \mathbf{W}_c(t) \, \underline{\lambda}(0) \tag{2.90}$$

where matrix $\mathbf{W}_c(t)$ is defined as follows:

$$\mathbf{W}_c(t) = \int_0^t e^{-\mathbf{A} \tau} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T e^{-\mathbf{A}^T \tau} d\tau \tag{2.91}$$

Then we set $\underline{\lambda}(0)$ as follows where $\underline{c}_0$ is a constant vector:

$$\underline{\lambda}(0) = \mathbf{W}_c^{-1}(t_f)\underline{c}_0 \tag{2.92}$$

We get:

$$\underline{x}(t) = e^{\mathbf{A}t}\underline{x}_0 + e^{\mathbf{A}t}\,\mathbf{W}_c(t)\,\mathbf{W}_c^{-1}(t_f)\underline{c}_0 \tag{2.93}$$

Constant vector $\underline{c}_0$ is used to satisfy the final value on the state vector $\underline{x}(t)$. Setting $\underline{x}(t_f) = \underline{x}_f$ leads to the value of constant vector $\underline{c}_0$:

$$\underline{x}(t_f) = \underline{x}_f \Rightarrow \underline{c}_0 = e^{-\mathbf{A}t_f}\underline{x}_f - \underline{x}_0 \tag{2.94}$$

Thus:

$$\underline{\lambda}(0) = \mathbf{W}_c^{-1}(t_f)\left(e^{-\mathbf{A}t_f}\underline{x}_f - \underline{x}_0\right) \tag{2.95}$$

Using (2.95) in (2.78) leads to the expression of the Lagrange multiplier $\underline{\lambda}(t)$:

$$\begin{aligned}\underline{\lambda}(t) &= e^{-\mathbf{A}^T t}\underline{\lambda}(0) \\ &= e^{-\mathbf{A}^T t}\mathbf{W}_c^{-1}(t_f)\left(e^{-\mathbf{A}t_f}\underline{x}_f - \underline{x}_0\right)\end{aligned} \tag{2.96}$$

Finally the control $\underline{u}(t)$ which moves with the minimum energy the system from the initial state $\underline{x}(0) = \underline{x}_0$ to a given final state $\underline{x}(t_f) = \underline{x}_f$ at final time $t = t_f$ has the following expression:

$$\begin{aligned}\underline{u}(t) &= -\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \\ &= -\mathbf{R}^{-1}\mathbf{B}^T e^{-\mathbf{A}^T t}\underline{\lambda}(0) \\ &= -\mathbf{R}^{-1}\mathbf{B}^T e^{-\mathbf{A}^T t}\mathbf{W}_c^{-1}(t_f)\left(e^{-\mathbf{A}t_f}\underline{x}_f - \underline{x}_0\right)\end{aligned} \tag{2.97}$$

It is clear from the preceding expression that the control $\underline{u}(t)$ explicitly depends on the initial state $\underline{x}_0$. When comparing the initial value $\underline{\lambda}(0)$ of the Lagrange multiplier obtained in (2.95) in the case where the final state is imposed to be $\underline{x}(t_f) = \underline{x}_f$ with the expression of the initial value of the Lagrange multiplier obtained in (2.86) in the case where the final state $\underline{x}(t_f)$ is close to a given final state $\underline{x}_f$ we can see that the expression in (2.95) corresponds to the limit of the initial value (2.86) when matrix $\mathbf{S}$ moves towards infinity (note that $\left(e^{\mathbf{A}t_f}\right)^{-1} = e^{-\mathbf{A}t_f}$):

$$\begin{aligned}\lim_{\mathbf{S}\to\infty} &\left(e^{-\mathbf{A}^T t_f} - \mathbf{S}e^{\mathbf{A}t_f}\mathbf{W}_c(t_f)\right)^{-1}\mathbf{S}\left(e^{\mathbf{A}t_f}\underline{x}_0 - \underline{x}_f\right) \\ &= \lim_{\mathbf{S}\to\infty}\left(-\mathbf{S}e^{\mathbf{A}t_f}\mathbf{W}_c(t_f)\right)^{-1}\mathbf{S}\left(e^{\mathbf{A}t_f}\underline{x}_0 - \underline{x}_f\right) \\ &= \lim_{\mathbf{S}\to\infty}\mathbf{W}_c^{-1}(t_f)e^{-\mathbf{A}t_f}\mathbf{S}^{-1}\mathbf{S}\left(e^{\mathbf{A}t_f}\underline{x}_0 - \underline{x}_f\right) \\ &= \mathbf{W}_c^{-1}(t_f)e^{-\mathbf{A}t_f}\left(e^{\mathbf{A}t_f}\underline{x}_0 - \underline{x}_f\right)\end{aligned} \tag{2.98}$$

2.9.3 Example

Given the following scalar plant:

$$\begin{cases}\dot{x}(t) = ax(t) + bu(t) \\ \quad x(0) = x_0\end{cases} \tag{2.99}$$

Find the optimal control for the following cost functional and final states constraints:

We wish to compute a finite horizon Linear Quadratic Regulator with either a fixed or a weighted final state x_f.

– When the final state $x(t_f)$ is set to a fixed value x_f and the cost functional is set to:

$$J = \frac{1}{2}\int_0^{t_f} \rho u^2(t)\, dt \tag{2.100}$$

– When the final state $x(t_f)$ shall be close of a fixed value x_f so that the cost functional is modified as follows where is a positive scalar ($\mathbf{S} > 0$):

$$J = \frac{1}{2}\left(x(t_f) - x_f\right)^T \mathbf{S}\left(x(t_f) - x_f\right) + \frac{1}{2}\int_0^{t_f} \rho u^2(t)\, dt \tag{2.101}$$

In both cases the two-point boundary value problem which shall be solved depends on the solution of the following differential equation where Hamiltonian matrix $\mathbf{H}$ appears:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\lambda}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix} = \begin{bmatrix} a & -b^2/\rho \\ 0 & -a \end{bmatrix} \begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix} = \mathbf{H} \begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix} \tag{2.102}$$

The solution of this differential equation reads:

$$\begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix} = e^{\mathbf{H}t} \begin{bmatrix} x(0) \\ \lambda(0) \end{bmatrix} \tag{2.103}$$

Denoting by s the Laplace variable, the exponential of matrix $\mathbf{H}t$ is obtained thanks to the inverse of the Laplace transform denoted $\mathcal{L}^{-1}$:

$$\begin{array}{rl} e^{\mathbf{H}t} & = \mathcal{L}^{-1}\left(\left(s\mathbb{I} - \mathbf{H}\right)^{-1}\right) \\ & = \mathcal{L}^{-1}\left(\begin{bmatrix} s-a & b^2/\rho \\ 0 & s+a \end{bmatrix}^{-1}\right) \\ & = \mathcal{L}^{-1}\left(\frac{1}{(s-a)(s+a)}\begin{bmatrix} s+a & -b^2/\rho \\ 0 & s-a \end{bmatrix}\right) \\ & = \mathcal{L}^{-1}\left(\begin{bmatrix} \frac{1}{s-a} & \frac{-b^2}{\rho(s^2-a^2)} \\ 0 & \frac{1}{s+a} \end{bmatrix}\right) \\ \Leftrightarrow e^{\mathbf{H}t} & = \begin{bmatrix} e^{at} & \frac{-b^2\left(e^{at}-e^{-at}\right)}{2\rho a} \\ 0 & e^{-at} \end{bmatrix} \end{array} \tag{2.104}$$

That is:

$$\begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix} = e^{\mathbf{H}t} \begin{bmatrix} x(0) \\ \lambda(0) \end{bmatrix} = \begin{bmatrix} e^{at} & \frac{-b^2\left(e^{at}-e^{-at}\right)}{2\rho a} \\ 0 & e^{-at} \end{bmatrix} \begin{bmatrix} x(0) \\ \lambda(0) \end{bmatrix} \tag{2.105}$$

– If the final state $x(t_f)$ is set to the value x_f then the value $\lambda(0)$ is obtained by solving the first equation of (2.105):

$$\begin{array}{l} x(t_f) = x_f = e^{at_f}x(0) - \frac{b^2\left(e^{at_f}-e^{-at_f}\right)}{2\rho a}\lambda(0) \\ \Rightarrow \lambda(0) = \frac{-2\rho a}{b^2\left(e^{at_f}-e^{-at_f}\right)}\left(x_f - e^{at_f}x(0)\right) \end{array} \tag{2.106}$$

And:

$$\left\{ \begin{array}{ll} x(t) & = e^{at}x(0) + \frac{e^{at}-e^{-at}}{e^{at_f}-e^{-at_f}}\left(x_f - e^{at_f}x(0)\right) \\ \lambda(t) & = e^{-at}\lambda(0) = \frac{-2\rho a e^{-at}}{b^2\left(e^{at_f}-e^{-at_f}\right)}\left(x_f - e^{at_f}x(0)\right) \end{array} \right. \tag{2.107}$$

The optimal control $u(t)$ is given by:

$$u(t) = -\mathbf{R}^{-1}\mathbf{B}^T\lambda(t) = \frac{-b}{\rho}\lambda(t) = \frac{2ae^{-at}}{b\left(e^{at_f} - e^{-at_f}\right)}\left(x_f - e^{at_f}x(0)\right) \tag{2.108}$$

Interestingly enough, the open-loop control is independent of the control weighting ρ.

– If the final state $x(t_f)$ is expected to be close to the final value x_f then we have to mix the two equations of (2.105) and the constraint $\lambda(t_f) = \mathbf{S}\left(x(t_f) - x_f\right)$ to compute the value of $\lambda(0)$:

$$\begin{array}{rl} \lambda(t_f) & = \mathbf{S}\left(x(t_f) - x_f\right) \\ \Rightarrow e^{-at_f}\lambda(0) & = \mathbf{S}\left(e^{at_f}x(0) - \frac{b^2\left(e^{at_f}-e^{-at_f}\right)}{2\rho a}\lambda(0) - x_f\right) \\ \Leftrightarrow \lambda(0) & = \frac{\mathbf{S}\left(e^{at_f}x(0)-x_f\right)}{e^{-at_f}+\frac{\mathbf{S}b^2\left(e^{at_f}-e^{-at_f}\right)}{2\rho a}} \end{array} \tag{2.109}$$

Obviously, when $\mathbf{S} \to \infty$ we obtain for $\lambda(0)$ the same expression than (2.106).

2.10 Finite horizon LQ regulator with cross-term in the performance index

Consider the following time invariant state differential equation:

$$\left\{ \begin{array}{l} \underline{\dot{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{x}(0) = \underline{x}_0 \end{array} \right. \tag{2.110}$$

Where:

– $\mathbf{A}$ is the state (or system) matrix

– $\mathbf{B}$ is the input matrix

– $\underline{x}(t)$ is the state vector of dimension n

– $\underline{u}(t)$ is the control vector of dimension m

We will assume that the pair $(\mathbf{A}, \mathbf{B})$ is controllable. The purpose of this section is to explicit the control $\underline{u}(t)$ which minimizes the following quadratic performance index with cross-terms:

$$J(\underline{u}(t)) = \frac{1}{2} \int_0^{t_f} \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t) + 2\underline{x}^T(t)\mathbf{S}\underline{u}(t)\, dt \tag{2.111}$$

With the constraint on terminal state:

$$\underline{x}(t_f) = \underline{0} \tag{2.112}$$

Matrices $\mathbf{S}$ and $\mathbf{Q}$ are symmetric positive semi-definite and matrix $\mathbf{R}$ symmetric positive definite:

$$\begin{cases} \mathbf{S} = \mathbf{S}^T \geq 0 \\ \mathbf{Q} = \mathbf{Q}^T \geq 0 \\ \mathbf{R} = \mathbf{R}^T > 0 \end{cases} \tag{2.113}$$

It can be seen that:

$$\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t) + 2\underline{x}^T(t)\mathbf{S}\underline{u}(t) = \underline{x}^T(t)\mathbf{Q}_m\underline{x}(t) + \underline{v}^T(t)\mathbf{R}\underline{v}(t) \tag{2.114}$$

Where:

$$\begin{cases} \mathbf{Q}_m = \mathbf{Q} - \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T \\ \underline{v}(t) = \underline{u}(t) + \mathbf{R}^{-1}\mathbf{S}^T\underline{x}(t) \end{cases} \tag{2.115}$$

Hence cost (2.111) to be minimized can be rewritten as:

$$J(\underline{u}(t)) = \frac{1}{2} \int_0^{\infty} \underline{x}^T(t)\mathbf{Q}_m\underline{x}(t) + \underline{v}^T(t)\mathbf{R}\underline{v}(t)\, dt \tag{2.116}$$

Furthermore (2.110) is rewritten as follows, where $\underline{v}(t)$ appears as the control vector rather than $\underline{u}(t)$. Using $\underline{u}(t) = \underline{v}(t) - \mathbf{R}^{-1}\mathbf{S}^T\underline{x}(t)$ in (2.110) leads to the following state equation:

$$\begin{array}{ll} \dot{\underline{x}}(t) & = \mathbf{A}\underline{x}(t) + \mathbf{B}\left(\underline{v}(t) - \mathbf{R}^{-1}\mathbf{S}^T\underline{x}(t)\right) \\ & = \left(\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^T\right)\underline{x}(t) + \mathbf{B}\underline{v}(t) \\ & = \mathbf{A}_m\underline{x}(t) + \mathbf{B}\underline{v}(t) \end{array} \tag{2.117}$$

We will assume that symmetric matrix $\mathbf{Q}_m$ is positive semi-definite:

$$\mathbf{Q}_m = \mathbf{Q} - \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T \geq 0 \tag{2.118}$$

Hamiltonian matrix $\mathbf{H}$ reads:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A}_m & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q}_m & -\mathbf{A}_m^T \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^T & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} + \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T & -\mathbf{A}^T + \mathbf{S}\mathbf{R}^{-1}\mathbf{B}^T \end{bmatrix} \tag{2.119}$$

The problem can be solved through the following Hamiltonian system whose state is obtained by extending the state $\underline{x}(t)$ of system (2.110) with costate $\underline{\lambda}(t)$:

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\underline{\lambda}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^T & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} + \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T & -\mathbf{A}^T + \mathbf{S}\mathbf{R}^{-1}\mathbf{B}^T \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} := \mathbf{H} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} \tag{2.120}$$

Ntogramatzidis[2] has shown the following results: let $\mathbf{P}_1$ and $\mathbf{P}_2$ be the *positive semi-definite solutions* of the following continuous time algebraic Riccati equations:

$$\begin{cases} \mathbf{0} = \mathbf{A}^T\mathbf{P}_1 + \mathbf{P}_1\mathbf{A} - (\mathbf{S} + \mathbf{P}_1\mathbf{B})\,\mathbf{R}^{-1}\,(\mathbf{S} + \mathbf{P}_1\mathbf{B})^T + \mathbf{Q} \\ \mathbf{0} = -\mathbf{A}^T\mathbf{P}_2 - \mathbf{P}_2\mathbf{A} - (\mathbf{S} - \mathbf{P}_2\mathbf{B})\,\mathbf{R}^{-1}\,(\mathbf{S} - \mathbf{P}_2\mathbf{B})^T + \mathbf{Q} \end{cases} \tag{2.121}$$

Notice that pair $(\mathbf{A}, \mathbf{B})$ has been replaced by $(-\mathbf{A}, -\mathbf{B})$ in the second equation. We will denote by $\mathbf{K}_1$ and $\mathbf{K}_2$ the following *infinite* horizon gain matrices:

$$\begin{cases} \mathbf{K}_1 = \mathbf{R}^{-1}\left(\mathbf{S}^T + \mathbf{B}^T\mathbf{P}_1\right) \\ \mathbf{K}_2 = \mathbf{R}^{-1}\left(\mathbf{S}^T - \mathbf{B}^T\mathbf{P}_2\right) \end{cases} \tag{2.122}$$

Then the optimal control reads:

$$\underline{u}(t) = \begin{cases} -\mathbf{K}(t)\underline{x}(t) \quad \forall\, 0 \le t < t_f \\ 0 \text{ for } t = t_f \end{cases} \tag{2.123}$$

Where:

$$\begin{cases} \mathbf{K}(t) = \mathbf{R}^{-1}\left(\mathbf{S}^T + \mathbf{B}^T\mathbf{P}(t)\right) \\ \mathbf{P}(t) = \mathbf{X}_2(t)\mathbf{X}_1^{-1}(t) \end{cases} \tag{2.124}$$

And:

$$\begin{cases} \mathbf{X}_1(t) = e^{(\mathbf{A}-\mathbf{B}\mathbf{K}_1)t} - e^{(\mathbf{A}-\mathbf{B}\mathbf{K}_2)\left(t-t_f\right)}e^{(\mathbf{A}-\mathbf{B}\mathbf{K}_1)t_f} \\ \mathbf{X}_2(t) = \mathbf{P}_1 e^{(\mathbf{A}-\mathbf{B}\mathbf{K}_1)t} + \mathbf{P}_2 e^{(\mathbf{A}-\mathbf{B}\mathbf{K}_2)\left(t-t_f\right)}e^{(\mathbf{A}-\mathbf{B}\mathbf{K}_1)t_f} \end{cases} \tag{2.125}$$

Matrix $\mathbf{P}(t)$ satisfy the following Riccati differential equation:

$$-\dot{\mathbf{P}}(t) = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - (\mathbf{S} + \mathbf{B}\mathbf{P}(t))\,\mathbf{R}^{-1}\,(\mathbf{S} + \mathbf{B}\mathbf{P}(t))^T + \mathbf{Q} \tag{2.126}$$

Furthermore the optimal state $\underline{x}(t)$ and costate $\underline{\lambda}(t)$ have the following expressions:

$$\begin{cases} \underline{x}(t) = \mathbf{X}_1(t)\,\mathbf{X}_1^{-1}(0)\,\underline{x}_0 \\ \underline{\lambda}(t) = \mathbf{X}_2(t)\,\mathbf{X}_1^{-1}(0)\,\underline{x}_0 \end{cases} \tag{2.127}$$

2.11 Extension to nonlinear system affine in control

We consider the following finite horizon optimal control problem consisting in finding the control $\underline{u}$ that minimizes the following performance index where $q(\underline{x})$ is positive semi-definite and $\mathbf{R} = \mathbf{R}^T > 0$:

$$J(\underline{u}(t)) = G\left(\underline{x}(t_f)\right) + \frac{1}{2}\int_0^{t_f} \left(q(\underline{x}) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right)\, dt \tag{2.128}$$

under the *constraint* that the system is nonlinear but affine in control:

$$\begin{cases} \underline{\dot{x}}(t) = \underline{f}(\underline{x}) + \underline{g}(\underline{x})\,\underline{u}(t) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{2.129}$$

[2] Lorenzo Ntogramatzidis, A simple solution to the finite-horizon LQ problem with zero terminal state, Kybernetika - 39(4):483-492, January 2003

Assuming no constraint, control $\underline{u}^*(t)$ that *minimizes* the performance index $J(\underline{u}(t))$ is defined by:

$$\underline{u}^*(t) = -\mathbf{R}^{-1}\underline{g}^T(\underline{x})\underline{\lambda}(t) \tag{2.130}$$

where:

$$\begin{aligned} \dot{\underline{\lambda}}(t) &= -\left(\frac{1}{2}\left(\frac{\partial q(\underline{x})}{\partial \underline{x}}\right)^T + \frac{\partial\big(\underline{f}(\underline{x})+\underline{g}(\underline{x})\,\underline{u}^*\big)}{\partial \underline{x}}\underline{\lambda}(t)\right) \\ &= -\left(\frac{1}{2}\left(\frac{\partial q(\underline{x})}{\partial \underline{x}}\right)^T + \frac{\partial\big(\underline{f}(\underline{x})-\underline{g}(\underline{x})\mathbf{R}^{-1}\underline{g}^T(\underline{x})\underline{\lambda}(t)\big)}{\partial \underline{x}}\underline{\lambda}(t)\right) \end{aligned} \tag{2.131}$$

For boundary value problems, efficient minimization of the Hamiltonian is possible[3].

[3] Todorov E. and Tassa Y., Iterative Local Dynamic Programming, IEEE ADPRL, 2009

Chapter 3

Infinite Horizon Linear Quadratic Regulator (LQR)

3.1 Problem to be solved

We recall that we consider the following linear time invariant system, where $\underline{x}(t)$ is the state vector of dimension n, $\underline{u}(t)$ is the control vector of dimension m and $\underline{z}(t)$ is the *controlled* output (that is not the *actual* output of the system but the output of *interest* for the design):

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{z}(t) = \mathbf{N}\,\underline{x}(t) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{3.1}$$

We recall hereafter the performance index which was under consideration in the previous chapter dealing with finite horizon Linear Quadratic Regulator (LQR) when the final state $\underline{x}_f$ is set:

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^{t_f} \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\,dt \tag{3.2}$$

where $\mathbf{Q} = \mathbf{N}^T\mathbf{N} \geq 0$ (thus $\mathbf{Q}$ is symmetric and positive semi-definite) and $\mathbf{R} = \mathbf{R}^T > 0$ is a symmetric and positive definite matrix.

In this chapter we will focus on the case where the final time t_f tends toward infinity ($t_f \to \infty$). The performance index to be minimized turns to be:

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^{\infty} \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right)\,dt \tag{3.3}$$

The results presented in this chapter can be envisioned as the results of the previous chapter as $\|\mathbf{S}\| \to \infty$ ($\underline{x}_f := \underline{0}$ here) and $t_f \to \infty$. When the final time t_f is set to infinity, the Kalman gain $\mathbf{K}(t)$ which has been computed in the previous chapter becomes constant. As a consequence, the control is easier to implement as far as it is no more necessary to integrate the differential Riccati equation and to store the gain $\mathbf{K}(t)$ before applying the control. In practice *infinity* means that final time t_f becomes large when compared to the time constants of the plant.

3.2 Stabilizability and detectability

We will assume in the following that $(\mathbf{A}, \mathbf{B})$ is stabilizable and $(\mathbf{A}, \mathbf{N})$ is detectable. We recall that the pair $(\mathbf{A}, \mathbf{B})$ is said stabilizable if the uncontrollable eigenvalues of $\mathbf{A}$, if any, have negative real parts. Thus even though not all system modes are controllable, the ones that are not controllable do not require stabilization.

Similarly the pair $(\mathbf{A}, \mathbf{N})$ is said detectable if the unobservable eigenvalues of $\mathbf{A}$, if any, have negative real parts. Thus even though not all system modes are observable, the ones that are not observable do not require stabilization. We may use the Kalman test to check the controllability of the system:

$$\text{rank}\begin{bmatrix}\mathbf{B} & \mathbf{AB} & \cdots & \mathbf{A}^{n-1}\mathbf{B}\end{bmatrix} = n \text{ where } n = \text{ size of state vector } \underline{x} \tag{3.4}$$

Or equivalently the Popov-Belevitch-Hautus (PBH) test which shall be applied to all eigenvalues of $\mathbf{A}$, denoted λ_i, to check the controllability of the system, or only on the eigenvalues which are not contained in the left half plane to check the stabilizability of the system:

$$\text{rank}\begin{bmatrix}\mathbf{A} - \lambda_i \mathbb{I} & \mathbf{B}\end{bmatrix} = n \begin{cases} \forall\ \lambda_i \text{ for controllability} \\ \forall\ \lambda_i \text{ s.t. } Re(\lambda_i) \geq 0 \text{ for stabilizability} \end{cases} \tag{3.5}$$

Similarly we may use the Kalman test to check the observability of the system:

$$\text{rank}\begin{bmatrix}\mathbf{N} \\ \mathbf{NA} \\ \vdots \\ \mathbf{NA}^{n-1}\end{bmatrix} = n \text{ where } n = \text{ size of state vector } \underline{x} \tag{3.6}$$

Or equivalently the Popov-Belevitch-Hautus (PBH) test which shall be applied to all eigenvalues of $\mathbf{A}$, denoted λ_i, to check the observability of the system, or only on the eigenvalues which are not contained in the left half plane to check the detectability of the system:

$$\text{rank}\begin{bmatrix}\mathbf{A} - \lambda_i \mathbb{I} \\ \mathbf{N}\end{bmatrix} = n \begin{cases} \forall\ \lambda_i \text{ for observability} \\ \forall\ \lambda_i \text{ s.t. } Re(\lambda_i) \geq 0 \text{ for detectability} \end{cases} \tag{3.7}$$

3.3 Algebraic Riccati equation

When final time t_f tends toward infinity the matrix $\mathbf{P}(t)$ turns to be a constant symmetric positive definite matrix denoted $\mathbf{P}$. The *Riccati* equation (2.44) reduces to an algebraic equation, which is known as the *algebraic Riccati equation* (ARE):

$$\mathbf{A}^T\mathbf{P} + \mathbf{PA} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \tag{3.8}$$

It is worth noticing that the *algebraic Riccati* equation (3.8) may have several solutions. The solution of the optimal control problem only retains the *positive semi-definite solution of the algebraic Riccati* equation.

The convergence of $\lim_{t_f \to \infty} \mathbf{P}(t) \to \mathbf{P}$ where $\mathbf{P} \geq 0$ is some positive semi-definite symmetric constant matrix is guaranteed by the stabilizability assumption ($\mathbf{P}^T$ is indeed a solution of the algebraic Riccati equation (3.8)). Since the matrix $\mathbf{P} = \mathbf{P}^T \geq 0$ is constant, the optimal gain $\mathbf{K}(t)$ also turns to be also a constant denoted $\mathbf{K}$. The optimal gain $\mathbf{K}$ and the optimal stabilizing control $\underline{u}(t)$ are then defined as follows:

$$\left\{ \begin{array}{l} \underline{u}(t) = -\mathbf{K}\underline{x}(t) \\ \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \end{array} \right. \tag{3.9}$$

The need for the detectability assumption is to ensure that the optimal control computed using the $\lim_{t_f \to \infty} \mathbf{P}(t)$ generates a feedback gain $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$ that stabilizes the plant, i.e. all the eigenvalues of $\mathbf{A} - \mathbf{B}\mathbf{K}$ lie on the open left half plane. In addition, it can be shown that the minimum cost achieved is given by:

$$J^* = \frac{1}{2}\underline{x}^T(0)\mathbf{P}\underline{x}(0) \tag{3.10}$$

To get this result first we notice that the Hamiltonian (1.63) reads:

$$\mathcal{H}(\underline{x}, \underline{u}, \underline{\lambda}) = \frac{1}{2}\left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right) + \underline{\lambda}^T(t)\left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right) \tag{3.11}$$

The necessary condition for optimality (1.72) yields:

$$\frac{\partial \mathcal{H}}{\partial \underline{u}} = \mathbf{R}\underline{u}(t) + \mathbf{B}^T\underline{\lambda}(t) = 0 \tag{3.12}$$

Taking into account that $\mathbf{R}$ is a symmetric matrix, we get:

$$\underline{u}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \tag{3.13}$$

Eliminating $\underline{u}(t)$ in equation (3.1) reads:

$$\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \tag{3.14}$$

The dynamics of Lagrange multipliers $\underline{\lambda}(t)$ is given by (1.69):

$$\dot{\underline{\lambda}}(t) = -\frac{\partial \mathcal{H}}{\partial \underline{x}} = -\mathbf{Q}\underline{x}(t) - \mathbf{A}^T\underline{\lambda}(t) \tag{3.15}$$

The key point in the LQR design is that Lagrange multipliers $\underline{\lambda}(t)$ are now assume to linearly depends on state vector $\underline{x}(t)$ through a constant symmetric positive definite matrix denoted $\mathbf{P}$:

$$\underline{\lambda}(t) = \mathbf{P}\underline{x}(t) \text{ where } \mathbf{P} = \mathbf{P}^T \geq 0 \tag{3.16}$$

By taking the time derivative of the Lagrange multipliers $\underline{\lambda}(t)$ and using again equation (3.1) we get:

$$\dot{\underline{\lambda}}(t) = \mathbf{P}\dot{\underline{x}}(t) = \mathbf{P}\left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right) = \mathbf{P}\mathbf{A}\underline{x}(t) + \mathbf{P}\mathbf{B}\underline{u}(t) \tag{3.17}$$

Then using the expression of control $\underline{u}(t)$ provided in (3.13) as well as (3.16) we get:

$$\begin{aligned} \dot{\underline{\lambda}}(t) &= \mathbf{PA}\underline{x}(t) - \mathbf{PBR}^{-1}\mathbf{B}^T\underline{\lambda}(t) \\ &= \mathbf{PA}\underline{x}(t) - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P}\underline{x}(t) \end{aligned} \tag{3.18}$$

Finally using (3.18) within (3.15) and using $\underline{\lambda}(t) = \mathbf{P}\underline{x}(t)$ (see (3.16)) we get:

$$\begin{aligned} -\mathbf{PA}\underline{x}(t) + \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P}\underline{x}(t) = \mathbf{Q}\underline{x}(t) + \mathbf{A}^T\mathbf{P}\underline{x}(t) \\ \Leftrightarrow \left(\mathbf{A}^T\mathbf{P} + \mathbf{PA} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q}\right)\underline{x}(t) = \underline{0} \end{aligned} \tag{3.19}$$

As far as this equality stands for every value of the state vector $\underline{x}(t)$ we retrieve the algebraic Riccati equation (3.8):

$$\mathbf{A}^T\mathbf{P} + \mathbf{PA} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \tag{3.20}$$

3.4 Extension to nonlinear system affine in control

We consider the following infinite horizon optimal control problem consisting in finding the control $\underline{u}$ that minimizes the following performance index where $q(\underline{x})$ is positive semi-definite:

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^{\infty} \left(q(\underline{x}) + \underline{u}^T(t)\underline{u}(t)\right)\, dt \tag{3.21}$$

under the *constraint* that the system is nonlinear but affine in control:

$$\begin{cases} \dot{\underline{x}} = \underline{f}(\underline{x}) + \underline{g}(\underline{x})\,\underline{u}(t) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{3.22}$$

We assume that vector field $\underline{f}$ is such that $\underline{f}(\underline{x}) = \underline{0}$. Thus $(\underline{x}_e := \underline{0}, \underline{u}_e := \underline{0})$ is an equilibrium point for the nonlinear system affine in control. Consequently $\underline{f}(\underline{x}) = \mathbf{F}(\underline{x})\,\underline{x}$ for some, possibly not unique, continuous function $F : \mathbb{R}^n \to \mathbb{R}^{n\times n}$. The classical optimal control design methodology relies on the solution of the Hamilton-Jacobi-Bellman (HJB) equation (1.112):

$$0 = min_{\underline{u}(t)\in\mathcal{U}} \left(\frac{1}{2}\left(q(\underline{x}) + \underline{u}^T\underline{u}\right) + \left(\frac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)^T \left(\underline{f}(\underline{x}) + \underline{g}(\underline{x})\,\underline{u}\right)\right) \tag{3.23}$$

Assuming no constraint, the minimum of the preceding Hamilton-Jacobi-Bellman (HJB) equation with respect to $\underline{u}$ is attained for optimal control $\underline{u}^*(t)$ defined by:

$$\underline{u}^*(t) = -\underline{g}^T(\underline{x})\left(\frac{\partial J^*(\underline{x})}{\partial \underline{x}}\right) \tag{3.24}$$

Then replacing $\underline{u}$ by $\underline{u}^* = -\underline{g}^T(\underline{x})\left(\frac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)$, the Hamilton-Jacobi-Bellman (HJB) equation reads:

$$\begin{aligned} 0 &= \tfrac{1}{2}\left(q(\underline{x}) + \left(\tfrac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)^T \underline{g}(\underline{x})\underline{g}^T(\underline{x})\left(\tfrac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)\right) \\ &\quad + \left(\tfrac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)^T \left(\underline{f}(\underline{x}) - \underline{g}(\underline{x})\underline{g}^T(\underline{x})\left(\tfrac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)\right) \end{aligned} \tag{3.25}$$

We finally get:

$$\frac{1}{2}q(\underline{x}) + \left(\frac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)^T \underline{f}(\underline{x}) - \frac{1}{2}\left(\frac{\partial J^*(\underline{x})}{\partial \underline{x}}\right)^T \underline{g}(\underline{x})\underline{g}^T(\underline{x})\left(\frac{\partial J^*(\underline{x})}{\partial \underline{x}}\right) = 0 \quad (3.26)$$

In the linearized case the solution of the optimal control problem is a linear static state feedback of the form $\underline{u} = -\mathbf{B}^T\bar{\mathbf{P}}$, where $\bar{\mathbf{P}}$ is the symmetric positive definite solution of the algebraic Riccati equation:

$$\mathbf{A}^T\bar{\mathbf{P}} + \bar{\mathbf{P}}\mathbf{A} - \bar{\mathbf{P}}\mathbf{B}\mathbf{B}^T\bar{\mathbf{P}} + \mathbf{Q} = \mathbf{0} \quad (3.27)$$

where:

$$\begin{cases} \mathbf{A} = \left.\frac{\partial \underline{f}(\underline{x})}{\partial \underline{x}}\right|_{\underline{x}=\underline{0}} \\ \mathbf{B} = \underline{g}(\underline{0}) \\ \mathbf{Q} = \frac{1}{2}\left.\frac{\partial^2 q(\underline{x})}{\partial \underline{x}^2}\right|_{\underline{x}=\underline{0}} \end{cases} \quad (3.28)$$

Following Sassano and Astolfi[1], there exists a matrix $\mathbf{R} = \mathbf{R}^T > 0$, a neighbourhood of the origin $\Omega \subseteq \mathbb{R}^{2n}$ and $\bar{k} \geq 0$ such that for all $k \geq \bar{k}$ the function $V(\underline{x}, \underline{\xi})$ is positive definite and satisfies the following partial differential inequality:

$$\frac{1}{2}q(\underline{x}) + V_{\underline{x}}(\underline{x}, \underline{\xi})\underline{f}(\underline{x}) + V_{\underline{\xi}}(\underline{x}, \underline{\xi})\,\dot{\underline{\xi}} - \frac{1}{2}V_{\underline{x}}(\underline{x}, \underline{\xi})\,\underline{g}(\underline{x})\underline{g}^T(\underline{x})V_{\underline{x}}^T(\underline{x}, \underline{\xi}) \leq 0 \quad (3.29)$$

where:

$$\begin{cases} V(\underline{x}, \underline{\xi}) = P(\underline{\xi})\underline{x} + \frac{1}{2}(\underline{x} - \underline{\xi})^T\mathbf{R}(\underline{x} - \underline{\xi}) \\ \dot{\underline{\xi}} = -k\,V_{\underline{\xi}}^T(\underline{x}, \underline{\xi}) \quad \forall\,(\underline{x}, \underline{\xi}) \in \Omega \end{cases} \quad (3.30)$$

The $\mathcal{C}^1$ mapping $P : \mathbb{R}^n \to \mathbb{R}^{1\times n}$, $P(\underline{0}) = \underline{0}^T$, is defined as follows:

$$\frac{1}{2}q(\underline{x}) + P(\underline{x})\underline{f}(\underline{x}) - \frac{1}{2}P(\underline{x})\underline{g}(\underline{x})\underline{g}^T(\underline{x})P(\underline{x})^T + \sigma(\underline{x}) = 0 \quad (3.31)$$

where $\sigma(\underline{x}) = \underline{x}^T\Sigma(\underline{x})\underline{x}$ with $\Sigma : \mathbb{R}^n \to \mathbb{R}^{n\times n}$, $\Sigma(\underline{0}) = \underline{0}$.
Furthermore $P(\underline{x})$ is tangent at $\underline{x} = \underline{0}$ to $\bar{\mathbf{P}}$:

$$\left.\frac{\partial P(\underline{x})^T}{\partial \underline{x}}\right|_{\underline{x}=\underline{0}} = \bar{\mathbf{P}} \quad (3.32)$$

Since $P(\underline{x})$ is tangent at $\underline{x} = \underline{0}$ to the solution $\bar{\mathbf{P}}$ of the algebraic Riccati equation, the function $P(\underline{x})\,\underline{x} : \mathbb{R}^n \to \mathbb{R}$ is locally quadratic around the origin and moreover has a local minimum for $\underline{x} = \underline{0}$.

Let $\mathbf{\Psi}(\underline{\xi})$ be Jacobian matrix of the mapping $P(\underline{\xi})$ and $\mathbf{\Phi} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n\times n}$ a continuous matrix valued function such that:

$$\begin{cases} P(\underline{\xi}) = \underline{\xi}^T\mathbf{\Psi}(\underline{\xi})^T \\ P(\underline{x}) - P(\underline{\xi}) = (\underline{x} - \underline{\xi})^T\mathbf{\Phi}(\underline{x}, \underline{\xi})^T \end{cases} \quad (3.33)$$

[1]Sassano M. and Astolfi A, Dynamic approximate solutions of the HJ inequality and of the HJB equation for input-affine nonlinear systems. IEEE Transactions on Automatic Control, 57(10):2490–2503, 2012.

Then the approximate regional dynamic optimal control is found to be[1]:

$$\begin{array}{ll} \underline{u} & = -\underline{g}(\underline{x})^T V_{\underline{x}}^T(\underline{x},\underline{\xi}) \\ & = -\underline{g}(\underline{x})^T \left(P(\underline{\xi})^T + \mathbf{R}(\underline{x}-\underline{\xi})\right) \\ & = -\underline{g}(\underline{x})^T \left(P(\underline{x})^T + \mathbf{R}(\underline{x}-\underline{\xi}) - \left(P(\underline{x})^T - P(\underline{\xi})^T\right)\right) \\ & = -\underline{g}(\underline{x})^T \left(P(\underline{x})^T + \left(\mathbf{R} - \mathbf{\Phi}(\underline{x},\underline{\xi})\right)(\underline{x}-\underline{\xi})\right) \end{array} \tag{3.34}$$

where:

$$\dot{\underline{\xi}} = -k\, V_{\underline{\xi}}^T(\underline{x},\underline{\xi}) = -k\, \left(\mathbf{\Psi}(\underline{\xi})^T \underline{x} - \mathbf{R}\left(\underline{x}-\underline{\xi}\right)\right) \tag{3.35}$$

Such control has been applied to internal combustion engine test benches[2].

3.5 Solving the algebraic Riccati equation

3.5.1 Hamiltonian matrix based solution

It can be shown that if the pair $(\mathbf{A},\mathbf{B})$ is stabilizable and the pair $(\mathbf{A},\mathbf{N})$ is detectable, with $\mathbf{Q} = \mathbf{N}^T\mathbf{N}$ positive semi-definite and $\mathbf{R}$ positive definite, then $\mathbf{P}$ is a the unique positive semi-definite (symmetric) solution of the *algebraic Riccati equation* (ARE) (3.8).

Combining (3.14) and (3.15) into a single state equation yields:

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\underline{\lambda}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} := \mathbf{H} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} \tag{3.36}$$

We have seen that the following $2n \times 2n$ matrix $\mathbf{H}$ is called the Hamiltonian matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \tag{3.37}$$

It can be shown that the Hamiltonian matrix $\mathbf{H}$ has n eigenvalues in the open left half plane and n eigenvalues in the open right half plane. The eigenvalues are symmetric with respects to the imaginary axis: if $\underline{\lambda}$ is and eigenvalue of $\mathbf{H}$ then $-\underline{\lambda}$ is also an eigenvalue of $\mathbf{H}$. In addition $\mathbf{H}$ has no pure imaginary eigenvalues.

Furthermore if the $2n \times n$ matrix $\begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix}$ has columns that comprise all the eigenvectors of $\mathbf{H}$ corresponding to the n eigenvalues in the *open left half plane*. Then $\mathbf{X}_1$ is invertible and the positive semi-definite solution of the *algebraic Riccati equation (ARE)* is:

$$\mathbf{P} = \mathbf{X}_2\mathbf{X}_1^{-1} \tag{3.38}$$

Similarly the negative semi-definite solution of the *algebraic Riccati equation (ARE)* is build thanks to the eigenvectors associated with the n eigenvalues in the open right half plane (i.e. the unstable invariant subspace). Once again the solution of the optimal control problem only retains the positive semi-definite solution of the *algebraic Riccati* equation.

[2] Passenbrunner T., Sassano M., del Re L., Optimal Control with Input Constraints applied to Internal Combustion Engine Test Benches, 9th IFAC Symposium on Nonlinear Control Systems, September 4-6, 2013. Toulouse, France

In addition it can be shown that the eigenvalues of $\mathbf{A} - \mathbf{BK}$ where $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$ (that are the eigenvalues of the closed-loop plant) are equal to the n eigenvalues in the open left half plane of the Hamiltonian matrix $\mathbf{H}$.

3.5.2 Proof of the results on the Hamiltonian matrix

We recall that, by definition, a matrix $\mathbf{H}$ is said to be an Hamiltonian matrix as soon as the following property holds:

$$(\mathbf{JH})^T = \mathbf{JH} \Leftrightarrow (\mathbf{HJ})^T = \mathbf{HJ} \text{ where } \mathbf{J} = \begin{bmatrix} \mathbf{0} & \mathbb{I} \\ -\mathbb{I} & \mathbf{0} \end{bmatrix} \tag{3.39}$$

Matrix $\mathbf{J}$ as the following properties:

$$\mathbf{J}^T\mathbf{J} = \mathbf{J}\mathbf{J}^T = \begin{bmatrix} \mathbb{I} & 0 \\ 0 & \mathbb{I} \end{bmatrix} \text{ and } \mathbf{JJ} = \mathbf{J}^T\mathbf{J}^T = -\begin{bmatrix} \mathbb{I} & 0 \\ 0 & \mathbb{I} \end{bmatrix} \tag{3.40}$$

In addition the following relation holds:

$$\mathbf{HJ} = (\mathbf{HJ})^T \Rightarrow \mathbf{J}^T\mathbf{HJ} = \mathbf{J}^T\mathbf{J}^T\mathbf{H}^T = -\mathbf{H}^T \tag{3.41}$$

Let λ be an eigenvalue of Hamiltonian matrix $\mathbf{H}$ associated with eigenvector $\underline{x}$. We get:

$$\begin{array}{l} \mathbf{H}\underline{x} = \lambda\,\underline{x} \\ \Rightarrow \mathbf{HJJ}^T\underline{x} = \lambda\,\underline{x} \\ \Rightarrow \mathbf{J}^T\mathbf{HJJ}^T\underline{x} = \lambda\,\mathbf{J}^T\underline{x} \\ \Leftrightarrow -\mathbf{H}^T\mathbf{J}^T\underline{x} = \lambda\,\mathbf{J}^T\underline{x} \\ \Leftrightarrow \mathbf{H}^T\mathbf{J}^T\underline{x} = -\lambda\,\mathbf{J}^T\underline{x} \end{array} \tag{3.42}$$

Thus $-\lambda$ is an eigenvalue of $\mathbf{H}^T$ with the corresponding eigenvector $\mathbf{J}^T\underline{x}$. Using the fact that $\det(\mathbf{M}) = \det\left(\mathbf{M}^T\right)$ we get:

$$\det\left(-\lambda\mathbb{I} - \mathbf{H}^T\right) = \det\left((-\lambda\mathbb{I} - \mathbf{H})^T\right) \tag{3.43}$$

As a consequence we conclude that $-\lambda$ is also an eigenvalue of $\mathbf{H}$.
To show that $\mathbf{H}$ has no eigenvalue on the imaginary axis suppose:

$$\mathbf{H}\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{BR}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix}\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} = \lambda\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} \tag{3.44}$$

Where $\underline{x}_1$ and $\underline{x}_2$ are *not* both zero and

$$\lambda + \lambda^* = 0 \tag{3.45}$$

where λ^* stands for the complex conjugate of λ. We seek a contradiction. Let's denote by $\underline{x}^*$ the transpose conjugate of vector $\underline{x}$.

– Equation (3.44) gives:

$$\mathbf{A}\underline{x}_1 - \mathbf{BR}^{-1}\mathbf{B}^T\underline{x}_2 = \lambda\underline{x}_1 \Rightarrow \underline{x}_2^*\mathbf{BR}^{-1}\mathbf{B}^T\underline{x}_2 = \underline{x}_2^*\mathbf{A}\underline{x}_1 - \lambda\underline{x}_2^*\underline{x}_1 \tag{3.46}$$

– Taking into account that $\mathbf{Q}$ is a real symmetric matrix, equation (3.44) also gives:

$$-\mathbf{Q}\underline{x}_1 - \mathbf{A}^T\underline{x}_2 = \lambda\underline{x}_2 \Rightarrow \lambda\underline{x}_2^T = -\underline{x}_1^T\mathbf{Q} - \underline{x}_2^T\mathbf{A} \Rightarrow \lambda^*\underline{x}_2^* = -\underline{x}_1^*\mathbf{Q} - \underline{x}_2^*\mathbf{A} \tag{3.47}$$

Denoting $\mathbf{M} = \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T$ and taking into account (3.47) into (3.46) yields:

$$\begin{array}{l} \left\{ \begin{array}{l} \underline{x}_2^*\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\underline{x}_2 = \underline{x}_2^*\mathbf{A}\underline{x}_1 - \lambda\underline{x}_2^*\underline{x}_1 \\ \underline{x}_2^*\mathbf{A} = -\underline{x}_1^*\mathbf{Q} - \lambda^*\underline{x}_2^* \end{array} \right. \\ \Rightarrow \underline{x}_2^*\mathbf{M}\underline{x}_2 = -\underline{x}_1^*\mathbf{Q}\underline{x}_1 - \lambda^*\underline{x}_2^*\underline{x}_1 - \lambda\underline{x}_2^*\underline{x}_1 = -\underline{x}_1^*\mathbf{Q}\underline{x}_1 - (\lambda^* + \lambda)\,\underline{x}_2^*\underline{x}_1 \end{array} \tag{3.48}$$

Using (3.45) we finally get:

$$\underline{x}_2^*\mathbf{M}\underline{x}_2 = -\underline{x}_1^*\,\mathbf{Q}\,\underline{x}_1 \tag{3.49}$$

Since $\mathbf{R}$ and $\mathbf{Q}$ are positive semi-definite matrices, and consequently also $\mathbf{M} = \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T$, this implies:

$$\left\{ \begin{array}{l} \mathbf{M}\underline{x}_2 = \underline{0} \\ \mathbf{Q}\underline{x}_1 = \underline{0} \end{array} \right. \tag{3.50}$$

Then using (3.46) we get:

$$\left\{ \begin{array}{l} \mathbf{A}\underline{x}_1 = \lambda\,\underline{x}_1 \\ \mathbf{Q}\underline{x}_1 = \underline{0} \end{array} \right. \Rightarrow \begin{bmatrix} \mathbf{A} - \lambda\mathbb{I} \\ \mathbf{Q} \end{bmatrix} \underline{x}_1 = \underline{0} \tag{3.51}$$

If $\underline{x}_1 \neq \underline{0}$ then this contradicts observability of the pair $(\mathbf{Q}, \mathbf{A})$ by the Popov-Belevitch-Hautus test. Similarly if $\underline{x}_2 \neq \underline{0}$ then $\underline{x}_2^*\begin{bmatrix}\mathbf{M} & \mathbf{A} + \lambda^*\mathbb{I}\end{bmatrix} = \underline{0}$ which contradicts the observability of the pair $(\mathbf{A}, \mathbf{M})$.

3.5.3 Solving general algebraic Riccati and Lyapunov equations

The general algebraic Riccati equation reads as follows where all matrices are square of dimension $n \times n$:

$$\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{B} + \mathbf{C} + \mathbf{X}\mathbf{D}\mathbf{X} = \mathbf{0} \tag{3.52}$$

Matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are known whereas matrix $\mathbf{X}$ has to be determined.

The general algebraic Lyapunov equation is obtained as a special case of the algebraic Riccati by setting $\mathbf{D} = \mathbf{0}$.

The general algebraic Riccati equation can be solved[3] by considering the following $2n \times 2n$ matrix $\mathbf{H}$:

$$\mathbf{H} = \begin{bmatrix} \mathbf{B} & \mathbf{D} \\ -\mathbf{C} & -\mathbf{A} \end{bmatrix} \tag{3.53}$$

Let the eigenvalues of matrix $\mathbf{H}$ be denoted λ_1, $i = 1, \cdots, 2n$, and the corresponding eigenvectors be denoted $\underline{v}_i$. Furthermore let $\mathbf{M}$ be the $2n \times 2n$ matrix composed of all real eigenvectors of matrix $\mathbf{H}$; for complex conjugate eigenvectors, the corresponding columns of matrix $\mathbf{M}$ are changed into the real

[3] Optimal Control of Singularly Perturbed Linear Systems with Applications: High Accuracy Techniques, Z. Gajic and M. Lim, Marcel Dekker, New York, 2001

and imaginary parts of such eigenvectors. Note that there are many ways to form matrix $\mathbf{M}$.

Then we can write the following relation:

$$\mathbf{H}\mathbf{M} = \mathbf{M}\mathbf{\Lambda} = \begin{bmatrix} \mathbf{M}_1 & \mathbf{M}_2 \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_2 \end{bmatrix} \tag{3.54}$$

Matrix $\mathbf{M}_1$ contains the n first columns of $\mathbf{M}$ whereas matrix $\mathbf{M}_2$ contains the n last columns of $\mathbf{M}$.

Matrices $\mathbf{\Lambda}_1$ and $\mathbf{\Lambda}_2$ are diagonal matrices formed by the eigenvalues of $\mathbf{H}$ as soon as there are distinct; for eigenvalues with multiplicity greater than 1, the corresponding part in matrix $\mathbf{\Lambda}$ represents the Jordan form.

Thus we have:

$$\begin{cases} \mathbf{H}\mathbf{M}_1 = \mathbf{M}_1\mathbf{\Lambda}_1 \\ \mathbf{H}\mathbf{M}_2 = \mathbf{M}_2\mathbf{\Lambda}_2 \end{cases} \tag{3.55}$$

We will focus our attention on the first equation and split matrix $\mathbf{M}_1$ as follows:

$$\mathbf{M}_1 = \begin{bmatrix} \mathbf{M}_{11} \\ \mathbf{M}_{12} \end{bmatrix} \tag{3.56}$$

Using the expression of $\mathbf{H}$ in (3.53), the relation $\mathbf{H}\mathbf{M}_1 = \mathbf{M}_1\mathbf{\Lambda}_1$ reads as follows:

$$\mathbf{H}\mathbf{M}_1 = \mathbf{M}_1\mathbf{\Lambda}_1 \Rightarrow \begin{cases} \mathbf{B}\mathbf{M}_{11} + \mathbf{D}\mathbf{M}_{12} = \mathbf{M}_{11}\mathbf{\Lambda}_1 \\ -\mathbf{C}\mathbf{M}_{11} - \mathbf{A}\mathbf{M}_{12} = \mathbf{M}_{12}\mathbf{\Lambda}_1 \end{cases} \tag{3.57}$$

Assuming that matrix $\mathbf{M}_{11}$ is not singular, we can check that a solution $\mathbf{X}$ of the general algebraic Riccati equation (3.52) reads:

$$\mathbf{X} = \mathbf{M}_{12}\mathbf{M}_{11}^{-1} \tag{3.58}$$

Indeed:

$$\begin{array}{ll} \begin{cases} \mathbf{B}\mathbf{M}_{11} + \mathbf{D}\mathbf{M}_{12} = \mathbf{M}_{11}\mathbf{\Lambda}_1 \\ \mathbf{C}\mathbf{M}_{11} + \mathbf{A}\mathbf{M}_{12} = -\mathbf{M}_{12}\mathbf{\Lambda}_1 \\ \mathbf{X} = \mathbf{M}_{12}\mathbf{M}_{11}^{-1} \end{cases} & \\ \Rightarrow \mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{B} + \mathbf{C} + \mathbf{X}\mathbf{D}\mathbf{X} & = \mathbf{A}\mathbf{M}_{12}\mathbf{M}_{11}^{-1} + \mathbf{M}_{12}\mathbf{M}_{11}^{-1}\mathbf{B} + \mathbf{C} \\ & \qquad + \mathbf{M}_{12}\mathbf{M}_{11}^{-1}\mathbf{D}\mathbf{M}_{12}\mathbf{M}_{11}^{-1} \\ & = \left(\mathbf{A}\mathbf{M}_{12} + \mathbf{C}\mathbf{M}_{11}\right)\mathbf{M}_{11}^{-1} \\ & \qquad + \mathbf{M}_{12}\mathbf{M}_{11}^{-1}\left(\mathbf{B}\mathbf{M}_{11} + \mathbf{D}\mathbf{M}_{12}\right)\mathbf{M}_{11}^{-1} \\ & = -\mathbf{M}_{12}\mathbf{\Lambda}_1\mathbf{M}_{11}^{-1} + \mathbf{M}_{12}\mathbf{M}_{11}^{-1}\mathbf{M}_{11}\mathbf{\Lambda}_1\mathbf{M}_{11}^{-1} \\ & = \mathbf{0} \end{array} \tag{3.59}$$

It is worth noticing that each selection of eigenvectors within matrix $\mathbf{M}_1$ leads to a new solution of the general algebraic Riccati equation (3.52). Consequently the solution to the general algebraic Riccati equation (3.52) is not unique. The same statement holds for different choice of matrix $\mathbf{M}_2$ and the corresponding solution of (3.52) is obtained from $\mathbf{X} = \mathbf{M}_{21}\mathbf{M}_{22}^{-1}$.

3.6 Application to the optimal control of any scalar LTI plant

We consider the following scalar linear time invariant plant where $x(t) \in \mathbb{R}$, $u(t) \in \mathbb{R}$:

$$\left\{ \begin{array}{l} \dot{x}(t) = a\, x(t) + b\, u(t) \quad \text{where } b \neq 0 \\ z(t) = c_1\, x(t) \quad \text{where } c_1 \neq 0 \end{array} \right. \tag{3.60}$$

We wish to minimize the following performance index:

$$J(u(t)) = \frac{1}{2} \int_0^\infty z^2(t) + \rho\, u^2(t)\, dt \text{ where } \rho > 0 \tag{3.61}$$

It is easy to check that pair (a, b) is controllable (meaning that $b \neq 0$) and that pair (a, c_1) is observable (meaning that $c_1 \neq 0$).

In order to match the considered performance index with the general expression $\frac{1}{2} \int_0^\infty \left(\underline{x}^T(t) \mathbf{Q} \underline{x}(t) + \underline{u}^T(t) \mathbf{R} \underline{u}(t) \right) dt$ of the performance index, we define weights $\mathbf{Q}$ and $\mathbf{R}$ as follows:

$$\begin{array}{l} z^2(t) = z^T(t) z(t) = (c_1\, x(t))^T\, (c_1\, x(t)) = x^T(t) c_1^T c_1 x(t) \\ \Rightarrow \left\{ \begin{array}{l} \mathbf{Q} := c_1^T c_1 = c_1^2 \\ \mathbf{R} := \rho \end{array} \right. \end{array} \tag{3.62}$$

The Hamiltonian matrix $\mathbf{H}$ reads:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{\lambda}(t) \end{bmatrix} = \begin{bmatrix} a & -\frac{b^2}{\rho} \\ -c_1^2 & -a \end{bmatrix} \tag{3.63}$$

The eigenvalues of $\mathbf{H}$ are obtained by solving:

$$\begin{array}{l} \det\left(s\mathbb{I} - \mathbf{H}\right) = 0 \Rightarrow \det\left(\begin{bmatrix} s - a & \frac{b^2}{\rho} \\ c_1^2 & s + a \end{bmatrix} \right) = 0 \\ \Leftrightarrow (s-a)(s+a) - \frac{(c_1 b)^2}{\rho} = 0 \\ \Leftrightarrow s^2 - a^2 - \frac{(c_1 b)^2}{\rho} = 0 \end{array} \tag{3.64}$$

Thus the two eigenvalues of $\mathbf{H}$ read:

$$\left\{ \begin{array}{l} \lambda_1 = +\sqrt{a^2 + \frac{(c_1 b)^2}{\rho}} \\ \lambda_2 = -\sqrt{a^2 + \frac{(c_1 b)^2}{\rho}} \end{array} \right. \tag{3.65}$$

We check that the eigenvalues of $\mathbf{H}$ are symmetric with respect to the imaginary axis.

The eigenvectors $\underline{v}_1$ and $\underline{v}_2$ corresponding to eigenvalues λ_1 and λ_2, respectively, are obtained as follows:

$$\begin{array}{l} \mathbf{H}\underline{v}_1 = \lambda_1 \underline{v}_1 \Rightarrow \begin{bmatrix} a & -\frac{b^2}{\rho} \\ -c_1^2 & -a \end{bmatrix} \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} = \lambda_1 \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} \\ \Leftrightarrow \left\{ \begin{array}{l} a\, v_{11} - \frac{b^2}{\rho}\, v_{12} = \lambda_1\, v_{11} \\ -c_1^2\, v_{11} - a\, v_{12} = \lambda_1\, v_{12} \end{array} \right. \\ \Leftrightarrow \left\{ \begin{array}{l} v_{11}\, (a - \lambda_1) = \frac{b^2}{\rho}\, v_{12} \\ -c_1^2\, v_{11} = v_{12}\, (a + \lambda_1) \end{array} \right. \end{array} \tag{3.66}$$

From the first equation we can choose for example the following components for eigenvector $\underline{v}_1$:

$$\underline{v}_1 = \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} = \begin{bmatrix} \frac{1}{a-\lambda_1} \\ \frac{\rho}{b^2} \end{bmatrix} \text{ where } \lambda_1 = +\sqrt{a^2 + \frac{(c_1 b)^2}{\rho}} \tag{3.67}$$

We can check that this choice for v_{11} and v_{12} is compatible with the second equation. Indeed:

$$-c_1^2\, v_{11} = v_{12}\,(a+\lambda_1) \Rightarrow \tfrac{-c_1^2}{a-\lambda_1} = \tfrac{\rho}{b^2}\,(a+\lambda_1) \Rightarrow a^2 - \lambda_1^2 = -\tfrac{(c_1 b)^2}{\rho} \tag{3.68}$$

Changing λ_1 by λ_2 leads to a possible choice of the components of eigenvector $\underline{v}_2$:

$$\underline{v}_2 = \begin{bmatrix} v_{21} \\ v_{22} \end{bmatrix} = \begin{bmatrix} \frac{1}{a-\lambda_2} \\ \frac{\rho}{b^2} \end{bmatrix} \text{ where } \lambda_2 = -\sqrt{a^2 + \frac{(c_1 b)^2}{\rho}} \tag{3.69}$$

As far as λ_2 is the eigenvalue of $\mathbf{H}$ in the left half plane, we conclude that λ_2 will be the closed-loop eigenvalue once the optimal control has been applied (notice that we don't know so far the expression of the optimal control !).

As far as $\underline{v}_2$ is the eigenvector of $\mathbf{H}$ corresponding to the eigenvalue in the left half plane, we split it as follows:

$$\underline{v}_2 = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{a-\lambda_2} \\ \frac{\rho}{b^2} \end{bmatrix} \text{ where } \lambda_2 = -\sqrt{a^2 + \frac{(c_1 b)^2}{\rho}} \tag{3.70}$$

Then the solution of the algebraic Riccati equation which leads to the computation of the optimal control reads:

$$\mathbf{P} = \mathbf{X}_2 \mathbf{X}_1^{-1} = \frac{\rho}{b^2}\,(a-\lambda_2) \text{ where } \lambda_2 = -\sqrt{a^2 + \frac{(c_1 b)^2}{\rho}} \tag{3.71}$$

Thus:

$$\mathbf{P} = \frac{\rho}{b^2}\left(a + \sqrt{a^2 + \frac{(c_1 b)^2}{\rho}}\right) \tag{3.72}$$

We will check those results by using the algebraic Ricatti equation, which reads:

$$\begin{array}{l} \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \\ \Leftrightarrow 2a\mathbf{P} - \frac{b^2}{\rho}\mathbf{P}^2 + c_1^2 = 0 \\ \Leftrightarrow \frac{b^2}{\rho}\mathbf{P}^2 - 2a\mathbf{P} - c_1^2 = 0 \end{array} \tag{3.73}$$

The roots of this quadratic equation are:

$$\begin{cases} \mathbf{P}_1 = \dfrac{2a+\sqrt{4a^2+4\frac{(c_1 b)^2}{\rho}}}{2\frac{b^2}{\rho}} = \frac{\rho}{b^2}\left(a + \sqrt{a^2 + \frac{(c_1 b)^2}{\rho}}\right) > 0 \\ \mathbf{P}_2 = \dfrac{2a-\sqrt{4a^2+4\frac{(c_1 b)^2}{\rho}}}{2\frac{b^2}{\rho}} = \frac{\rho}{b^2}\left(a - \sqrt{a^2 + \frac{(c_1 b)^2}{\rho}}\right) < 0 \end{cases} \tag{3.74}$$

It is clear that $\mathbf{P}_1$ is the positive definite solution of the algebraic Riccati equation (ARE). Thus we retrieve the result (3.72):

$$\mathbf{P} := \mathbf{P}_1 = \frac{\rho}{b^2}\left(a + \sqrt{a^2 + \frac{(c_1 b)^2}{\rho}}\right) \tag{3.75}$$

Furthermore we are now in position to compute the feedback gain $\mathbf{K}$:

$$\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} = \frac{b}{\rho}\frac{\rho}{b^2}\left(a + \sqrt{a^2 + \frac{(c_1 b)^2}{\rho}}\right) = \frac{1}{b}\left(a + \sqrt{a^2 + \frac{(c_1 b)^2}{\rho}}\right) \tag{3.76}$$

Finally, the eigenvalue of the feedback loop reads:

$$\begin{aligned} spec\,(\mathbf{A} - \mathbf{BK}) &= \mathbf{A} - \mathbf{BK} = a - b\tfrac{1}{b}\left(a + \sqrt{a^2 + \tfrac{(c_1 b)^2}{\rho}}\right) \\ &= -\sqrt{a^2 + \tfrac{(c_1 b)^2}{\rho}} \end{aligned} \tag{3.77}$$

We obviously retrieve the result (3.69) obtained through the Hamiltonian matrix $\mathbf{H}$.

3.7 Hamiltonian matrix properties

Let $\mathbf{H}$ be the following Hamiltonian matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{G} \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \text{ where } \mathbf{G} = \mathbf{G}^T, \quad \mathbf{Q} = \mathbf{Q}^T \tag{3.78}$$

By definition, a matrix $\mathbf{H}$ is said to be an Hamiltonian matrix as soon as the following property holds:

$$(\mathbf{JH})^T = \mathbf{JH} \tag{3.79}$$

where $\mathbf{J}$ is the following skew-symmetric matrix:

$$\mathbf{J} = -\mathbf{J}^T = \begin{bmatrix} \mathbf{0} & \mathbb{I} \\ -\mathbb{I} & \mathbf{0} \end{bmatrix} \tag{3.80}$$

Any matrix $\mathbf{S} \in \mathbb{R}^{2n \times 2n}$ satisfying the following relation is called a symplectic matrix:

$$\mathbf{S}^T\mathbf{JS} = \mathbf{SJS}^T = \mathbf{J} \tag{3.81}$$

If $\mathbf{H}$ has no eigenvalues on the imaginary axis, then the invariant subspace $\mathcal{X}$ corresponding to the n (counting multiplicities) eigenvalues in the open left half plane is called the stable invariant subspace of $\mathbf{H}$. Let $\mathbf{X}$ be a matrix whose columns form a basis for $\mathcal{X}$. If the columns of $\mathbf{X}$ form an *orthonormal* basis for

$\mathcal{X}$, then $\begin{bmatrix} \mathbf{X} & \mathbf{JX} \end{bmatrix}$ is orthogonal and the following Hamiltonian block-Schur decomposition is obtained[4]:

$$\begin{bmatrix} \mathbf{X} & \mathbf{JX} \end{bmatrix}^T \mathbf{H} \begin{bmatrix} \mathbf{X} & \mathbf{JX} \end{bmatrix} = \begin{bmatrix} \mathbf{T} & -\mathbf{G} \\ \mathbf{0} & -\mathbf{T}^T \end{bmatrix} \tag{3.82}$$

where $\mathbf{T} \in \mathbb{R}^{n \times n}$ is an upper triangular matrix (we said that $\mathbf{T}$ has a real Schur form):

$$\mathbf{T} = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1n} \\ 0 & t_{21} & \ddots & \\ \vdots & \ddots & \ddots & \\ 0 & 0 & \cdots & t_{nn} \end{bmatrix} \tag{3.83}$$

Moreover, given Hamiltonian matrix $\mathbf{H}$ defined in (3.78), there is always a corresponding algebraic Riccati equation (ARE)[4]:

$$\begin{array}{c} \mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{G} \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \text{ where } \mathbf{G} = \mathbf{G}^T, \quad \mathbf{Q} = \mathbf{Q}^T \\ \text{Corresponding ARE} : \mathbf{A}^T\mathbf{P} + \mathbf{PA} - \mathbf{PGP} + \mathbf{Q} = \mathbf{0} \end{array} \tag{3.84}$$

Indeed, we can always write the algebraic Riccati equation (ARE) as follows:

$$\mathbf{A}^T\mathbf{P} + \mathbf{PA} - \mathbf{PGP} + \mathbf{Q} = \begin{bmatrix} \mathbf{P} & -\mathbb{I}_n \end{bmatrix} \mathbf{H} \begin{bmatrix} \mathbb{I}_n \\ \mathbf{P} \end{bmatrix} = \mathbf{0} \tag{3.85}$$

Now assume that $\mathbf{P} = \mathbf{P}^T$ is a symmetric solution of the algebraic Riccati equation (ARE). Then it is easy to see that the following relation holds:

$$\mathbf{H} \begin{bmatrix} \mathbb{I}_n & \mathbf{0} \\ \mathbf{P} & \mathbb{I}_n \end{bmatrix} = \begin{bmatrix} \mathbb{I}_n & \mathbf{0} \\ \mathbf{P} & \mathbb{I}_n \end{bmatrix} \begin{bmatrix} \mathbf{A} - \mathbf{GP} & -\mathbf{G} \\ \mathbf{0} & -(\mathbf{A} - \mathbf{GP})^T \end{bmatrix} \tag{3.86}$$

Hence $\mathbf{H} \begin{bmatrix} \mathbb{I}_n \\ \mathbf{P} \end{bmatrix} = \begin{bmatrix} \mathbb{I}_n \\ \mathbf{P} \end{bmatrix} (\mathbf{A} - \mathbf{GP})$. Thus the columns of $\begin{bmatrix} \mathbb{I}_n \\ \mathbf{P} \end{bmatrix}$ span the $\mathbf{H}$-invariant subspace corresponding to $\lambda(\mathbf{H}) \cap \lambda(\mathbf{A} - \mathbf{GP})$. This implies that AREs can be solved by computing $\mathbf{H}$-invariant subspaces.

Finally, let $\mathbf{G} = \mathbf{BR}^{-1}\mathbf{B}^T$ and $\mathbf{Q} = \mathbf{N}^T\mathbf{N}$. Thus Hamiltonian matrix (3.78) reads:

$$\left\{ \begin{array}{l} \mathbf{G} = \mathbf{BR}^{-1}\mathbf{B}^T \\ \mathbf{Q} = \mathbf{N}^T\mathbf{N} \end{array} \right. \Rightarrow \mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{G} \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{BR}^{-1}\mathbf{B}^T \\ -\mathbf{N}^T\mathbf{N} & -\mathbf{A}^T \end{bmatrix} \tag{3.87}$$

where $\mathbf{R} = \mathbf{R}^T > 0 \Rightarrow \mathbf{R} = \left(\mathbf{R}^{0.5}\right)^T \mathbf{R}^{0.5}$ and $\mathbf{R}^{-0.5} = \left(\mathbf{R}^{-0.5}\right)^T$.

[4] Peter Benner, Daniel Kressner, Volker Mehrmann, Skew-Hamiltonian and Hamiltonian Eigenvalue Problems: Theory, Algorithms and Applications, January 2005, Proceedings of the Conference on Applied Mathematics and Scientific Computing (pp.3-39), DOI: 10.1007/1-4020-3197-1_1

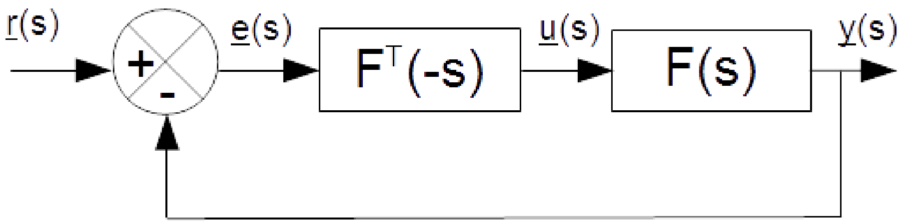


Figure 3.1: Closed-loop Hamiltonian transfer function

Then $\mathbf{GP} = \mathbf{BR}^{-1}\mathbf{B}^T\mathbf{P} := \mathbf{BK}$ where $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$ and relation(3.86) reads as follows:

$$\begin{aligned}&\begin{bmatrix} \mathbf{A} & -\mathbf{BR}^{-1}\mathbf{B}^T \\ -\mathbf{N}^T\mathbf{N} & -\mathbf{A}^T \end{bmatrix} \begin{bmatrix} \mathbb{I}_n & \mathbf{0} \\ \mathbf{P} & \mathbb{I}_n \end{bmatrix} \\ &= \begin{bmatrix} \mathbb{I}_n & \mathbf{0} \\ \mathbf{P} & \mathbb{I}_n \end{bmatrix} \begin{bmatrix} \mathbf{A}-\mathbf{BK} & -\mathbf{BR}^{-1}\mathbf{B}^T \\ \mathbf{0} & -(\mathbf{A}-\mathbf{BK})^T \end{bmatrix}\end{aligned} \tag{3.88}$$

From the preceding relation, and using the fact that $\det\left(\begin{bmatrix} \mathbb{I}_n & \mathbf{0} \\ \mathbf{P} & \mathbb{I}_n \end{bmatrix}\right) = 1$, we get:

$$\det(s\mathbb{I} - \mathbf{H}) = (-1)^n \beta(s)\,\beta(-s) \text{ where } \beta(s) := \det(s\mathbb{I} - \mathbf{A} + \mathbf{BK}) \tag{3.89}$$

Furthermore let $(\mathbf{A}, \mathbf{B}, \mathbf{N})$ be the realization of a strictly proper transfer matrix $\mathbf{F}(s)$:

$$\mathbf{F}(s) = \left(\begin{array}{c|c} \mathbf{A} & \mathbf{BR}^{-0.5} \\ \hline \mathbf{N} & \mathbf{0} \end{array}\right) := \mathbf{N}(s\mathbb{I} - \mathbf{A})^{-1}\mathbf{BR}^{-0.5} \tag{3.90}$$

Then is can be shown that the following relation holds:

$$\left(\mathbb{I} + \mathbf{F}(s)\mathbf{F}^T(-s)\right)^{-1} = \begin{bmatrix} -\mathbf{N} & \mathbf{0} \end{bmatrix} (s\mathbb{I} - \mathbf{H})^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{N}^T \end{bmatrix} + \mathbb{I} \tag{3.91}$$

To get this result, consider Figure 3.1. The relation between $\underline{e}(s)$ and $\underline{r}(s)$ is obtained by reading Figure 3.1 *against* the arrows:

$$\begin{aligned} \underline{e}(s) &= \underline{r}(s) - \mathbf{F}(s)\mathbf{F}^T(-s)\underline{e}(s) \\ \Rightarrow \underline{e}(s) &= \left(\mathbb{I} + \mathbf{F}(s)\mathbf{F}^T(-s)\right)^{-1} \underline{r}(s) \end{aligned} \tag{3.92}$$

On the other hand, the realization of $\mathbf{F}^T(-s)$ is obtained from the realization of $\mathbf{F}(s)$ as follows:

$$\begin{aligned} \mathbf{F}(s) &= \left(\begin{array}{c|c} \mathbf{A} & \mathbf{BR}^{-0.5} \\ \hline \mathbf{N} & \mathbf{0} \end{array}\right) := \mathbf{N}(s\mathbb{I} - \mathbf{A})^{-1}\mathbf{BR}^{-0.5} \\ \Rightarrow \mathbf{F}^T(-s) &= \left(\mathbf{N}(-s\mathbb{I} - \mathbf{A})^{-1}\mathbf{BR}^{-0.5}\right)^T = -\mathbf{R}^{-0.5}\mathbf{B}^T\left(s\mathbb{I} - \left(-\mathbf{A}^T\right)\right)^{-1}\mathbf{N}^T \\ &= \left(\begin{array}{c|c} -\mathbf{A}^T & \mathbf{N}^T \\ \hline -\mathbf{R}^{-0.5}\mathbf{B}^T & \mathbf{0} \end{array}\right) \end{aligned} \tag{3.93}$$

Thus, in the time domain we have:

$$\begin{cases} \mathbf{F}(s) = \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B}\mathbf{R}^{-0.5} \\ \hline \mathbf{N} & \mathbf{0} \end{array}\right) \Rightarrow \begin{cases} \underline{\dot{x}}_1 = \mathbf{A}\underline{x}_1 + \mathbf{B}\mathbf{R}^{-0.5}\underline{u} \\ \underline{y} = \mathbf{N}\underline{x}_1 \end{cases} \\ \mathbf{F}^T(-s) = \left(\begin{array}{c|c} -\mathbf{A}^T & \mathbf{N}^T \\ \hline -\mathbf{R}^{-0.5}\mathbf{B}^T & \mathbf{0} \end{array}\right) \Rightarrow \begin{cases} \underline{\dot{x}}_2 = -\mathbf{A}^T\underline{x}_2 + \mathbf{N}^T\underline{e} \\ \underline{u} = -\mathbf{R}^{-0.5}\mathbf{B}^T\underline{x}_2 \end{cases} \end{cases} \tag{3.94}$$

From Figure 3.1 we see that $\underline{e} = \underline{r} - \underline{y}$. Thus the realization of Figure 3.1 reads as follows:

$$\begin{cases} \underline{e} = \underline{r} - \underline{y} \\ \underline{u} = -\mathbf{R}^{-0.5}\mathbf{B}^T\underline{x}_2 \\ \underline{y} = \mathbf{N}\underline{x}_1 \end{cases} \Rightarrow \begin{cases} \begin{bmatrix} \underline{\dot{x}}_1 \\ \underline{\dot{x}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{N}^T\mathbf{N} & -\mathbf{A}^T \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{N}^T \end{bmatrix} \underline{r} \\ \underline{e} = \begin{bmatrix} -\mathbf{N} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} + \mathbb{I}\,\underline{r} \end{cases} \tag{3.95}$$

In the frequency domain we get:

$$\underline{e}(s) = \left(\begin{bmatrix} -\mathbf{N} & \mathbf{0} \end{bmatrix} (s\mathbb{I} - \mathbf{H})^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{N}^T \end{bmatrix} + \mathbb{I} \right) \underline{r}(s) \tag{3.96}$$

When identifying (3.92) with (3.96) we get relation (3.91). An alternate relation can also be obtained by replacing $\mathbf{F}^T(-s)\mathbf{F}(s)$ in Figure 3.1 by $\mathbf{F}(s)\mathbf{F}^T(-s)$. Then we get:

$$\left(\mathbb{I} + \mathbf{F}^T(-s)\mathbf{F}(s)\right)^{-1} = \begin{bmatrix} \mathbf{0} & -\mathbf{R}^{-0.5}\mathbf{B}^T \end{bmatrix} (s\mathbb{I} - \mathbf{H})^{-1} \begin{bmatrix} \mathbf{B}\mathbf{R}^{-0.5} \\ \mathbf{0} \end{bmatrix} + \mathbb{I} \tag{3.97}$$

Having in mind that for any square invertible matrix $\mathbf{Y}$ we have $\mathbf{X}\mathbf{Y}^{-1}\mathbf{Z} = \frac{\mathbf{X}\,\mathrm{adj}(\mathbf{Y})\mathbf{Z}}{\det(\mathbf{Y})}$ (here $\mathbf{X} = \begin{bmatrix} \mathbf{0} & -\mathbf{R}^{-0.5}\mathbf{B}^T \end{bmatrix}$, $\mathbf{Y} = (s\mathbb{I} - \mathbf{H})$ and $\mathbf{Z} = \begin{bmatrix} \mathbf{B}\mathbf{R}^{-0.5} \\ \mathbf{0} \end{bmatrix}$), we conclude that relation (3.91) indicates that the eigenvalues of Hamiltonian matrix $\mathbf{H}$ are the roots of $\det\left(\mathbb{I} + \mathbf{F}(s)\mathbf{F}^T(-s)\right)$.

Moreover, let:

$$\mathbf{F}(s) = \mathbf{N}\,(s\mathbb{I} - \mathbf{A})^{-1}\,\mathbf{B}\mathbf{R}^{-0.5} = \frac{\mathbf{N}\,\mathrm{adj}\,(s\mathbb{I} - \mathbf{A})\,\mathbf{B}\mathbf{R}^{-0.5}}{\det\,(s\mathbb{I} - \mathbf{A})} := \frac{\mathbf{N}_{ol}(s)}{D(s)} \tag{3.98}$$

Then:

$$\begin{aligned} \det\left(\mathbb{I} + \mathbf{F}(s)\mathbf{F}^T(-s)\right) &= \det\left(\mathbb{I} + \frac{\mathbf{N}_{ol}(s)}{D(s)}\frac{\mathbf{N}_{ol}^T(-s)}{D(-s)}\right) \\ &= \det\left(\frac{D(s)D(-s)\,\mathbb{I} + \mathbf{N}_{ol}(s)\mathbf{N}_{ol}^T(-s)}{D(s)D(-s)}\right) \end{aligned} \tag{3.99}$$

Consequently, the eigenvalues of the Hamiltonian matrix $\mathbf{H}$ are the roots of $\det\left(D(s)D(-s)\,\mathbb{I} + \mathbf{N}_{ol}(s)\mathbf{N}_{ol}^T(-s)\right)$:

$$\boxed{\det\,(s\mathbb{I} - \mathbf{H})\big|_{s=\lambda} = 0 \Leftrightarrow \det\left(D(s)D(-s)\,\mathbb{I} + \mathbf{N}_{ol}(s)\mathbf{N}_{ol}^T(-s)\right)\big|_{s=\lambda} = 0} \tag{3.100}$$

Alternatively, the preceding relation indicates that $D(\lambda)D(-\lambda)$ is an eigenvalue of matrix $-\mathbf{N}_{ol}(\lambda)\mathbf{N}_{ol}^T(-\lambda)$. This remark may be used for design purposes, especially to select matrix $\mathbf{N}$ to achieve some specified closed-loop eigenvalues (we recall that weighting matrix $\mathbf{Q}$ is given by $\mathbf{Q} = \mathbf{N}^T\mathbf{N}$).

3.8 Discrete time LQ regulator

3.8.1 Finite horizon LQ regulator

There is an equivalent theory for discrete time systems. Indeed, for the system:

$$\begin{cases} \underline{x}(k+1) = \mathbf{A}\underline{x}(k) + \mathbf{B}\underline{u}(k) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{3.101}$$

with an equivalent performance criteria:

$$J(\underline{u}(k)) = \frac{1}{2}\underline{x}^T(N)\mathbf{S}\underline{x}(N) + \frac{1}{2}\sum_{k=0}^{N-1} \underline{x}^T(k)\mathbf{Q}\underline{x}(k) + \underline{u}^T(k)\mathbf{R}\underline{u}(k) \tag{3.102}$$

Where $\mathbf{Q} \geq 0$ is a constant positive definite matrix and $\mathbf{R} > 0$ a constant positive definite matrix. The optimal control is given by:

$$\underline{u}(k) = -\mathbf{K}(k)\underline{x}(k) \tag{3.103}$$

Where:

$$\mathbf{K}(k) = \left(\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}(k+1)\mathbf{A} \tag{3.104}$$

And $\mathbf{P}(k)$ is given by the solution of the discrete time Riccati equation:

$$\begin{cases} \mathbf{P}(k) = \mathbf{A}^T\mathbf{P}(k+1)\mathbf{A} + \mathbf{Q} - \mathbf{A}^T\mathbf{P}(k+1)\mathbf{B}\left(\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}(k+1)\mathbf{A} \\ \mathbf{P}(N) = \mathbf{S} \end{cases} \tag{3.105}$$

3.8.2 Finite horizon LQ regulator with zero terminal state

We consider the following performance criteria to be minimized:

$$J(\underline{u}(k)) = \frac{1}{2}\sum_{k=0}^{N-1} \underline{x}^T(k)\mathbf{Q}\underline{x}(k) + \underline{u}^T(k)\mathbf{R}\underline{u}(k) + 2\underline{x}^T(k)\mathbf{S}\underline{u}(k) \tag{3.106}$$

With the constraint on terminal state:

$$\underline{x}(N) = \underline{0} \tag{3.107}$$

We will assume that matrices $\mathbf{R} > 0$ and $\mathbf{Q} - \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T \geq 0$ are symmetric. Ntogramatzidis[2] has shown the results presented hereafter: denote by $\mathbf{P}_1$ and $\mathbf{P}_2$ the positive definite solutions of the following continuous time algebraic Riccati equations:

$$\begin{cases} \mathbf{0} = & \mathbf{P}_1 + \left(\mathbf{A}^T\mathbf{P}_1\mathbf{B} + \mathbf{S}\right)\left(\mathbf{R} + \mathbf{B}^T\mathbf{P}_1\mathbf{B}\right)^{-1}\left(\mathbf{B}^T\mathbf{P}_1\mathbf{A} + \mathbf{S}^T\right) \\ & -\mathbf{A}^T\mathbf{P}_1\mathbf{A} - \mathbf{Q} \\ \mathbf{0} = & \mathbf{P}_2 + \left(\mathbf{A}_b^T\mathbf{P}_2\mathbf{B}_b + \mathbf{S}_b\right)\left(\mathbf{R}_b + \mathbf{B}_b^T\mathbf{P}_2\mathbf{B}_b\right)^{-1}\left(\mathbf{B}_b^T\mathbf{P}_2\mathbf{A}_b + \mathbf{S}_b^T\right) \\ & -\mathbf{A}_b^T\mathbf{P}_2\mathbf{A}_b - \mathbf{Q}_b \end{cases} \tag{3.108}$$

Where:

$$\begin{cases} \mathbf{A}_b = \mathbf{A}^{-1} \\ \mathbf{B}_b = -\mathbf{A}^{-1}\mathbf{B} \\ \mathbf{Q}_b = \mathbf{A}^{-T}\mathbf{Q}\mathbf{A}^{-1} \\ \mathbf{R}_b = \mathbf{R} - \mathbf{S}^T\mathbf{A}^{-1}\mathbf{B} - \mathbf{B}^T\mathbf{A}^{-T}\mathbf{S} + \mathbf{B}^T\mathbf{A}^{-T}\mathbf{Q}\mathbf{A}^{-1}\mathbf{B} \\ \mathbf{S}_b = \mathbf{A}^{-T}\mathbf{S} - \mathbf{A}^{-T}\mathbf{Q}\mathbf{A}^{-1}\mathbf{B} \end{cases} \tag{3.109}$$

We will denote by $\mathbf{K}_1$ and $\mathbf{K}_2$ the following *infinite* horizon gain matrices:

$$\begin{cases} \mathbf{K}_1 = \left(\mathbf{R} + \mathbf{B}^T\mathbf{P}_1\mathbf{B}\right)^{-1}\left(\mathbf{B}^T\mathbf{P}_1\mathbf{A} + \mathbf{S}^T\right) \\ \mathbf{K}_2 = \left(\mathbf{R}_b + \mathbf{B}_b^T\mathbf{P}_2\mathbf{B}_b\right)^{-1}\left(\mathbf{B}_b^T\mathbf{P}_2\mathbf{A}_b + \mathbf{S}_b^T\right) \end{cases} \tag{3.110}$$

Then the optimal control is:

$$\underline{u}(k) = \begin{cases} -\mathbf{K}(k)\underline{x}(k) \;\; \forall\, 0 \le k < N \\ 0 \text{ for } k = N \end{cases} \tag{3.111}$$

Where:

$$\begin{cases} \mathbf{K}(k) = \left(\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}\right)^{-1}\left(\mathbf{B}^T\mathbf{P}(k+1)\mathbf{A} + \mathbf{S}^T\right) \\ \mathbf{P}(k) = \mathbf{X}_2(k)\mathbf{X}_1^{-1}(k) \end{cases} \tag{3.112}$$

And:

$$\begin{cases} \mathbf{X}_1(k) = (\mathbf{A} - \mathbf{B}\mathbf{K}_1)^k - (\mathbf{A}_b - \mathbf{B}_b\mathbf{K}_2)^{(k-N)}(\mathbf{A} - \mathbf{B}\mathbf{K}_1)^N \\ \mathbf{X}_2(k) = \mathbf{P}_1(\mathbf{A} - \mathbf{B}\mathbf{K}_1)^k + \mathbf{P}_2(\mathbf{A}_b - \mathbf{B}_b\mathbf{K}_2)^{(k-N)}(\mathbf{A} - \mathbf{B}\mathbf{K}_1)^N \end{cases} \tag{3.113}$$

Matrix $\mathbf{P}(k)$ satisfy the following Riccati difference equation:

$$\mathbf{P}(k) + \left(\mathbf{A}^T\mathbf{P}(k+1)\mathbf{B} + \mathbf{S}\right)\left(\mathbf{R} + \mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}\right)^{-1}\left(\mathbf{B}^T\mathbf{P}(k+1)\mathbf{A} + \mathbf{S}^T\right) - \mathbf{A}^T\mathbf{P}(k+1)\mathbf{A} - \mathbf{Q} = 0 \tag{3.114}$$

Furthermore the optimal state $\underline{x}(k)$ and costate $\lambda(k)$ have the following expressions:

$$\begin{cases} \underline{x}(k+1) = (\mathbf{A} - \mathbf{B}\mathbf{K}_1)\, e_1(k) - (\mathbf{A}_b - \mathbf{B}_b\mathbf{K}_2)\, e_2(k) \\ \lambda(k+1) = \mathbf{P}_1(\mathbf{A} - \mathbf{B}\mathbf{K}_1)\, e_1(k) + \mathbf{P}_2(\mathbf{A}_b - \mathbf{B}_b\mathbf{K}_2)\, e_2(k) \end{cases} \tag{3.115}$$

Where:

$$\begin{cases} e_1(k) = (\mathbf{A} - \mathbf{B}\mathbf{K}_1)^k\, \mathbf{X}_1^{-1}(0)\underline{x}_0 \\ e_2(k) = (\mathbf{A}_b - \mathbf{B}_b\mathbf{K}_2)^{(k-N)}(\mathbf{A} - \mathbf{B}\mathbf{K}_1)^N\, \mathbf{X}_1^{-1}(0)\underline{x}_0 \end{cases} \tag{3.116}$$

3.8.3 Infinite horizon LQ regulator

For the infinite horizon problem $N \to \infty$. We will assume that the performance criteria to be minimized is:

$$J(\underline{u}(k)) = \frac{1}{2}\sum_{k=0}^{\infty} \underline{x}^T(k)\mathbf{Q}\underline{x}(k) + \underline{u}^T(k)\mathbf{R}\underline{u}(k) \tag{3.117}$$

Then matrix $\mathbf{P}$ satisfies the following discrete time algebraic Riccati equation:

$$\mathbf{P} + \mathbf{A}^T\mathbf{P}\mathbf{B}\left(\mathbf{R} + \mathbf{B}^T\mathbf{P}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}\mathbf{A} - \mathbf{A}^T\mathbf{P}\mathbf{A} - \mathbf{Q} = \mathbf{0} \tag{3.118}$$

And the discrete time control $\underline{u}(k)$ is given by:

$$\underline{u}(k) = -\mathbf{K}\underline{x}(k) \tag{3.119}$$

Where:

$$\mathbf{K} = \left(\mathbf{R} + \mathbf{B}^T\mathbf{P}\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}\mathbf{A} \tag{3.120}$$

If $(\mathbf{A}, \mathbf{B})$ is stabilizable, then the closed-loop system is stable, meaning that all the eigenvalues of $(\mathbf{A} - \mathbf{B}\mathbf{K})$, with $\mathbf{K}$ given by (3.120), will lie within the unit disk (i.e. have magnitudes less than 1). Let's define the following symplectic matrix[5]:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A}^{-1} & \mathbf{A}^{-1}\mathbf{G} \\ \mathbf{Q}\mathbf{A}^{-1} & \mathbf{A}^T + \mathbf{Q}\mathbf{A}^{-1}\mathbf{G} \end{bmatrix} \tag{3.121}$$

Where:

$$\mathbf{G} = \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \tag{3.122}$$

A symplectic matrix is a matrix which satisfies:

$$\mathbf{H}^T\mathbf{J}\mathbf{H} = \mathbf{J} \text{ where } \mathbf{J} = \begin{bmatrix} 0 & \mathbb{I} \\ -\mathbb{I} & 0 \end{bmatrix} \text{ and } \mathbf{J}^{-1} = -\mathbf{J} \tag{3.123}$$

This implies:

$$\begin{array}{c} \mathbf{H}^T\mathbf{J} = \mathbf{J}\mathbf{H}^{-1} \Leftrightarrow \mathbf{J}^{-1}\mathbf{H}^T\mathbf{J} = \mathbf{H}^{-1} \\ \Rightarrow \mathbf{H}^{-1} = \begin{bmatrix} \mathbf{A} + \mathbf{G}\mathbf{A}^{-T}\mathbf{Q} & -\mathbf{G}\mathbf{A}^{-T} \\ -\mathbf{A}^{-T}\mathbf{Q} & \mathbf{A}^{-T} \end{bmatrix} \end{array} \tag{3.124}$$

Where $\mathbf{A}^{-T} = (\mathbf{A}^{-1})^T$. Under detectability and stabilizability assumptions, it can be shown that the eigenvalues of the closed-loop plant (that are the eigenvalues of $\mathbf{A} - \mathbf{B}\mathbf{K}$) are equal to the n eigenvalues inside the unit circle of the Hamiltonian matrix $\mathbf{H}$. The optimal control stabilizes the plant. Furthermore if the $2n \times n$ matrix $\begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix}$ has columns that comprise all the eigenvectors associated with the n eigenvalues of the Hamiltonian matrix $\mathbf{H}$ outside the unit circle (unstable eigenvalues) then $\mathbf{X}_1$ is invertible and the positive definite solution of the *algebraic Riccati equation (ARE)* is:

$$\mathbf{P} = \mathbf{X}_2\mathbf{X}_1^{-1} \tag{3.125}$$

Thus matrix $\mathbf{P}$ for the optimal steady state feedback can be computed thanks to the unstable (eigenvalues outside the unit circle) eigenvectors of $\mathbf{H}$ or the stable (eigenvalues inside the unit circle) eigenvectors of $\mathbf{H}^{-1}$.

[5]Alan J. Laub, A Schur Method for Solving Algebraic Riccati equations, IEEE Transactions On Automatic Control, VOL. AC-24, NO. 6, December 1979

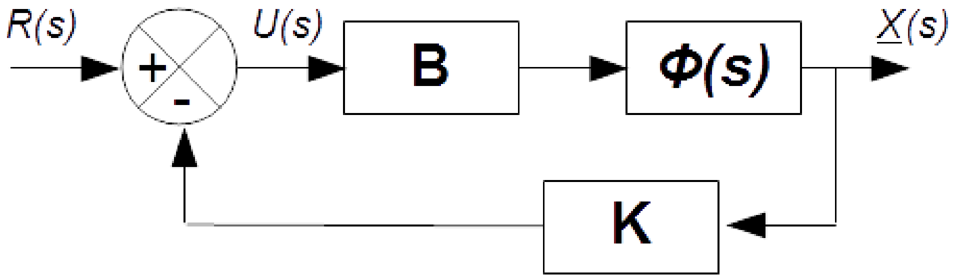


Figure 3.2: Full-state feedback control

3.9 Robustness property

3.9.1 Hsu-Chen theorem

Let's consider a linear plant controlled through a state feedback as follows:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{u}(t) = -\mathbf{K}\underline{x}(t) + r(t) \end{cases} \tag{3.126}$$

The dynamics of the closed-loop system reads:

$$\dot{\underline{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K})\,\underline{x}(t) + \mathbf{B}r(t) \tag{3.127}$$

In order to compute the closed-loop transfer matrix between $\underline{X}(s)$ and $R(s)$ we take the Laplace transform of (3.126) assuming no initial condition:

$$\begin{array}{l} s\underline{X}(s) = \mathbf{A}\underline{X}(s) + \mathbf{B}(-\mathbf{K}\underline{X}(s) + R(s)) \\ \Rightarrow \underline{X}(s)(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}) = \mathbf{B}R(s) \\ \Rightarrow \underline{X}(s) = (s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K})^{-1}\mathbf{B}R(s) \end{array} \tag{3.128}$$

On the other hand, let $\mathbf{\Phi}(s)$ be resolvent of the state (transition) matrix $\mathbf{A}$. Matrix $\mathbf{\Phi}(s)$ is defined as follows:

$$\mathbf{\Phi}(s) = (s\mathbb{I} - \mathbf{A})^{-1} \tag{3.129}$$

The block diagram of the full-state feedback control is shown in Figure 3.2. We get:

$$\begin{array}{rl} \underline{X}(s) & = \mathbf{\Phi}(s)\mathbf{B}(R(s) - \mathbf{K}\underline{X}(s)) \\ & = (\mathbb{I} + \mathbf{\Phi}(s)\mathbf{B}\mathbf{K})^{-1}\mathbf{\Phi}(s)\mathbf{B}R(s) \end{array} \tag{3.130}$$

Using the fact that $(\mathbf{A}\mathbf{B})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$ we get:

$$\begin{array}{rl} \underline{X}(s) & = (\mathbf{\Phi}^{-1}(s)(\mathbb{I} + \mathbf{\Phi}(s)\mathbf{B}\mathbf{K}))^{-1}\mathbf{B}R(s) \\ & = (\mathbf{\Phi}^{-1}(s) + \mathbf{B}\mathbf{K})^{-1}\mathbf{B}R(s) \\ & = (s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K})^{-1}\mathbf{B}R(s) \end{array} \tag{3.131}$$

The open-loop characteristic polynomial is given by:

$$\det(s\mathbb{I} - \mathbf{A}) = \det\left(\mathbf{\Phi}^{-1}(s)\right) \tag{3.132}$$

Whereas the closed-loop characteristic polynomial is given by:

$$\det\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\right) \tag{3.133}$$

Sylvester's determinant theorem[6] states that the following relation holds where $\mathbf{M}_1$ is an $m \times n$ matrix and $\mathbf{M}_2$ an $n \times m$ matrix (so that $\mathbf{M}_1$ and $\mathbf{M}_2$ have dimensions allowing them to be multiplied in either order forming a square matrix):

$$\det\left(\mathbb{I}_m + \mathbf{M}_1\mathbf{M}_2\right) = \det\left(\mathbb{I}_n + \mathbf{M}_2\mathbf{M}_1\right) \tag{3.134}$$

Sylvester's determinant theorem may be proven using the Schur's formula, which is recalled hereafter:

$$\begin{aligned}\det\left(\begin{bmatrix}\mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{21} & \mathbf{A}_{22}\end{bmatrix}\right) &= \det\left(\mathbf{A}_{22}\right)\det\left(\mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21}\right)\\ &= \det\left(\mathbf{A}_{11}\right)\det\left(\mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\right)\end{aligned} \tag{3.135}$$

Thus if $\mathbf{M} = \begin{bmatrix}\mathbb{I}_m & -\mathbf{M}_1\\ \mathbf{M}_2 & \mathbb{I}_n\end{bmatrix}$, we get:

$$\begin{aligned}\det\left(\mathbf{M}\right) = \det\left(\begin{bmatrix}\mathbb{I}_m & -\mathbf{M}_1\\ \mathbf{M}_2 & \mathbb{I}_n\end{bmatrix}\right) &= \det\left(\mathbb{I}_m + \mathbf{M}_1\mathbf{M}_2\right)\\ &= \det\left(\mathbb{I}_n + \mathbf{M}_2\mathbf{M}_1\right)\end{aligned} \tag{3.136}$$

In addition, for square matrices $\mathbf{M}_3$ and $\mathbf{M}_4$ of equal size, the determinant of the matrix product equals the product of their determinants:

$$\det\left(\mathbf{M}_3\mathbf{M}_4\right) = \det\left(\mathbf{M}_3\right)\det\left(\mathbf{M}_4\right) \tag{3.137}$$

Then we get:

$$\begin{aligned}\det\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\right) &= \det\left(\left(s\mathbb{I} - \mathbf{A}\right)\left(\mathbb{I} + \left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B}\mathbf{K}\right)\right)\\ &= \det\left(\left(s\mathbb{I} - \mathbf{A}\right)\left(\mathbb{I} + \mathbf{\Phi}(s)\mathbf{B}\mathbf{K}\right)\right)\\ &= \det\left(s\mathbb{I} - \mathbf{A}\right)\det\left(\mathbb{I} + \mathbf{\Phi}(s)\mathbf{B}\mathbf{K}\right)\\ &= \det\left(s\mathbb{I} - \mathbf{A}\right)\det\left(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}\right)\end{aligned} \tag{3.138}$$

We finally get the following relation, which is known as the Hsu-Chen theorem[7]:

$$\boxed{\det\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\right) = \det\left(s\mathbb{I} - \mathbf{A}\right)\det\left(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}\right)} \tag{3.139}$$

The roots of $\det\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\right)$ are the eigenvalues of the closed-loop system. Consequently they are related to the stability of the closed-loop system.

Moreover the roots of $\det\left(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}\right)$ are exactly the roots of $\det\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\right)$. Indeed, as far as $\mathbf{\Phi}(s) = \left(s\mathbb{I} - \mathbf{A}\right)^{-1}$, the inverse of

[6] https://en.wikipedia.org/wiki/Determinant

[7] Pole-shifting techniques for multivariable feedback systems, Retallack D.G., MacFarlane A.G.J., Proceedings of the Institution of Electrical Engineers, 1970

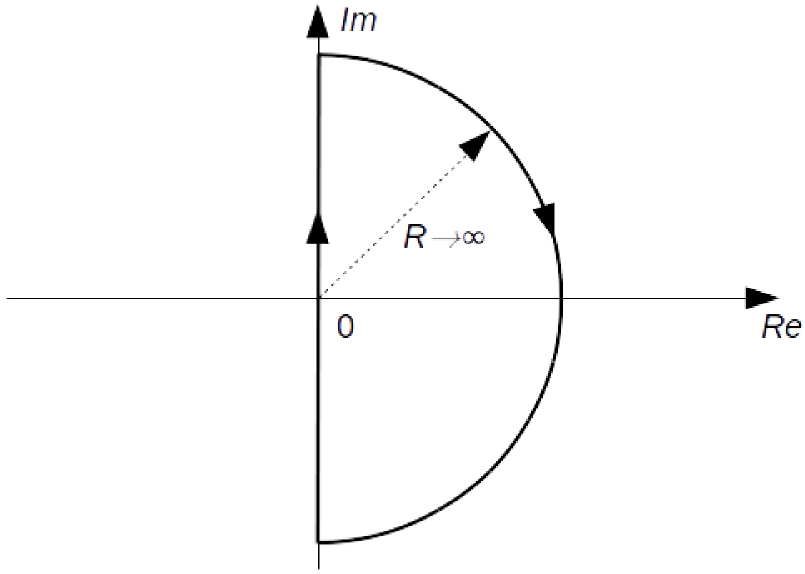


Figure 3.3: Nyquist contour

$(s\mathbb{I} - \mathbf{A})$ is computed as the adjugate of matrix $(s\mathbb{I} - \mathbf{A})$ divided by $\det(s\mathbb{I} - \mathbf{A})$ which finally becomes the denominator of $\det(\mathbb{I} + \mathbf{K\Phi}(s)\mathbf{B})$:

$$
\begin{array}{rl}
\det(\mathbb{I} + \mathbf{K\Phi}(s)\mathbf{B}) & = \det\left(\mathbb{I} + \mathbf{K}(s\mathbb{I} - \mathbf{A})^{-1}\mathbf{B}\right) \\
& = \det\left(\mathbb{I} + \mathbf{K}\frac{\mathrm{adj}(s\mathbb{I}-\mathbf{A})}{\det(s\mathbb{I}-\mathbf{A})}\mathbf{B}\right) \\
& = \det\left(\frac{\det(s\mathbb{I}-\mathbf{A})\mathbb{I}+\mathbf{K}\,\mathrm{adj}(s\mathbb{I}-\mathbf{A})\mathbf{B}}{\det(s\mathbb{I}-\mathbf{A})}\right) \\
& = \frac{\det(\det(s\mathbb{I}-\mathbf{A})\mathbb{I}+\mathbf{K}\,\mathrm{adj}(s\mathbb{I}-\mathbf{A})\mathbf{B})}{\det(s\mathbb{I}-\mathbf{A})} \\
\Rightarrow \det(s\mathbb{I} - \mathbf{A} + \mathbf{BK}) & = \det\left(\det(s\mathbb{I} - \mathbf{A})\,\mathbb{I} + \mathbf{K}\,\mathrm{adj}(s\mathbb{I} - \mathbf{A})\,\mathbf{B}\right)
\end{array}
\tag{3.140}
$$

Thus:

$$\det(s\mathbb{I} - \mathbf{A} + \mathbf{BK}) = 0 \Leftrightarrow \det(\mathbb{I} + \mathbf{K\Phi}(s)\mathbf{B}) = 0 \tag{3.141}$$

Consequently, the eigenvalues of full-state feedback loop are the roots of $\det(\mathbb{I} + \mathbf{K\Phi}(s)\mathbf{B})$.

3.9.2 Generalized (MIMO) Nyquist stability criterion

Let's recall the generalized (MIMO) Nyquist stability criterion which will be applied in the next section to the LQR design through Kalman equality.

We remind that the Nyquist plot of $\det(\mathbb{I}+\mathbf{K\Phi}(s)\mathbf{B})$ is the image of $\det(\mathbb{I}+\mathbf{K\Phi}(s)\mathbf{B})$ as s goes clockwise around the Nyquist contour: this includes the entire imaginary axis ($s = j\omega$) and an infinite semi-circle around the right half plane as shown in Figure 3.3.

The generalized (MIMO) Nyquist stability criterion states that the number of unstable closed-loop poles (that are the roots of $\det(s\mathbb{I}-\mathbf{A}+\mathbf{BK})$) is equal to the number of unstable open-loop poles (that are the roots of $\det(s\mathbb{I} - \mathbf{A})$) plus the number of encirclements of the critical point $(0,0)$ by the Nyquist plot of $\det(\mathbb{I}+\mathbf{K}\mathbf{\Phi}(s)\mathbf{B})$; the encirclement is counted positive in the clockwise direction and negative otherwise.

An easy way to determine the number of encirclements of the critical point is to draw a line out from the critical point, in any directions. Then by counting the number of times that the Nyquist plot crosses the line in the clockwise direction (i.e. left to right) and by subtracting the number of times it crosses in the counterclockwise direction then the number of clockwise encirclements of the critical point is obtained. A negative number indicates counterclockwise encirclements.

It is worth noticing that for Single-Input Single-Output (SISO) systems $\mathbf{K}$ is a row vector whereas $\mathbf{B}$ is a column vector. Consequently $\mathbf{K}\mathbf{\Phi}(s)\mathbf{B}$ is a scalar and we have:

$$\det(\mathbb{I}+\mathbf{K}\mathbf{\Phi}(s)\mathbf{B}) = \det(1+\mathbf{K}\mathbf{\Phi}(s)\mathbf{B}) = 1+\mathbf{K}\mathbf{\Phi}(s)\mathbf{B} \tag{3.142}$$

Thus for Single-Input Single-Output (SISO) systems the number of encirclements of the critical point $(0,0)$ by the Nyquist plot of $\det(\mathbb{I}+\mathbf{K}\mathbf{\Phi}(s)\mathbf{B})$ is equivalent to the number of encirclements of the critical point $(-1,0)$ by the Nyquist plot of $\mathbf{K}\mathbf{\Phi}(s)\mathbf{B}$.

In the context of output feedback the control $u(t) = -\mathbf{K}\underline{x}(t)$ is replaced by $u(t) = -\mathbf{K}y(t)$ where $y(t)$ is the output of the plant: $y(t) = \mathbf{C}\underline{x}(t)$. As a consequence the control $u(t)$ reads $u(t) = -\mathbf{K}\mathbf{C}\underline{x}(t)$ and state feedback gain $\mathbf{K}$ is replaced by output feedback gain $\mathbf{K}\mathbf{C}$ in equation (3.139):

$$\det(s\mathbb{I}-\mathbf{A}+\mathbf{B}\mathbf{K}\mathbf{C}) = \det(s\mathbb{I}-\mathbf{A})\det(\mathbb{I}+\mathbf{K}\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}) \tag{3.143}$$

This equation involves the transfer function $\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}$ between the output $Y(s)$ and the control $U(s)$ of the plant without any feedback and is used in the Nyquist stability criterion for Single-Input Single-Output (SISO) systems.

It is also worth noticing that $(\mathbb{I}+\mathbf{K}\mathbf{C}\mathbf{\Phi}(s)\mathbf{B})^{-1}$ is attached to the so called sensitivity function of the closed-loop whereas $\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}$ is attached to the open-loop transfer function from the process' input $\underline{U}(s)$ to the plant output $\underline{Y}(s)$.

3.9.3 Kalman equality

Let's consider the full-state feedback control is shown in Figure 3.2. Kalman has shown the following result, known as *Kalman equality*:

$$\boxed{(\mathbb{I}+\mathbf{L}(-s))^T\,\mathbf{R}\,(\mathbb{I}+\mathbf{L}(s)) = \mathbf{R}+(\mathbf{\Phi}(-s)\mathbf{B})^T\,\mathbf{Q}\,(\mathbf{\Phi}(s)\mathbf{B})} \tag{3.144}$$

where $\mathbf{L}(s)$ is the loop gain, $\mathbf{K}$ the optimal feedback gain (obtained through the algebraic Riccati equation) and $\mathbf{\Phi}(s)$ the resolvent of the state (transition) matrix $\mathbf{A}$:

$$\boxed{\left\{\begin{array}{l} \mathbf{L}(s) = \mathbf{K}\mathbf{\Phi}(s)\mathbf{B} \\ \mathbf{\Phi}(s) = (s\mathbb{I}-\mathbf{A})^{-1} \end{array}\right.} \tag{3.145}$$

The proof of the Kalman equality is provided hereafter. Consider the *algebraic Riccati equation* (3.8):

$$\mathbf{P}\mathbf{A} + \mathbf{A}^T\mathbf{P} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \tag{3.146}$$

Because $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$, $\mathbf{P} = \mathbf{P}^T$ and $\mathbf{R} = \mathbf{R}^T$, the previous equation can be re-written as:

$$\mathbf{P}\left(s\mathbb{I} - \mathbf{A}\right) - \left(-s\mathbb{I} - \mathbf{A}\right)^T\mathbf{P} + \mathbf{K}^T\mathbf{R}\mathbf{K} = \mathbf{Q} \tag{3.147}$$

Using the fact that $\mathbf{\Phi}(s) = (s\mathbb{I} - \mathbf{A})^{-1}$ we get:

$$\mathbf{P}\mathbf{\Phi}^{-1}(s) + \left(\mathbf{\Phi}^{-1}(-s)\right)^T\mathbf{P} + \mathbf{K}^T\mathbf{R}\mathbf{K} = \mathbf{Q} \tag{3.148}$$

Left multiplying by $\mathbf{B}^T\mathbf{\Phi}^T(-s)$ and right multiplying by $\mathbf{\Phi}(s)\mathbf{B}$ yields:

$$\begin{aligned} &\mathbf{B}^T\mathbf{\Phi}^T(-s)\mathbf{P}\mathbf{B} + \mathbf{B}^T\mathbf{P}\mathbf{\Phi}(s)\mathbf{B} + \mathbf{B}^T\mathbf{\Phi}^T(-s)\mathbf{K}^T\mathbf{R}\mathbf{K}\mathbf{\Phi}(s)\mathbf{B} = \\ &\qquad \mathbf{B}^T\mathbf{\Phi}^T(-s)\mathbf{Q}\mathbf{\Phi}(s)\mathbf{B} \end{aligned} \tag{3.149}$$

Adding $\mathbf{R}$ to both sides of equation (3.149) as using the fact that $\mathbf{R}\mathbf{K} = \mathbf{B}^T\mathbf{P}$ we get:

$$\begin{aligned} &\mathbf{R} + \mathbf{B}^T\mathbf{\Phi}^T(-s)\mathbf{K}^T\mathbf{R} + \mathbf{R}\mathbf{K}\mathbf{\Phi}(s)\mathbf{B} + \mathbf{B}^T\mathbf{\Phi}^T(-s)\mathbf{K}^T\mathbf{R}\mathbf{K}\mathbf{\Phi}(s)\mathbf{B} = \\ &\qquad \mathbf{R} + \mathbf{B}^T\mathbf{\Phi}^T(-s)\mathbf{Q}\mathbf{\Phi}(s)\mathbf{B} \end{aligned} \tag{3.150}$$

The previous equation can be re-written as:

$$\left(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(-s)\mathbf{B}\right)^T\mathbf{R}\left(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}\right) = \mathbf{R} + \left(\mathbf{\Phi}(-s)\mathbf{B}\right)^T\mathbf{Q}\left(\mathbf{\Phi}(s)\mathbf{B}\right) \tag{3.151}$$

This completes the proof. ■

Let $\mathbf{R}^{-0.5}$ be the root-square of matrix $\mathbf{R}^{-1}$:

$$\mathbf{R}^{-1} = \left(\mathbf{R}^{-0.5}\right)^T\mathbf{R}^{-0.5} \tag{3.152}$$

Multiplying Kalman equality (3.144) by $\left(\mathbf{R}^{-0.5}\right)^T$ on the left side and by $\mathbf{R}^{-0.5}$ on the right side we get:

$$\begin{aligned} &\left(\mathbf{R}^{-0.5}\right)^T\left(\mathbb{I} + \mathbf{L}(-s)\right)^T\mathbf{R}\left(\mathbb{I} + \mathbf{L}(s)\right)\mathbf{R}^{-0.5} \\ &\qquad = \left(\mathbf{R}^{-0.5}\right)^T\left(\mathbf{R} + \left(\mathbf{\Phi}(-s)\mathbf{B}\right)^T\mathbf{Q}\left(\mathbf{\Phi}(s)\mathbf{B}\right)\right)\mathbf{R}^{-0.5} \end{aligned} \tag{3.153}$$

Matrix $\mathbf{R}^{0.5} = \left(\mathbf{R}^{0.5}\right)^T$ is the root square of matrix $\mathbf{R}$. By getting the modal decomposition of matrix $\mathbf{R}$, that is $\mathbf{R} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$ where $\mathbf{V}$ is the matrix whose columns are the eigenvectors of $\mathbf{R}$ and $\mathbf{D}$ is the diagonal matrix whose diagonal elements are the corresponding positive eigenvalues, the square root $\mathbf{R}^{0.5}$ of $\mathbf{R}$ is given by $\mathbf{R}^{0.5} = \mathbf{V}\mathbf{D}^{0.5}\mathbf{V}^{-1}$, where $\mathbf{D}^{0.5}$ is any diagonal matrix whose elements are the square root of the diagonal elements of $\mathbf{D}$. Thus we get:

$$\begin{aligned} \mathbf{R} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1} &\Rightarrow \mathbf{R}^{-1} = \mathbf{V}\mathbf{D}^{-1}\mathbf{V}^{-1} \\ \Rightarrow \left(\mathbf{R}^{-0.5}\right)^T\mathbf{R}\mathbf{R}^{-0.5} &= \mathbf{R}^{-0.5}\mathbf{R}\mathbf{R}^{-0.5} \\ &= \left(\mathbf{V}\mathbf{D}^{-0.5}\mathbf{V}^{-1}\right)\left(\mathbf{V}\mathbf{D}\mathbf{V}^{-1}\right)\left(\mathbf{V}\mathbf{D}^{-0.5}\mathbf{V}^{-1}\right) \\ &= \mathbf{V}\mathbf{D}^{-0.5}\mathbf{D}\mathbf{D}^{-0.5}\mathbf{V}^{-1} \\ &= \mathbb{I} \end{aligned} \tag{3.154}$$

Thus:

$$\begin{aligned}&\left(\mathbf{R}^{-0.5}\right)^T \left(\mathbf{R} + \left(\mathbf{\Phi}(-s)\mathbf{B}\right)^T \mathbf{Q}\left(\mathbf{\Phi}(s)\mathbf{B}\right)\right)\mathbf{R}^{-0.5}\\ &\qquad = \mathbb{I} + \left(\mathbf{\Phi}(-s)\mathbf{B}\mathbf{R}^{-0.5}\right)^T \mathbf{Q}\left(\mathbf{\Phi}(s)\mathbf{B}\mathbf{R}^{-0.5}\right)\end{aligned} \tag{3.155}$$

On the other hand, we have:

$$\begin{array}{l}\left(\mathbf{R}^{-0.5}\right)^T\left(\mathbb{I}+\mathbf{L}(-s)\right)^T\mathbf{R}\left(\mathbb{I}+\mathbf{L}(s)\right)\mathbf{R}^{-0.5}\\ \quad = \left(\mathbf{R}^{-0.5}+\mathbf{L}(-s)\mathbf{R}^{-0.5}\right)^T\mathbf{R}\left(\mathbf{R}^{-0.5}+\mathbf{L}(s)\mathbf{R}^{-0.5}\right)\\ \quad = \left(\mathbf{R}^{-0.5}+\mathbf{L}(-s)\mathbf{R}^{-0.5}\right)^T\mathbf{R}^{0.5}\mathbf{R}^{0.5}\left(\mathbf{R}^{-0.5}+\mathbf{L}(s)\mathbf{R}^{-0.5}\right)\\ \quad = \left(\mathbb{I}+\mathbf{R}^{0.5}\mathbf{L}(-s)\mathbf{R}^{-0.5}\right)^T\left(\mathbb{I}+\mathbf{R}^{0.5}\mathbf{L}(s)\mathbf{R}^{-0.5}\right)\end{array} \tag{3.156}$$

Finally, let $\mathbf{Q} := \mathbf{N}^T\mathbf{N}$. Then Kalman equality (3.144) can equivalently be written as follows:

$$\begin{aligned}&\left(\mathbb{I}+\mathbf{R}^{0.5}\mathbf{L}(-s)\mathbf{R}^{-0.5}\right)^T\left(\mathbb{I}+\mathbf{R}^{0.5}\mathbf{L}(s)\mathbf{R}^{-0.5}\right)\\ &\qquad = \mathbb{I} + \left(\mathbf{N}\mathbf{\Phi}(-s)\mathbf{B}\mathbf{R}^{-0.5}\right)^T\left(\mathbf{N}\mathbf{\Phi}(s)\mathbf{B}\mathbf{R}^{-0.5}\right)\end{aligned} \tag{3.157}$$

3.9.4 Robustness of Linear Quadratic Regulator

The robustness of the LQR design can be assessed through the Kalman equality (3.144). We will specialize Kalman equality to the specific case where the plant is a Single Input - Single Output (SISO) system. Then $\mathbf{K}\mathbf{\Phi}(s)\mathbf{B}$ and $\mathbf{R}$ are scalars. Setting $\mathbf{Q} = \mathbf{N}^T\mathbf{N}$, and using the fact that $\mathbf{N}\mathbf{\Phi}(s)\mathbf{B}$ is scalar for SISO plants, Kalman equality (3.144) reduces as follows:

$$\mathbf{Q} = \mathbf{N}^T\mathbf{N} \Rightarrow \left(1+\mathbf{L}(-s)\right)\left(1+\mathbf{L}(s)\right) = 1 + \tfrac{1}{\mathbf{R}}\left(\mathbf{N}\mathbf{\Phi}(-s)\mathbf{B}\right)\left(\mathbf{N}\mathbf{\Phi}(s)\mathbf{B}\right) \tag{3.158}$$

Transfer function $\mathbf{L}(s)$ is the loop gain, which is scalar for SISO plants:

$$\mathbf{L}(s) = \mathbf{K}\mathbf{\Phi}(s)\mathbf{B} \tag{3.159}$$

Substituting $s = j\omega$ yields:

$$\|1+\mathbf{L}(j\omega)\|^2 = 1 + \frac{1}{\mathbf{R}}\|\mathbf{N}\mathbf{\Phi}(j\omega)\mathbf{B}\|^2 \tag{3.160}$$

Therefore:

$$\|1+\mathbf{L}(j\omega)\| \geq 1 \quad \forall\omega\in\mathbb{R} \tag{3.161}$$

For SISO plants, the sensitivity function $\mathbf{S}(s)$ and the complementary sensitivity function $\mathbf{T}(s)$ are defined as follows:

$$\left\{\begin{array}{l}\mathbf{S}(s) = \frac{1}{1+\mathbf{L}(s)}\\ \mathbf{T}(s) = 1 - \mathbf{S}(s) = \frac{\mathbf{L}(s)}{1+\mathbf{L}(s)}\end{array}\right. \tag{3.162}$$

Substituting $s = j\omega$, Kalman's inequality guarantees that:

$$\left\{\begin{array}{l}\|\mathbf{S}(j\omega)\| \leq 1\\ \|\mathbf{T}(j\omega)\| \leq 2\end{array}\right. \tag{3.163}$$

Those inequalities are represented in Figure 3.4.

We recall that:

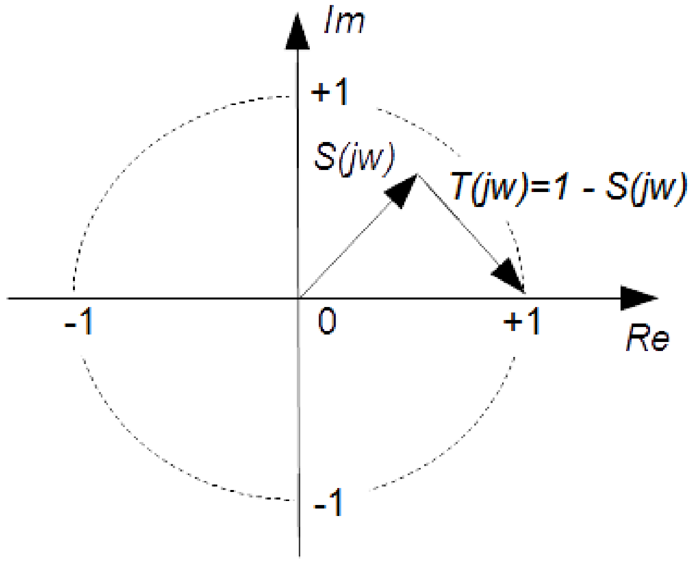


Figure 3.4: Upper bounds of sensitivity function $\mathbf{S}(s)$ and complementary sensitivity function $\mathbf{T}(s)$ through LQR design

- A small sensitivity function is desirable for good disturbance rejection. Generally, this is especially important at low frequencies.

- A complementary sensitivity function close to one is desirable for good reference tracking. Generally, this is especially important at low frequencies.

- A small complementary sensitivity function is desirable for good noise rejection. Generally, this is especially important at high frequencies.

Furthermore, let's introduce the real part $X(\omega)$ and the imaginary part $Y(\omega)$ of $\mathbf{L}(j\omega)$:

$$\mathbf{L}(j\omega) = X(\omega) + jY(\omega) \tag{3.164}$$

Then $\|1 + \mathbf{L}(j\omega)\|^2$ reads as follows:

$$\|1 + \mathbf{L}(j\omega)\|^2 = \|1 + X(\omega) + jY(\omega)\|^2 = (1 + X(\omega))^2 + Y(\omega)^2 \tag{3.165}$$

Consequently inequality (3.161) reads as follows:

$$\begin{array}{l} \|1 + \mathbf{L}(j\omega)\| \geq 1 \\ \Leftrightarrow \|1 + \mathbf{L}(j\omega)\|^2 \geq 1 \\ \Leftrightarrow (1 + X(\omega))^2 + Y(\omega)^2 \geq 1 \end{array} \tag{3.166}$$

As a consequence, the Nyquist plot of $\mathbf{L}(j\omega)$ will be outside the circle of unit radius centered at $(-1, 0)$. Thus applying the generalized (MIMO) Nyquist stability criterion and knowing that the LQR design always leads to a stable closed-loop plant, the implications of Kalman inequality are the following:

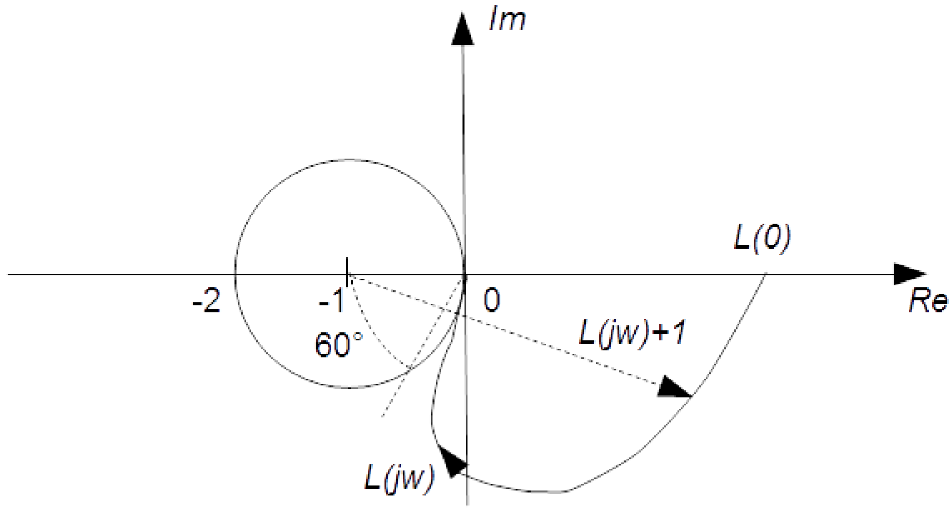


Figure 3.5: Nyquist plot of $\mathbf{L}(s)$: example where the open-loop system has no unstable pole

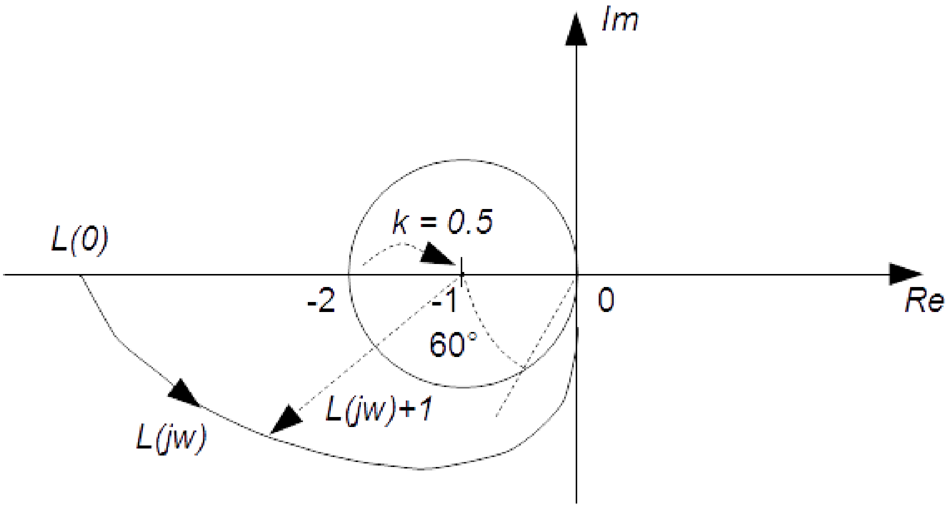


Figure 3.6: Nyquist plot of $\mathbf{L}(s)$: example where the open-loop system has unstable poles

– If the open-loop system has no unstable pole, then the Nyquist plot of $\mathbf{L}(j\omega)$ does not encircle the critical point $(-1, 0)$. This corresponds to a positive gain margin of $+\infty$ as depicted as depicted in Figure 3.5.

– On the other hand if $\mathbf{\Phi}(s)$ has unstable poles, the Nyquist plot of $\mathbf{L}(j\omega)$ encircles the critical point $(-1, 0)$ a number on times which corresponds to the number of unstable open-loop poles. This corresponds to a negative gain margin which is always lower or equal to $20 \log_{10}(0.5) = -6\ dB$ as depicted in Figure 3.6.

In both situations, if the process' phase increases by 60 degrees its Nyquist plots rotates by 60 degrees but the number of encirclements still does not change. Thus the LQR design always leads to a phase margin which is always greater or equal to 60 degrees.

Last but not least, it can be seen in Figure 3.5 and Figure 3.6 that at high-frequency the loop gain $\mathbf{L}(j\omega)$ can have at most -90 degrees phase for high-frequencies and therefore the *roll-off* rate is at most -20 dB/decade.

Unfortunately those nice properties are lost as soon as the performance index $J(\underline{u}(t))$ contains state / control cross-terms [8]:

$$J(\underline{u}(t)) = \frac{1}{2} \int_0^{t_f} \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t) + 2\underline{x}^T(t)\mathbf{S}\underline{u}(t)\, dt \tag{3.167}$$

This is especially the case for LQG (Linear Quadratic Gaussian) regulator where the plant dynamics as well as the output measurement are subject to stochastic disturbances and where a state estimator has to be used.

[8]Doyle J.C., Guaranteed margins for LQG regulators, IEEE Transactions on Automatic Control, Volume: 23, Issue: 4, Aug 1978

Chapter 4

Design methods

4.1 Symmetric Root Locus

4.1.1 Characteristics polynomials

Let's consider the following state space realization $(\mathbf{A}, \mathbf{B}, \mathbf{N})$:

$$\begin{cases} \underline{\dot{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}u(t) \\ \underline{z}(t) = \mathbf{N}\,\underline{x}(t) \end{cases} \tag{4.1}$$

We will assume that $(\mathbf{A}, \mathbf{B}, \mathbf{N})$ is minimal, or equivalently that $(\mathbf{A}, \mathbf{B})$ is controllable and $(\mathbf{A}, \mathbf{N})$ is observable, or equivalently that the following loop gain (or open-loop) transfer function is irreducible:

$$G(s) = \mathbf{N}\left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B} = \frac{\mathbf{N}\,\mathrm{adj}\left(s\mathbb{I} - \mathbf{A}\right)\mathbf{B}}{\det\left(s\mathbb{I} - \mathbf{A}\right)} = \frac{N(s)}{D(s)} \tag{4.2}$$

The polynomial $D(s) = \det\left(s\mathbb{I} - \mathbf{A}\right)$ is the loop gain characteristics polynomial, which is assumed to be of degree n, and polynomial matrix $N(s)$ is the numerator of $\mathbf{N}\left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B}$. From the fact that the numerator of $G(s)$ involves $\mathrm{adj}\left(s\mathbb{I} - \mathbf{A}\right)$ it is clear that the degree of its numerator $N(s)$, which will be denoted m, is strictly lower than the degree of its denominator $D(s)$, which will be denoted n:

$$deg(N(s)) = m < deg(D(s)) = n \tag{4.3}$$

It can be shown that for single-input single-ouput (SISO) systems we have the following relation where $N(s)$ is the polynomial (not matrix) numerator of the transfer function:

$$G(s) = \mathbf{N}\left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B} = \frac{\det\left(\begin{bmatrix} s\mathbb{I} - \mathbf{A} & -\mathbf{B} \\ \mathbf{N} & 0 \end{bmatrix}\right)}{\det\left(s\mathbb{I} - \mathbf{A}\right)} = \frac{N(s)}{D(s)} \tag{4.4}$$

Now let's assume that the system is closed thanks to the following output (not state !) feedback control $u(t)$:

$$u(t) = -k_p\mathbf{K}_o\underline{z}(t) + \mathbf{F}r(t) \tag{4.5}$$

Where:

- k_p is a scaling factor
- $\mathbf{K}_o$ is the output (not state !) feedback matrix gain
- $\mathbf{F}$ is the pre-filter gain

Then the state matrix of the closed-loop system reads $\mathbf{A} - k_p\mathbf{B}\mathbf{K}_o\mathbf{N}$ and the polynomial $\det(s\mathbb{I} - \mathbf{A} + k_p\mathbf{B}\mathbf{K}_o\mathbf{N})$ is the closed-loop characteristics polynomial.

4.1.2 Root Locus reminder

The root locus technique[1] has been developed in 1948 by Walter R. Evans (1920-1999). This is a graphical method for sketching in the s-plane the locus of roots of the following polynomial when parameter k_p varies to 0 to infinity:

$$\det(s\mathbb{I} - \mathbf{A} + k_p\mathbf{B}\mathbf{K}_o\mathbf{N}) = D(s) + k_pN(s) \tag{4.6}$$

Usually polynomial $D(s) + k_pN(s)$ represents the denominator of a closed-loop transfer function. Polynomial $D(s) + k_pN(s)$ represents here the denominator of the closed-loop transfer function when control $u(t)$ reads:

$$u(t) = -k_p\mathbf{K}_oy(t) + \mathbf{F}r(t) \tag{4.7}$$

It is worth noticing that the roots of $D(s) + k_pN(s)$ are also the roots of $1 + k_p\frac{N(s)}{D(s)}$:

$$D(s) + k_pN(s) = 0 \Leftrightarrow 1 + k_p\frac{N(s)}{D(s)} = 0 \Leftrightarrow L(s) := k_pF(s) = -1 \tag{4.8}$$

Without loss of generality let's define transfer function $F(s)$ as follows:

$$F(s) = \frac{N(s)}{D(s)} = a\frac{\prod_{j=1}^{m\leq n}(s - z_j)}{\prod_{i=1}^{n}(s - p_i)} \tag{4.9}$$

Transfer function $L(s) = k_pF(s)$ is called the loop transfer function. In the SISO case the numerator of the loop transfer function $L(s)$ is scalar as well as its denominator.

Equation $L(s) = -1$ can be equivalently split into two equations:

$$\begin{cases} |L(s)| = 1 \\ \arg(L(s)) = (2k+1)\pi, \quad k = 0, \pm 1, \cdots \end{cases} \tag{4.10}$$

The magnitude condition can always be satisfied by a suitable choice of k_p. On the other hand the phase condition does not depend on the value of k_p but only on the sign of k_p. Thus we have to find all the points in the s-plane that satisfy the phase condition. When scalar gain k_p varies from zero to infinity (i.e. k_p is positive), the root locus technique is based on the following rules:

[1] Walter R. Evans , Graphical Analysis of Control Systems, Transactions of the American Institute of Electrical Engineers, vol. 67, pp. 547 - 551, 1948

- The root locus is symmetrical with respect to the horizontal real axis (because roots are either real or complex conjugate);
- The number of branches is equal to the number of poles of the loop transfer function. Thus the root locus has n branches;
- The root locus starts at the n poles of the loop transfer function;
- The root locus ends at the zeros of the loop transfer function. Thus m branches of the root locus end on the m zeros of $F(s)$ and there are $(n-m)$ asymptotic branches;
- Assuming that coefficient a in $F(s)$ is positive, a point s^* on the real axis belongs to the root locus as soon as there is an odd number of poles and zeros on its right. Conversely assuming that coefficient a in $F(s)$ is negative, a point s^* on the real axis belongs to the root locus as soon as there is an even number of poles and zeros on its right. Be careful to take into account the multiplicity of poles and zeros in the counting process;
- The $(n - m)$ asymptotic branches of the root locus which diverge to ∞ are asymptotes.
  - The angle δ_k of each asymptote with the real axis is defined by:

    $$\delta_k = \frac{\pi + \arg(a) + 2k\pi}{n - m} \quad \forall\, k = 0, \dots, n - m - 1 \tag{4.11}$$

  - Denoting by p_i the n poles of the loop transfer function (that are the roots of $D(s)$) and by z_j the m zeros of the loop transfer function (that are the roots of $N(s)$), the asymptotes intersect the real axis at a point (called pivot or centroid) given by:

    $$\sigma = \frac{\sum_{i=1}^{n} p_i - \sum_{j=1}^{m \le n} z_j}{n - m} \tag{4.12}$$

- The breakaway / break-in points are located on the real axis and always have a vertical tangent. They are located at the roots s_b of the following equation as soon as there is an odd (if coefficient a in $F(s)$ is positive) or even (if coefficient a in $F(s)$ is negative) number of poles and zeros on its right (Be careful to take into account the multiplicity of poles and zeros in the counting process):

  $$\begin{array}{l} \frac{d}{ds}\left(\frac{1}{F(s)}\right)_{s=s_b} = \frac{d}{ds}\left(\frac{D(s)}{N(s)}\right)_{s=s_b} = 0 \\ \Leftrightarrow D'(s_b)N(s_b) - D(s_b)N'(s_b) = 0 \end{array} \tag{4.13}$$

  Indeed from the fact that breakaway / break-in points have always a vertical tangent we can write:

  $$1 + k_p F(s) = 1 + k_p \frac{N(s)}{D(s)} = 0 \Rightarrow \frac{dk_p}{dp} = -\frac{D'(s)N(s) - D(s)N'(s)}{N^2(s)} = 0 \tag{4.14}$$

  From this relation we get (4.13).

- On the imaginary axis we have $s = j\omega$. Thus the value of the (positive) critical gain beyond which the closed-loop system becomes unstable is the value of k_p ($k_p \geq 0$) such that the root locus of $F(s)$ crosses the imaginary axis. In that situation at least one pole of the closed-loop system is purely imaginary. As far as $D(s) + k_p N(s)$ represents the denominator of the closed-loop transfer function the critical gain can be obtained by replacing s by $j\omega$ and by solving:

$$1 + k_p F(j\omega) = 0 \Leftrightarrow D(j\omega) + k_p N(j\omega) = 0 \tag{4.15}$$

  The previous equation is then split into its real and imaginary part and provides a system of 2 equations which lead to the value of the critical gain and the oscillation frequency at the critical gain. It is worth noticing that the Routh criterion can be used for the same purpose.

- Note that if the degree of polynomial $D(s)$ is greater than or equal to the degree of polynomial $N(s)$ plus 2, meaning that the relative degree of transfer function $F(s)$ is greater than or equal to 2 ($n - m \geq 2$), then the sum of the poles of the feedback system is independent of the value of parameter k_p, and therefore is equal to the sum of the poles of the open loop system when $k_p = 0$. This property is known as the *centroid theorem*. To get this result, we have simply to expand $D(s) + k_p N(s)$ taking into account $n - m \geq 2$:

$$\begin{array}{rl} F(s) = \frac{N(s)}{D(s)} & = a \frac{\prod_{j=1}^{m \leq n-2}(s - z_j)}{\prod_{i=1}^{n}(s - p_i)} \\ \Rightarrow D(s) + k_p N(s) & = \prod_{i=1}^{n}(s - p_i) + k_p\, a \prod_{j=1}^{m \leq n-2}(s - z_j) \\ & = s^n - (r_1 + r_2 + \cdots + r_n)\, s^{n-1} + \cdots \end{array} \tag{4.16}$$

  Assuming that $n - m \geq 2$, the coefficient of the term s^{n-1} in polynomial $D(s) + k_p N(s)$ does not depend on parameter k_p. Because this coefficient is obtained has the opposite of the sum $r_1 + r_2 + \cdots + r_n$ of the roots of polynomial $D(s) + k_p N(s)$, we conclude the sum of the poles of the feedback system is independent of the value of parameter k_p.

4.1.3 Chang-Letov design procedure

The purpose of this section is to have some insight on how to drive the modes of the closed-loop plant thanks to the LQR design applied to SISO plants. More precisely, we focus on single-input plants for which the cost to be minimized is defined as in (3.3):

$$J(u(t)) = \frac{1}{2} \int_0^\infty \left(\underline{x}^T(t) \mathbf{Q} \underline{x}(t) + u^T(t) \mathbf{R} u(t) \right) dt \tag{4.17}$$

Nevertheless, weight matrix $\mathbf{Q}$ is here defined as follows, where matrix $\mathbf{N}$ is a design matrix:

$$\mathbf{Q} = \mathbf{N}^T \mathbf{N} \tag{4.18}$$

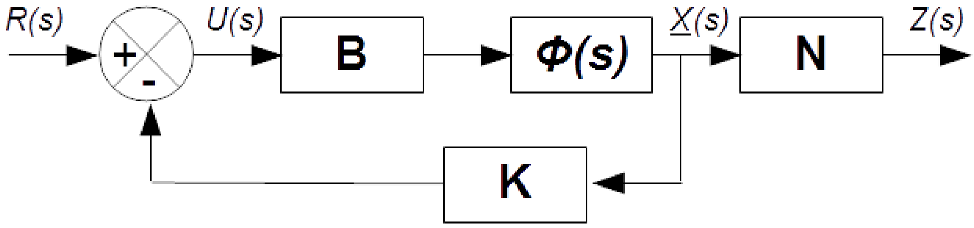


Figure 4.1: Full-state feedback control with *fictitious* output z

Let $\underline{z}(t) := \mathbf{N}\,\underline{x}(t)$ be the *controlled* output: this is a *fictitious* output which represents the output of interest for the design. The *controlled* output $\underline{z}(t)$ is expressed as a linear function of the state vector $\underline{x}(t)$ as:

$$\underline{z}(t) := \mathbf{N}\,\underline{x}(t) \tag{4.19}$$

Thus the cost to be minimized can be rewritten as follows:

$$\begin{array}{rl} J(u(t)) & = \frac{1}{2}\int_0^\infty \left(\underline{x}^T(t)\mathbf{N}^T\mathbf{N}\underline{x}(t) + \mathbf{R}u^2(t)\right)\,dt \\ & = \frac{1}{2}\int_0^\infty \left(\underline{z}^T(t)\underline{z}(t) + \mathbf{R}u^2(t)\right)\,dt \end{array} \tag{4.20}$$

Furthermore the cost to be minimized is now *constrained* by the dynamics of the system with the following state space representation:

$$\left\{ \begin{array}{l} \underline{\dot{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}u(t) \\ \underline{z}(t) = \mathbf{N}\underline{x}(t) \end{array} \right. \tag{4.21}$$

From this state space representation we obtain the following open-loop transfer function which is written as the ratio between a numerator $N(s)$ and a denominator $D(s)$:

$$\mathbf{N}\left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B} = \frac{N(s)}{D(s)} \tag{4.22}$$

We recall that the cost (4.20) is minimized by choosing the following control law, where $\mathbf{P}$ is the solution of the algebraic Riccati equation:

$$\left\{ \begin{array}{l} u(t) = -\mathbf{K}\underline{x}(t) \\ \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \end{array} \right. \tag{4.23}$$

This leads to a full-state feedback control with *fictitious* output z which is represented in Figure 4.1 where $\mathbf{\Phi}(s) = (s\mathbb{I} - \mathbf{A})^{-1}$.

Let $D(s)$ be the open-loop characteristics polynomial and $\beta(s)$ be the closed-loop characteristic polynomial:

$$\left\{ \begin{array}{l} D(s) = \det\left(s\mathbb{I} - \mathbf{A}\right) \\ \beta(s) = \det\left(s\mathbb{I} - \mathbf{A} + \mathbf{BK}\right) \end{array} \right. \tag{4.24}$$

In the single control case which is under consideration, it can be shown (see section 4.1.4) that the characteristic polynomial of the closed-loop system is linked with the numerator and the denominator of the loop transfer function as follows:

$$\beta(s)\,\beta(-s) = D(s)D(-s) + \frac{1}{\mathbf{R}}N(s)N(-s) \tag{4.25}$$

This relation can be associated with the root locus of $G(s)G(-s) = \frac{N(s)N(-s)}{D(s)D(-s)}$ where fictitious gain $k_p = \frac{1}{\mathbf{R}}$ varies from 0 to ∞. This leads to the so-called Chang-Letov design procedure, which enables to find the closed-loop poles based on the open-loop poles and zeros of $G(s)G(-s)$. The difference with the root locus of $G(s)$ is that both the open-loop poles and zeros and their reflections about the imaginary axis have to be taken into account (this is due to the multiplication by $G(-s)$). The actual closed-loop poles are those located in the left half plane with negative real part; indeed optimal control leads always to a stabilizing gain. It is worth noticing that matrix $\mathbf{N}$ is actually a design parameter which is used to shape the root locus.

4.1.4 Proof of the symmetric root locus result

The proof of (4.25) can be done as follows: taking the determinant of the Kalman equality (3.144) and having in mind that $\det(\mathbf{M}^T) = \det(\mathbf{M})$ and that for SISO systems $\mathbf{R}$ is scalar yields:

$$\begin{array}{l}
\det\left((\mathbb{I} + \mathbf{K}\mathbf{\Phi}(-s)\mathbf{B})^T\,\mathbf{R}\,(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B})\right) = \det\left(\mathbf{R} + (\mathbf{\Phi}(-s)\mathbf{B})^T\,\mathbf{Q}\,(\mathbf{\Phi}(s)\mathbf{B})\right) \\
\Leftrightarrow \det\left((\mathbb{I} + \mathbf{K}\mathbf{\Phi}(-s)\mathbf{B})^T\,(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B})\right) = \det\left(\mathbb{I} + \frac{(\mathbf{\Phi}(-s)\mathbf{B})^T\mathbf{Q}(\mathbf{\Phi}(s)\mathbf{B})}{\mathbf{R}}\right) \\
\Leftrightarrow \det\left((\mathbb{I} + \mathbf{K}\mathbf{\Phi}(-s)\mathbf{B})^T\right)\det(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}) = \det\left(\mathbb{I} + \frac{(\mathbf{\Phi}(-s)\mathbf{B})^T\mathbf{Q}(\mathbf{\Phi}(s)\mathbf{B})}{\mathbf{R}}\right) \\
\Leftrightarrow \det(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(-s)\mathbf{B})\det(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}) = \det\left(\mathbb{I} + \frac{(\mathbf{\Phi}(-s)\mathbf{B})^T\mathbf{Q}(\mathbf{\Phi}(s)\mathbf{B})}{\mathbf{R}}\right)
\end{array} \tag{4.26}$$

Where:

$$\mathbf{\Phi}(s) = (s\mathbb{I} - \mathbf{A})^{-1} = \frac{\operatorname{adj}(s\mathbb{I} - \mathbf{A})}{\det(s\mathbb{I} - \mathbf{A})} \tag{4.27}$$

Furthermore it has been seen in (3.139) that thanks to the Hsu-Chen theorem we have:

$$\det(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}) = \frac{\det(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K})}{\det(s\mathbb{I} - \mathbf{A})} \tag{4.28}$$

Let $D(s)$ be the open-loop characteristics polynomial and $\beta(s)$ be the closed-loop characteristic polynomial:

$$\left\{\begin{array}{l} D(s) = \det(s\mathbb{I} - \mathbf{A}) \\ \beta(s) = \det(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}) \end{array}\right. \tag{4.29}$$

As a consequence, using (4.28) in the left part of (4.26) yields:

$$\frac{\beta(s)\,\beta(-s)}{D(s)D(-s)} = \det\left(\mathbb{I} + \frac{(\mathbf{\Phi}(-s)\mathbf{B})^T\,\mathbf{Q}\,(\mathbf{\Phi}(s)\mathbf{B})}{\mathbf{R}}\right) \tag{4.30}$$

In the single control case $\mathbf{R}$ and $\mathbb{I}$ are scalars ($\mathbb{I} = 1$). Using $\mathbf{Q} = \mathbf{N}^T\mathbf{N}$ (4.30) becomes:

$$\begin{aligned} \frac{\beta(s)\,\beta(-s)}{D(s)D(-s)} &= \det\left(1 + \frac{(\mathbf{N\Phi}(-s)\mathbf{B})^T(\mathbf{N\Phi}(s)\mathbf{B})}{\mathbf{R}}\right) \\ &= 1 + \frac{(\mathbf{N\Phi}(-s)\mathbf{B})^T(\mathbf{N\Phi}(s)\mathbf{B})}{\mathbf{R}} \end{aligned} \tag{4.31}$$

We recognize in $\mathbf{N\Phi}(s)\mathbf{B} = \mathbf{N}\left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B}$ the open-loop transfer function $G(s)$ which is the ratio between numerator polynomial $N(s)$ and denominator polynomial $D(s)$:

$$G(s) = \mathbf{N\Phi}(s)\mathbf{B} = \mathbf{N}\left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B} = \frac{N(s)}{D(s)} \tag{4.32}$$

Using (4.32) in (4.31) yields:

$$\begin{aligned} &\frac{\beta(s)\,\beta(-s)}{D(s)D(-s)} = 1 + \frac{1}{\mathbf{R}}\frac{N(s)N(-s)}{D(s)D(-s)} \\ &\Leftrightarrow \beta(s)\,\beta(-s) = D(s)D(-s) + \frac{1}{\mathbf{R}}N(s)N(-s) \end{aligned} \tag{4.33}$$

This completes the proof. ■

4.2 Asymptotic properties of LQR applied to SISO plants

We will see that Kalman equality allows for loop shaping through LQR design for SISO plants. Lectures from professor Faryar Jabbari (Henry Samueli School of Engineering, University of California) and professor Perry Y. Li (University of Minnesota) are the primary sources of this section.

We recall that $\mathbf{\Phi}(s) = (s\mathbb{I} - \mathbf{A})^{-1}$ where $dim(\mathbf{A}) = n \times n$, which is also the dimension of weight $\mathbf{Q} := \mathbf{N}^T\mathbf{N}$.

4.2.1 Closed-loop poles location

Relation (4.25) reads:

$$\beta(s)\,\beta(-s) = D(s)D(-s) + \frac{1}{\mathbf{R}}N(s)N(-s) \tag{4.34}$$

From (4.34) we can get the following results:

– When $\mathbf{R}$ is large, i.e. $1/\mathbf{R}$ is small so that the control energy is weighted very heavily in the performance index, the roots of $\beta(s)$, that are the closed-loop poles, approach the stable open-loop poles or the negative of the unstable open-loop poles:

$$\beta(s)\,\beta(-s) \approx D(s)D(-s) \text{ as } \mathbf{R} \to \infty \tag{4.35}$$

– When $\mathbf{R}$ is small (i.e. $\mathbf{R} \to 0$) then $1/\mathbf{R}$ is large and the control is cheap. Then the roots of $\beta(s)$, that are the closed-loop poles, approach the stable open-loop zeros or the negative of the non-minimum phase open-loop zeros:

$$\beta(s)\,\beta(-s) \approx \frac{1}{\mathbf{R}} N(s)N(-s) \text{ as } \mathbf{R} \to 0 \tag{4.36}$$

Equation (4.36) shows that any roots of $\beta(s)\,\beta(-s)$ that remains finite as $\mathbf{R} \to 0$ must tend toward the roots of $N(s)N(-s)$. But from (4.3) we know that the degree of $N(s)N(-s)$, say $2m$, is less than the degree of $\beta(s)\,\beta(-s)$, which is $2n$. Therefore m roots of $\beta(s)$ are the roots of $N(s)N(-s)$ in the open left half plane (*stable* roots). The remaining $n-m$ roots of $\beta(s)$ asymptotically approach infinity in the left half plane. For very large s we can ignore all but the highest power of s in (4.34) so that the magnitude (or modulus) of the roots that tend toward infinity shall satisfy the following approximate relation:

$$(-1)^n s^{2(n-m)} \approx \frac{b_m^2}{\mathbf{R}} (-1)^m \tag{4.37}$$

where we denote:

$$\begin{cases} \beta(s) = \det\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\right) = s^n + \beta_{n-1}\, s^{n-1} + \cdots + \beta_1\, s + \beta_0 \\ N(s) = b_m\, s^m + b_{m-1}\, s^{m-1} + \cdots + b_1\, s + b_0 \end{cases} \tag{4.38}$$

The roots of $\beta(-s)$ are the reflection across the imaginary of the roots of $\beta(s)$. Now express s in the exponential form:

$$s = r\, e^{j\theta} \tag{4.39}$$

We get from (4.37):

$$(-1)^n r^{2n} e^{j2n\theta} \approx \frac{b_m^2}{\mathbf{R}} (-1)^m r^{2m} e^{j2m\theta} \Rightarrow r^{2(n-m)} \approx \frac{b_m^2}{\mathbf{R}} \tag{4.40}$$

Therefore, the remaining $n-m$ zeros of $\beta(s)$ lie on a circle of radius r defined by:

$$r \approx \left(\frac{b_m}{\sqrt{\mathbf{R}}}\right)^{\frac{1}{n-m}} \tag{4.41}$$

The particular pattern to which the $2(n-m)$ solutions of (4.41) lie is known as the Butterworth configuration. The angle of the $2(n-m)$ branches which diverge to ∞ are obtained by adapting relation (4.11) to the case where transfer function reads $G(s)G(-s) = \frac{N(s)N(-s)}{D(s)D(-s)}$.

4.2.2 Shape of the magnitude of the loop gain

We recall that the loop gain $\mathbf{L}(s)$ is defined as follows:

$$\mathbf{L}(s) := \mathbf{K}\boldsymbol{\Phi}(s)\mathbf{B} \tag{4.42}$$

When $\mathbf{Q} = \mathbf{N}^T\mathbf{N}$, and assuming a SISO plant, Kalman equality (3.144) becomes:

$$\begin{aligned} \left(1 + \mathbf{L}(-s)\right)^T \left(1 + \mathbf{L}(s)\right) &= 1 + \tfrac{1}{\mathbf{R}} \left(\boldsymbol{\Phi}(-s)\mathbf{B}\right)^T \mathbf{N}^T\mathbf{N} \left(\boldsymbol{\Phi}(s)\mathbf{B}\right) \\ &= 1 + \tfrac{1}{\mathbf{R}} \left(\mathbf{N}\boldsymbol{\Phi}(-s)\mathbf{B}\right)^T \left(\mathbf{N}\left(\boldsymbol{\Phi}(s)\mathbf{B}\right)\right) \end{aligned} \tag{4.43}$$

Denoting by $\lambda(\mathbf{X}(j\omega))$ the eigenvalues of matrix $\mathbf{X}(j\omega)$ and by $\sigma(\mathbf{X}(j\omega))$ its singular values (that are the root square of the strictly positive eigenvalues of either $\mathbf{X}^T(-j\omega)\mathbf{X}(j\omega)$ or $\mathbf{X}(j\omega)\mathbf{X}^T(-j\omega)$), the preceding equality implies:

$$\begin{array}{rl} \lambda\left((1+\mathbf{L}(-j\omega))^T(1+\mathbf{L}(j\omega))\right) & = 1 + \frac{1}{\mathbf{R}}\lambda\left((\mathbf{N}\mathbf{\Phi}(-j\omega)\mathbf{B})^T(\mathbf{N}(\mathbf{\Phi}(j\omega)\mathbf{B}))\right) \\ \Leftrightarrow \sigma(1+\mathbf{L}(j\omega)) & = \sqrt{1+\frac{1}{\mathbf{R}}\sigma^2(\mathbf{N}\mathbf{\Phi}(j\omega)\mathbf{B})} \end{array} \tag{4.44}$$

For the range of frequencies for which $\sigma(\mathbf{N}\mathbf{\Phi}(j\omega)\mathbf{B}) \gg 1$ (typically low frequencies) equation (4.44) shows that:

$$\sigma(\mathbf{L}(j\omega)) \approx \frac{1}{\sqrt{\mathbf{R}}}\sigma(\mathbf{N}\mathbf{\Phi}(j\omega)\mathbf{B}) \tag{4.45}$$

For SISO system matrices $\mathbf{N}$ and $\mathbf{K}$ have the same dimension. Denoting by $|\mathbf{K}|$ the absolute value of each element of $\mathbf{K}$, and using the fact that $\mathbf{L}(s) := \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}$, we get from the previous equation :

$$\sigma(\mathbf{L}(j\omega)) \approx \frac{1}{\sqrt{\mathbf{R}}}\sigma(\mathbf{N}\mathbf{\Phi}(j\omega)\mathbf{B}) \Rightarrow |\mathbf{K}| \approx \frac{|\mathbf{N}|}{\sqrt{\mathbf{R}}} \text{ where } \mathbf{Q} = \mathbf{N}^T\mathbf{N} \tag{4.46}$$

Assuming that $\underline{z} = \mathbf{N}\,\underline{x}$, then $\mathbf{N}\mathbf{\Phi}(s)\mathbf{B}$ represents the transfer function from the control signal $u(t)$ to the controlled output $\underline{z}(t)$. As a consequence:

- The shape of the magnitude of the loop gain $\mathbf{L}(s)$ is determined by the magnitude of the transfer function from the control input $u(t)$ to the controlled output $\underline{z}(t)$;
- Parameter $\sqrt{\mathbf{R}}$, that is weight $\mathbf{R}$, moves the magnitude Bode plot up and down.

Note that although the magnitude of $\mathbf{L}(s)$ mimics the magnitude of $\mathbf{N}\mathbf{\Phi}(s)\mathbf{B}$, the phase of the loop gain $\mathbf{L}(s)$ always leads to a stable closed-loop with an appropriate phase margin. At high-frequency, it has been seen in Figure 3.5 and Figure 3.6 that the loop gain $\mathbf{L}(j\omega)$ can have at most -90 degrees phase for high-frequencies and therefore the *roll-off* rate is at most -20 dB/decade. In practice, this means that for $\omega \gg 1$, and for some constant a , we have the following approximation (remind that $\mathbf{\Phi}(s) = (s\mathbb{I} - \mathbf{A})^{-1} = \frac{\text{adj}(s\mathbb{I}-\mathbf{A})}{\det(s\mathbb{I}-\mathbf{A})}$ so that the degree of the denominator of $\mathbf{L}(s)$ is n and the degree of its numerator is at most $n-1$):

$$|\mathbf{L}(j\omega)| \approx \frac{a}{\omega\sqrt{\mathbf{R}}} \text{ where } \frac{a}{\sqrt{\mathbf{R}}} = \lim_{s\to\infty} s\,|\mathbf{L}(s)| \approx \lim_{s\to\infty} s\frac{1}{\sqrt{\mathbf{R}}}|\mathbf{N}\mathbf{\Phi}(s)\mathbf{B}| \tag{4.47}$$

Thus:

$$a = \lim_{s\to\infty} s\,|\mathbf{N}\mathbf{\Phi}(s)\mathbf{B}| \tag{4.48}$$

Therefore the cross-over frequency ω_c is approximately given by:

$$|\mathbf{L}(j\omega_c)| = 1 \approx \frac{a}{\omega_c\sqrt{\mathbf{R}}} \Rightarrow \omega_c \approx \frac{a}{\sqrt{\mathbf{R}}} \tag{4.49}$$

Consequently:

- LQR controllers always exhibit a high-frequency magnitude decay of -20 dB/decade. The (slow) -20 dB/decade magnitude decrease is the main shortcoming of state-feedback LQR controllers because it may not be sufficient to clear high-frequency upper bounds on the loop gain needed to reject disturbances and/or for robustness with respect to process uncertainty.
- The cross-over frequency is proportional to $1/\sqrt{\mathbf{R}}$ and generally small values for $\mathbf{R}$ result in faster step responses.

4.2.3 Weighting matrices selection

The preceding results motivates the following design rule extended to the case of multiple input multiple output systems:

- Modal point of view: assuming that all states are available for control, choose $\mathbf{N}$ (remind that $\mathbf{Q} = \mathbf{N}^T\mathbf{N} \Rightarrow \mathbf{Q}^{0.5} = \mathbf{N}$) such that $n-1$ zeros of $\mathbf{N\Phi}(s)\mathbf{B}$ are at the desired pole location. Then use cheap control $\sqrt{\mathbf{R}} \to 0$ to design LQ system so that $n-1$ poles of the closed-loop system approach these desired locations. It is worth noticing that for SISO plants the roots of $\mathbf{N\Phi}(s)\mathbf{B}$ are also the roots of:

$$\det\left(\begin{bmatrix} s\mathbb{I}-\mathbf{A} & -\mathbf{B} \\ \mathbf{N} & 0 \end{bmatrix}\right) = 0 \tag{4.50}$$

- Frequency point of view: alternatively we have seen that at low frequencies $|\mathbf{K}| \approx \frac{|\mathbf{N}|}{\sqrt{\mathbf{R}}}$ so that the loop gain is approximately $|\mathbf{L}(s)| \approx \frac{1}{\sqrt{\mathbf{R}}}|\mathbf{N\Phi}(s)\mathbf{B}|$. So the shape of the magnitude of the loop gain $\mathbf{L}(s)$ is determined by the magnitude of $\mathbf{N\Phi}(s)\mathbf{B}$, that is the transfer function from the control input $u(t)$ to the controlled output $\underline{z}(t)$. In addition, we have seen that at high frequency $|\mathbf{N\Phi}(j\omega)\mathbf{B}| \approx \frac{a}{\omega\sqrt{\mathbf{R}}}$, where $a = \lim_{s\to\infty} s|\mathbf{N\Phi}(s)\mathbf{B}|$ is some constant. So we can choose $\sqrt{\mathbf{R}}$ to pick the bandwidth ω_c which is where $|\mathbf{L}(j\omega)| = 1$. Thus choose $\sqrt{\mathbf{R}} \approx \frac{a}{\omega_c}$ where ω_c is the desired bandwidth.

Thus contrary to the Chang-Letov design procedure for Single-Input Single-Output (SISO) systems where scalar $\mathbf{R}$ was the design parameter the following design rules for Multi Input Multi Output (MIMO) systems use matrix $\mathbf{Q}$ as the design parameter. We may also use the fact that if λ_i is a stable eigenvalue (i.e. eigenvalue in the open left half plane) of the Hamiltonian matrix $\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{BR}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix}$ with eigenvector $\begin{bmatrix} \mathbf{X}_{1i} \\ \mathbf{X}_{2i} \end{bmatrix}$ then λ_i is also an eigenvalue of $\mathbf{A}-\mathbf{BK}$ with eigenvector $\mathbf{X}_{1i}$. Therefore in the single input case we can use this result by finding the eigenvalues of $\mathbf{H}$ and then realizing that the stable eigenvalues are the poles of the optimal closed-loop plant.

Alternatively, a simpler choice for matrices $\mathbf{Q}$ and $\mathbf{R}$ is given by the Bryson's rule who proposed to take $\mathbf{Q}$ and $\mathbf{R}$ as diagonal matrices such that:

$$\begin{cases} q_{ii} = \frac{1}{\text{max. acceptable value of } z_i^2} \\ r_{jj} = \frac{1}{\text{max. acceptable value of } u_j^2} \end{cases} \tag{4.51}$$

Diagonal matrices $\mathbf{Q}$ and $\mathbf{R}$ are associated to the following performance index where ρ is a free parameter to be set by the designer:

$$J(u(t)) = \frac{1}{2}\int_0^\infty \left(\sum_i q_{ii}\, z_i^2(t) + \rho^2 \sum_i r_{jj}\, u_j^2(t) \right) dt \tag{4.52}$$

If after simulation $|z_i(t)|$ is to large then increase q_{ii} ; similarly if after simulation $|u_j(t)|$ is to large then increase r_{jj}.

4.2.4 Poles assignment in optimal regulator using root locus

Let λ_i be an eigenvalue of the open-loop state matrix $\mathbf{A}$ corresponding to eigenvector $\underline{v}_i$. This open-loop eigenvalue will not be modified by state feedback gain $\mathbf{K}$ by setting in (4.120) the $m \times 1$ vector $\underline{p}_i$ to zero and the $n \times 1$ eigenvector $\underline{v}_{Ki}$ to the open-loop eigenvector $\underline{v}_i$ corresponding to eigenvalue λ_i:

$$\begin{array}{l} \left\{ \begin{array}{l} \mathbf{A}\underline{v}_i = \lambda_i\, \underline{v}_i \\ \mathbf{K} = \left[\begin{array}{ccc} \cdots & \underbrace{\underline{0}_{m\times 1}}_{i^{th} \text{ column}} & \cdots \end{array} \right] \left[\begin{array}{ccc} \cdots & \underbrace{\underline{v}_i}_{i^{th} \text{ column}} & \cdots \end{array} \right]^{-1} \end{array} \right. \\ \Rightarrow (\mathbf{A} - \mathbf{B}\mathbf{K})\, \underline{v}_i = \lambda_i \underline{v}_i \end{array} \tag{4.53}$$

Coming back to the general case, let $\underline{v}_1, \cdots, \underline{v}_n$ be the eigenvectors of the open-loop state matrix $\mathbf{A}$. Matrix $\mathbf{V}$ is defined as follows:

$$\mathbf{V} = \left[\begin{array}{ccc} \underline{v}_1 & \cdots & \underline{v}_n \end{array} \right] \tag{4.54}$$

Note that if λ_i and $\lambda_j := \bar{\lambda}_i$ are complex conjugate eigenvalues, then the corresponding eigenvectors $\underline{v}_i$ and $\underline{v}_j$ are also complex conjugate:

$$\lambda_j = \bar{\lambda}_i \Leftrightarrow \underline{v}_j = \bar{\underline{v}}_i \tag{4.55}$$

In order to get a real valued matrix $\mathbf{V}$, $\underline{v}_i$ and $\underline{v}_j$ shall be changed into the real part and imaginary part of $\underline{v}_i$, that is $Re(\underline{v}_i)$ and $Im(\underline{v}_i)$, respectively.

Let $\lambda_1, \cdots, \lambda_r$ be the $r \leq n$ eigenvalues that are desired to be changed by state feedback gain $\mathbf{K}$ and $\underline{v}_1, \cdots, \underline{v}_r$ the corresponding eigenvectors of the state matrix $\mathbf{A}$. Similarly let $\lambda_{r+1}, \cdots, \lambda_n$ be the $n-r$ eigenvalues that are desired to be kept invariant by state feedback gain $\mathbf{K}$ and $\underline{v}_{r+1}, \cdots, \underline{v}_n$ the corresponding eigenvectors of the state matrix $\mathbf{A}$. Assuming that matrix $\mathbf{V}$ is invertible, matrix $\mathbf{M}$ is defined and split as follows where $\mathbf{M}_r$ is an $r \times n$ matrix and $\mathbf{M}_{n-r}$ is an $(n-r) \times n$ matrix:

$$\mathbf{M} = \mathbf{V}^{-1} = \left[\begin{array}{cccccc} \underline{v}_1 & \cdots & \underline{v}_r & \underline{v}_{r+1} & \cdots & \underline{v}_n \end{array} \right]^{-1} = \left[\begin{array}{c} \mathbf{M}_r \\ \mathbf{M}_{n-r} \end{array} \right] \tag{4.56}$$

Shieh & al.[7] have shown that, once weighting matrix $\mathbf{R} = \mathbf{R}^T > 0$ is set, the characteristic polynomial $\beta(s)$ of the closed-loop transfer function is linked with the numerator and the denominator of the loop transfer function $\mathbf{\Phi}(s)\mathbf{B} = (s\mathbb{I} - \mathbf{A})^{-1}\mathbf{B}$ as follows:

$$\begin{array}{l} \mathbf{\Phi}(s)\mathbf{B} = (s\mathbb{I} - \mathbf{A})^{-1}\mathbf{B} = \frac{\mathrm{adj}(s\mathbb{I}-\mathbf{A})\mathbf{B}}{\det(s\mathbb{I}-\mathbf{A})} := \frac{\mathbf{N}_{ol}(s)}{D(s)} \\ \Rightarrow \beta(s)\,\beta(-s) = D(s)\,D(-s) + k_p\,\underline{N}_{rl}(s)\ \left(\underline{N}_{rl}(-s)\right)^T \end{array} \tag{4.57}$$

where:

$$\begin{cases} \mathbf{N}_{ol}(s) = \mathrm{adj}\,(s\mathbb{I} - \mathbf{A})\,\mathbf{B} \\ \underline{N}_{rl}(s) = \underline{q}_0^T \mathbf{M}_r \mathbf{N}_{ol}(s) \left(\mathbf{R}^{0.5}\right)^{-1} \\ \underline{q}_0 \in \mathbb{R}^{r\times 1} \end{cases} \tag{4.58}$$

Matrix $\mathbf{R}^{0.5} = \left(\mathbf{R}^{0.5}\right)^T$ is the root square of matrix $\mathbf{R}$. By getting the modal decomposition of matrix $\mathbf{R}$, that is $\mathbf{R} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$ where $\mathbf{V}$ is the matrix whose columns are the eigenvectors of $\mathbf{R}$ and $\mathbf{D}$ is the diagonal matrix whose diagonal elements are the corresponding positive eigenvalues, the square root $\mathbf{R}^{0.5}$ of $\mathbf{R}$ is given by $\mathbf{R}^{0.5} = \mathbf{V}\mathbf{D}^{0.5}\mathbf{V}^{-1}$, where $\mathbf{D}^{0.5}$ is any diagonal matrix whose elements are the square root of the diagonal elements of $\mathbf{D}$[2].

Relation (4.57) can be associated with root locus of the fictitious transfer function $G(s)G(-s) = \frac{N_{rl}^T(s)\,N_{rl}(-s)}{D(s)\,D(-s)}$ where fictitious gain k_p varies from 0 to ∞. The arbitrary nonzero $r \times 1$ column vector $\underline{q}_0$ is used to shape the locus. It is worth noticing that $\mathbf{M}_r\mathbf{N}_{ol}(s)$ and $D(s)$ share $\lambda_{r+1}, \cdots, \lambda_n$ as common roots, and thus pole / zero simplification within $G(s)$ and $G(-s)$ shall be done before drawing the root locus.

Once the positive scalar k_p has been selected on the root locus such that the r closed-loop eigenvalues $\lambda_{K1}, \cdots, \lambda_{Kr}$ are appropriately placed (note that for $k_p = 0$, the corresponding branches start at $\lambda_1, \cdots, \lambda_r$), the weighting matrix $\mathbf{Q}$ has the following expression:

$$\mathbf{Q} = k_p\,\mathbf{M}_r^T \left(\underline{q}_0\,\underline{q}_0^T\right) \mathbf{M}_r \tag{4.59}$$

The following relation also holds:

$$\mathbf{Q} \begin{bmatrix} \underline{v}_{r+1} & \cdots & \underline{v}_n \end{bmatrix} = \mathbf{0} \tag{4.60}$$

The preceding results is a generalization of the Chang-Letov design procedure seen in section 4.1.3. This may be used as follows: first choose $\mathbf{R}$ and set $\mathbf{Q} = \mathbf{0}$ to get the minimum energy optimal control law. Then identify the r closed-loop eigenvalues which do not meet design specifications. Finally compute $\mathbf{Q}$ as previously seen such that all eigenvalues are appropriately placed.

4.3 Poles shifting in optimal regulator

4.3.1 Mirror property

The purpose of this section is to underline the relation between the weighting matrix $\mathbf{Q}$ and the closed-loop eigenvalues of the optimal regulator.

[2] https://en.wikipedia.org/wiki/Square_root_of_a_matrix

We recall the expression of the $2n \times 2n$ Hamiltonian matrix $\mathbf{H}$:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \tag{4.61}$$

which corresponds to the following algebraic Riccati equation:

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \tag{4.62}$$

The characteristic polynomial of matrix $\mathbf{H}$ in (4.61) is given by[3]:

$$\det(s\mathbb{I} - \mathbf{H}) = \det(s\mathbb{I} - \mathbf{A})\det(\mathbb{I} - \mathbf{Q}\mathbf{S}(s))\det\left(s\mathbb{I} + \mathbf{A}^T\right) \tag{4.63}$$

Where the term $\mathbf{S}(s)$ is defined by:

$$\mathbf{S}(s) = (s\mathbb{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\left(s\mathbb{I} + \mathbf{A}^T\right)^{-1} \tag{4.64}$$

Setting $\mathbf{Q} = \mathbf{Q}^T = 2\alpha\mathbf{P}$, where $\alpha \geq 0$ is a design parameter, the algebraic Riccati equation reads:

$$\mathbf{Q} = \mathbf{Q}^T = 2\alpha\mathbf{P} \Rightarrow (\mathbf{A} + \alpha\mathbb{I})^T\mathbf{P} + \mathbf{P}(\mathbf{A} + \alpha\mathbb{I}) - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} = \mathbf{0} \tag{4.65}$$

which corresponds to the following Hamiltonian matrix $\mathbf{H}$:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} + \alpha\mathbb{I} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ \mathbf{0} & -(\mathbf{A} + \alpha\mathbb{I})^T \end{bmatrix} \tag{4.66}$$

Let λ_i be the open-loop eigenvalues, that are the eigenvalues of matrix $\mathbf{A}$. As far as the eigenvalues of a matrix are the same than the eigenvalues of its transpose, we can see that the $2n$ eigenvalues of the preceding Hamiltonian matrix is the set $\{\lambda_i + \alpha\} \cup \{-(\lambda_i + \alpha)\}$, $i = 1, \cdots, n$.

Because the eigenvalues λ_{α_i} of $\mathbf{A} + \alpha\mathbb{I} - \mathbf{B}\mathbf{K}$ are the n eigenvalues of the Hamiltonian matrix $\mathbf{H}$ with negative real part, and denoting $Re(\lambda)$ the real part of λ, there are two possibilities:

$$\begin{cases} Re(\lambda_i + \alpha) = Re(\lambda_i) + \alpha \leq 0 \Rightarrow \lambda_{\alpha_i} = \lambda_i + \alpha \\ Re(-(\lambda_i + \alpha)) = -Re(\lambda_i) - \alpha < 0 \Rightarrow \lambda_{\alpha_i} = -\lambda_i - \alpha \end{cases} \tag{4.67}$$

Finally let λ_{Ki} be the closed-loop eigenvalues, that are the eigenvalues of matrix $\mathbf{A} - \mathbf{B}\mathbf{K}$. The eigenvalues of $\mathbf{A} - \mathbf{B}\mathbf{K}$ are obtained from the eigenvalues of $\mathbf{A} + \alpha\mathbb{I} - \mathbf{B}\mathbf{K}$ by subtracting α to λ_{α_i}. Thus from (4.67), and given a controllable pair $(\mathbf{A}, \mathbf{B})$, a positive definite symmetric matrix $\mathbf{R}$ and a positive real constant α, the algebraic Riccati equation (4.65) where $\mathbf{Q} = \mathbf{Q}^T = 2\alpha\mathbf{P}$ has a unique positive definite solution $\mathbf{P} = \mathbf{P}^T > 0$ such that λ_{Ki} have the following property:

$$\begin{cases} Re(\lambda_i) \leq -\alpha \Rightarrow \lambda_{Ki} = \lambda_i \\ Re(\lambda_i) > -\alpha \Rightarrow \lambda_{Ki} = -\lambda_i - 2\alpha \quad \forall\, i = 1, \cdots, n \\ Im(\lambda_{Ki}) = Im(\lambda_i) \end{cases} \tag{4.68}$$

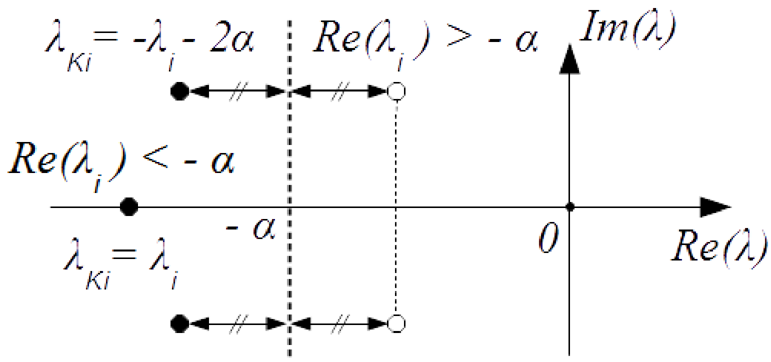


Figure 4.2: Mirror property of LQR design when $\mathbf{Q} = \mathbf{Q}^T = 2\alpha\mathbf{P}$

This *mirror property* is illustrated in Figure 4.2.

Consequently, and denoting by $Re(\lambda_{Ki})$ the real part of λ_{Ki}, it can be shown that the positive definite real symmetric solution $\mathbf{P}$ of (4.65) is such that the following *mirror property* holds[4]:

$$\begin{cases} Re(\lambda_{Ki}) \leq -\alpha \\ Im(\lambda_{Ki}) = Im(\lambda_i) \\ (\alpha + \lambda_i)^2 = (\alpha + \lambda_{Ki})^2 \end{cases} \quad \forall\, i = 1, \cdots, n \tag{4.69}$$

Once the algebraic Riccati equation (4.65) is solved in $\mathbf{P}$ the classical LQR design is applied:

$$\begin{cases} u(t) = -\mathbf{K}\underline{x}(t) \\ \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \end{cases} \tag{4.70}$$

It is worth noticing that the algebraic Riccati equation (4.65) can be changed into a Lyapunov equation by pre- and post-multiplying (4.65) by $\mathbf{P}^{-1}$ and setting $\mathbf{X} := \mathbf{P}^{-1}$:

$$\begin{array}{l} (\mathbf{A} + \alpha\mathbb{I})^T\mathbf{P} + \mathbf{P}(\mathbf{A} + \alpha\mathbb{I}) - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} = \mathbf{0} \\ \Rightarrow \mathbf{P}^{-1}(\mathbf{A} + \alpha\mathbb{I})^T + (\mathbf{A} + \alpha\mathbb{I})\mathbf{P}^{-1} - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T = \mathbf{0} \\ \mathbf{X} := \mathbf{P}^{-1} \Rightarrow \mathbf{X}(\mathbf{A} + \alpha\mathbb{I})^T + (\mathbf{A} + \alpha\mathbb{I})\mathbf{X} = \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \end{array} \tag{4.71}$$

Matrix $\mathbf{R}$ remains the degree of freedom for the design and it seems that it may be used to set the damping ratio of the complex conjugate dominant poles for example. Unfortunately (4.66) indicates that the eigenvalues of the Hamiltonian matrix $\mathbf{H}$, which are closely related to eigenvalues of the closed-loop system, are independent of matrix $\mathbf{R}$. Thus matrix $\mathbf{R}$ has no influence on the location of the closed-loop poles in that situation.

Furthermore it is worth reminding that the higher the displacement of closed-loop eigenvalues with respect to the open-loop eigenvalue is, the higher the control effort is. Thus specifying very fast dominant poles may lead to unacceptable control effort.

[3] Y. Ochi and K. Kanai, Pole placement in optimal regulator by continuous pole-shifting, Journal of Guidance Control and Dynamics, Vol. 18, No. 6 (1995), pp. 1253-1258

[4] Optimal pole shifting for continuous multivariable linear systems, M. H. Amin, Int. Journal of Control 41 No. 3 (1985), 701–707.

4.3.2 Reduced-order model

The preceding result can be used to recursively shift on the left all the real parts of the poles of a system to any positions while preserving their imaginary parts. Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be the state matrix of the system to be controlled and $\mathbf{B} \in \mathbb{R}^{n\times m}$ the input matrix. We assume that all the eigenvalues of $\mathbf{A}$ are distinct and that $(\mathbf{A}, \mathbf{B})$ is controllable and that the symmetric positive definite weighting matrix $\mathbf{R}$ for the control is given. The purpose of this section is to compute the state weighting matrix $\mathbf{Q}$ which leads to the desired closed-loop eigenvalues by shifting recursively the actual eigenvalues of the state matrix. It is worth noticing that, through the shifting process, real eigenvalues remain real eigenvalues whereas complex conjugate eigenvalues remain complex conjugate eigenvalues.

The core idea of the method is to consider the transformation $\underline{z}_i = \mathbf{C}^T \underline{x}$ where $\mathbf{C}$ is appropriately chosen. This leads to the following reduced order model where matrix $\mathbf{\Lambda}$ corresponds to the diagonal (or Jordan) form of state matrix $\mathbf{A}$:

$$\underline{z}_i = \mathbf{C}^T \underline{x} \Rightarrow \dot{\underline{z}}_i = \mathbf{\Lambda}\underline{z}_i + \mathbf{G}\underline{u} \text{ where } \begin{cases} \mathbf{C}^T\mathbf{A} = \mathbf{\Lambda}\mathbf{C}^T \Leftrightarrow \mathbf{A}^T\mathbf{C} = \mathbf{C}\mathbf{\Lambda}^T \\ \mathbf{G} = \mathbf{C}^T\mathbf{B} \end{cases} \tag{4.72}$$

In this new basis the performance index turns to be:

$$J_i = \frac{1}{2}\int_0^\infty \left(\underline{z}_i^T \widetilde{\mathbf{Q}}_i \underline{z}_i + \underline{u}^T\mathbf{R}\underline{u}\right) dt \text{ where } \mathbf{Q} = \mathbf{C}\widetilde{\mathbf{Q}}_i\mathbf{C}^T \tag{4.73}$$

4.3.3 Shifting one real eigenvalue

Let λ_i be an eigenvalue of $\mathbf{A}$. We will first assume that λ_i is real. We wish to shift λ_i to λ_{Ki}.

Let $\underline{v}$ be a *left* eigenvector of $\mathbf{A}$: $\underline{v}^T\mathbf{A} = \lambda_i\, \underline{v}^T$. In other words, $\underline{v}$ is a (right) eigenvector of $\mathbf{A}^T$ corresponding to λ_i: $\mathbf{A}^T\underline{v} = \lambda_i\, \underline{v}$. Then we define $\underline{z}_i$ as follows:

$$\underline{z}_i := \mathbf{C}^T\underline{x} \text{ where } \mathbf{C} = \underline{v} \tag{4.74}$$

Using the fact that $\underline{v}$ is a (right) eigenvector of $\mathbf{A}^T$ ($\underline{z}_i = \underline{v}^T\underline{x}$), we can write:

$$\begin{array}{ll} \dot{\underline{z}}_i & = \underline{v}^T\mathbf{A}\underline{x} + \underline{v}^T\mathbf{B}\underline{u} \\ & = \underline{v}^T\lambda_i\underline{x} + \underline{v}^T\mathbf{B}\underline{u} \\ & = \lambda_i\underline{v}^T\underline{x} + \underline{v}^T\mathbf{B}\underline{u} \\ & = \lambda_i\underline{z}_i + \underline{v}^T\mathbf{B}\underline{u} \\ & = \lambda_i\underline{z}_i + \mathbf{G}\underline{u} \text{ where } \mathbf{G} := \underline{v}^T\mathbf{B} = \mathbf{C}^T\mathbf{B} \end{array} \tag{4.75}$$

Then setting $\underline{u} := -\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}}\underline{z}_i$, where scalar $\widetilde{\mathbf{P}} > 0$ is a design parameter, and having in mind that $\underline{z}_i$ is scalar (thus $\lambda_i\mathbb{I} = \lambda_i$), we get:

$$\dot{\underline{z}}_i = \left(\lambda_i - \mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}}\right)\underline{z}_i \tag{4.76}$$

Let λ_{Ki} be the desired eigenvalue of the preceding reduced-order model. Then we shall have:

$$\lambda_{Ki} = \lambda_i - \mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}} \tag{4.77}$$

Thus, matrix $\widetilde{\mathbf{P}}$ reads:

$$\widetilde{\mathbf{P}} = \frac{\lambda_i - \lambda_{Ki}}{\mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T} \tag{4.78}$$

The state weighting matrix $\mathbf{Q}_i$ that will shift the open-loop eigenvalue λ_i to the closed-loop eigenvalue λ_{Ki} is obtained through the following identification: $\underline{z}_i^T \widetilde{\mathbf{Q}}_i \underline{z}_i = \underline{x}^T \mathbf{Q}_i \underline{x}$. We finally get:

$$\underline{z}_i = \underline{v}^T \underline{x} := \mathbf{C}^T \underline{x} \Rightarrow \mathbf{Q}_i = \mathbf{C}\widetilde{\mathbf{Q}}_i\mathbf{C}^T \tag{4.79}$$

Once matrix $\widetilde{\mathbf{P}}$ has been computed, matrix $\widetilde{\mathbf{Q}}_i$ is obtained thanks to the corresponding algebraic Riccati equation:

$$\begin{array}{l} \mathbf{0} = \widetilde{\mathbf{P}}\lambda_i + \lambda_i\widetilde{\mathbf{P}} - \widetilde{\mathbf{P}}\mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}} + \widetilde{\mathbf{Q}}_i \\ \Leftrightarrow \widetilde{\mathbf{Q}}_i = -2\lambda_i\widetilde{\mathbf{P}} + \widetilde{\mathbf{P}}\mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}} \end{array} \tag{4.80}$$

4.3.4 Shifting a pair of complex conjugate eigenvalues

The procedure to shift a pair of complex conjugate eigenvalues follows the same idea: let λ_i and $\bar{\lambda}_i$ be a pair of complex conjugate eigenvalues of $\mathbf{A}$. We wish to shift λ_i and $\bar{\lambda}_i$ to λ_{Ki} and $\bar{\lambda}_{Ki}$.

Let $\underline{v}$ and $\bar{\underline{v}}$ be a pair *left* eigenvectors of $\mathbf{A}$. In other words, $\underline{v}$ and $\bar{\underline{v}}$ is a pair of (right) eigenvector of $\mathbf{A}^T$ corresponding to λ_i:

$$\begin{array}{rl} \begin{bmatrix} \underline{v}^T & \bar{\underline{v}}^T \end{bmatrix} \mathbf{A} & = \begin{bmatrix} \underline{v}^T & \bar{\underline{v}}^T \end{bmatrix} \begin{bmatrix} \lambda_i & 0 \\ 0 & \bar{\lambda}_i \end{bmatrix} \\ \Leftrightarrow \mathbf{A}^T \begin{bmatrix} \underline{v} & \bar{\underline{v}} \end{bmatrix} & = \begin{bmatrix} \underline{v} & \bar{\underline{v}} \end{bmatrix} \begin{bmatrix} \lambda_i & 0 \\ 0 & \bar{\lambda}_i \end{bmatrix} \end{array} \tag{4.81}$$

In order to manipulate real values, we will use the real part and the imaginary part of the preceding equation. Denoting $\lambda_i := a + j\,b$, that is $a := Re(\lambda_i)$ and $b := Im(\lambda_i)$, the preceding relation is equivalently replaced by the following one:

$$\begin{array}{rl} \mathbf{A}^T \begin{bmatrix} \underline{v} & \bar{\underline{v}} \end{bmatrix} & = \begin{bmatrix} \underline{v} & \bar{\underline{v}} \end{bmatrix} \begin{bmatrix} \lambda_i & 0 \\ 0 & \bar{\lambda}_i \end{bmatrix} \\ \Leftrightarrow \mathbf{A}^T \begin{bmatrix} Re(\underline{v}) & Im(\underline{v}) \end{bmatrix} & = \begin{bmatrix} Re(\underline{v}) & Im(\underline{v}) \end{bmatrix} \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \end{array} \tag{4.82}$$

Then define $\underline{z}_i$ as follows:

$$\underline{z}_i := \mathbf{C}^T \underline{x} \text{ where } \mathbf{C} = \begin{bmatrix} Re(\underline{v}) & Im(\underline{v}) \end{bmatrix} \tag{4.83}$$

Using the fact that $\underline{v}$ and $\bar{\underline{v}}$ is a pair of (right) eigenvector of $\mathbf{A}^T$, we get:

$$\dot{\underline{z}}_i \;\; = \mathbf{A}_i \underline{z}_i + \mathbf{G}\underline{u} \tag{4.84}$$

where:

$$\left\{ \begin{array}{l} \mathbf{G} = \mathbf{C}^T\mathbf{B} \\ \mathbf{A}_i = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \end{array} \right. \tag{4.85}$$

Then setting $\underline{u} = -\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}}\,\underline{z}_i$, , where 2×2 positive definite matrix $\widetilde{\mathbf{P}}$ is a design parameter, we get:

$$\dot{\underline{z}}_i = \mathbf{\Lambda}_i\,\underline{z}_i \text{ where } \left\{ \begin{array}{l} \mathbf{\Lambda}_i = \mathbf{A}_i - \mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}} \\ \widetilde{\mathbf{P}} = \widetilde{\mathbf{P}}^T = \left[\begin{array}{cc} \widetilde{p}_1 & \widetilde{p}_2 \\ \widetilde{p}_2 & \widetilde{p}_3 \end{array} \right] > 0 \end{array} \right. \tag{4.86}$$

Thus the closed-loop eigenvalues are the eigenvalues of matrix $\mathbf{\Lambda}_i$. Here the design process becomes a little bit more involved because parameters $\widetilde{p}_1$, $\widetilde{p}_2$ and $\widetilde{p}_3$ of matrix $\widetilde{\mathbf{P}}$ shall be chosen to meet the desired complex conjugate closed-loop eigenvalues λ_{Ki} and $\bar{\lambda}_{Ki}$ while minimizing the trace of $\widetilde{\mathbf{P}}$ (indeed it can be shown that $min(J_i) = min(\text{tr}(\widetilde{\mathbf{P}}))$). The design process has been described by Arar & Sawan[5].

Alternatively, we can choose the three coefficients $\widetilde{q}_1$, $\widetilde{q}_2$ and $\widetilde{q}_3$ of matrix $\widetilde{\mathbf{Q}}_i = \widetilde{\mathbf{Q}}_i^T \geq 0$ such that the eigenvalues with negative real part of the following Hamiltonian matrix $\widetilde{\mathbf{H}}_i$ correspond to the desired eigenvalues λ_{Ki} and $\bar{\lambda}_{Ki}$, as proposed by Fujinaka & Omatu[6].

Thus the problem consists to find matrix $\widetilde{\mathbf{Q}}_i$:

$$\widetilde{\mathbf{Q}}_i = \left[\begin{array}{cc} \widetilde{q}_1 & \widetilde{q}_2 \\ \widetilde{q}_2 & \widetilde{q}_3 \end{array} \right] = \widetilde{\mathbf{Q}}_i^T \geq 0 \tag{4.87}$$

such that:

$$det(s\mathbb{I} - \widetilde{\mathbf{H}}_i) = (s - \lambda_{Ki})(s - \bar{\lambda}_{Ki})(s + \lambda_{Ki})(s + \bar{\lambda}_{Ki}) = s^4 + c_2\,s^2 + c_0 \tag{4.88}$$

where:

$$\widetilde{\mathbf{H}}_i = \left[\begin{array}{cc} \mathbf{A}_i & -\mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T \\ -\widetilde{\mathbf{Q}}_i & -\mathbf{A}_i \end{array} \right] \tag{4.89}$$

Once matrix $\widetilde{\mathbf{Q}}_i$ has been computed, matrix $\mathbf{Q}$ is obtained as follows:

$$\mathbf{Q} = \mathbf{C}\widetilde{\mathbf{Q}}_i\mathbf{C}^T \tag{4.90}$$

4.3.5 Sequential pole shifting via reduced-order models

When the imaginary part of the shifted eigenvalues is preserved, that is when $Im(\lambda_{Ki}) = Im(\lambda_i)$ and $Im(\bar{\lambda}_{Ki}) = Im(\bar{\lambda}_i)$, then the design process can be simplified by using the mirror property underlined by Amin[4] and presented in Section 4.3.1: given a controllable pair $(\mathbf{\Lambda}_i, \mathbf{G})$, a positive definite symmetric matrix $\mathbf{R}$ and a positive real constant α, then the following algebraic Riccati equation has a unique positive definite solution $\widetilde{\mathbf{P}} = \widetilde{\mathbf{P}}^T > 0$:

$$(\mathbf{\Lambda}_i + \alpha\mathbb{I})^T\,\widetilde{\mathbf{P}} + \widetilde{\mathbf{P}}\,(\mathbf{\Lambda}_i + \alpha\mathbb{I}) - \widetilde{\mathbf{P}}\mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}} = \mathbf{0} \tag{4.91}$$

[5] Abdul-Razzaq S. Arar, Mahmoud E. Sawan, Optimal pole placement with prescribed eigenvalues for continuous systems, Journal of the Franklin Institute, Volume 330, Issue 5, September 1993, Pages 985-994

[6] Toru Fujinaka, Sigeru Omatu, Pole Placement Using Optimal Regulators, IEEJ Transactions on Electronics Information and Systems 121(1):240-245, January 2001, DOI:10.1541/ieejeiss1987.121.1_240

Moreover the feedback control law $\underline{u} = -\mathbf{K}_i \, \underline{x}$ shift the pair of complex conjugate eigenvalues $(\lambda_i, \bar{\lambda}_i)$ of matrix $\mathbf{A}$ to a pair of complex conjugate eigenvalues $(\lambda_{Ki}, \bar{\lambda}_{Ki})$ as follows, assuming $\alpha + Re(\lambda_i) \geq 0$:

$$\left\{ \begin{array}{l} \mathbf{P}_i = \mathbf{C}\widetilde{\mathbf{P}}\mathbf{C}^T \\ \mathbf{Q}_i = 2\alpha \, \mathbf{P}_i \\ \mathbf{K}_i = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_i = \mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}}\mathbf{C}^T \end{array} \right. \Rightarrow \left\{ \begin{array}{l} Re(\lambda_{Ki}) = -\left(2\alpha + Re(\lambda_i)\right) \\ Im(\lambda_{Ki}) = Im(\lambda_i) \end{array} \right. \tag{4.92}$$

The design process proposed by Amin[4] to shift several eigenvalues recursively is the following:

1. Set $i = 1$ and $\mathbf{A}_1 = \mathbf{A}$.

2. Let λ_i be the eigenvalue of matrix $\mathbf{A}_i$ which is desired to be shifted:

   – Assume that λ_i is real. We wish to shift λ_i to $\lambda_{Ki} \leq \lambda_i$. Then compute a (right) eigenvector $\underline{v}$ of $\mathbf{A}_i^T$ corresponding to λ_i. In other words $\underline{v}^T$ is the *left* eigenvector of $\mathbf{A}_i$: $\underline{v}^T \mathbf{A}_i = \lambda_i \, \underline{v}^T$. Then compute $\mathbf{C}$, $\mathbf{G}$, α and $\mathbf{\Lambda}_i$ defined by:

   $$\left\{ \begin{array}{l} \mathbf{C} = \underline{v} \\ \mathbf{G} = \mathbf{C}^T\mathbf{B} \\ \alpha = -\frac{\lambda_{Ki}+\lambda_i}{2} \geq 0 \text{ where } \lambda_{Ki} \leq \lambda_i \in \mathbb{R} \\ \mathbf{\Lambda}_i = \lambda_i \, \in \mathbb{R} \end{array} \right. \tag{4.93}$$

   – Now assume that $\lambda_i = a + jb$ is complex. We wish to shift λ_i and $\bar{\lambda}_i$ to λ_{Ki} and $\bar{\lambda}_{Ki}$ where:

   $$\left\{ \begin{array}{l} Re(\lambda_{Ki}) \leq Re(\lambda_i) := a \\ Im(\lambda_{Ki}) = Im(\lambda_i) := b \end{array} \right. \tag{4.94}$$

   This means that the shifted poles shall have the same imaginary parts than the original ones. Then compute (right) eigenvectors $(\underline{v}_1, \underline{v}_2)$ of $\mathbf{A}_i^T$ corresponding to λ_i and $\bar{\lambda}_i$. In other words $(\underline{v}_1^T, \underline{v}_2^T)$ are the *left* eigenvectors of $\mathbf{A}_i$: $\begin{bmatrix} \underline{v}_1^T \\ \underline{v}_2^T \end{bmatrix} \mathbf{A}_i = \begin{bmatrix} \lambda_i & 0 \\ 0 & \bar{\lambda}_i \end{bmatrix} \begin{bmatrix} \underline{v}_1^T \\ \underline{v}_2^T \end{bmatrix}$. Then compute $\mathbf{C}$, $\mathbf{G}$, α and $\mathbf{\Lambda}_i$ defined by:

   $$\left\{ \begin{array}{l} \underline{v}_1 = \bar{\underline{v}}_2 \Rightarrow \mathbf{C} = \begin{bmatrix} Re(\underline{v}_1) & Im(\underline{v}_1) \end{bmatrix} \\ \mathbf{G} = \mathbf{C}^T\mathbf{B} \\ \alpha = -\frac{Re(\lambda_{Ki}+\lambda_i)}{2} \geq 0 \\ \lambda_i = a + jb \, \in \mathbb{C} \Rightarrow \mathbf{\Lambda}_i = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \end{array} \right. \tag{4.95}$$

3. Compute $\widetilde{\mathbf{P}} = \widetilde{\mathbf{P}}^T > 0$, which is defined as the unique positive definite solution of the following algebraic Riccati equation:

   $$(\mathbf{\Lambda}_i + \alpha\mathbb{I})^T \, \widetilde{\mathbf{P}} + \widetilde{\mathbf{P}} \, (\mathbf{\Lambda}_i + \alpha\mathbb{I}) - \widetilde{\mathbf{P}}\mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}} = \mathbf{0} \tag{4.96}$$

Alternatively, $\widetilde{\mathbf{P}}$ can be defined as follows:

$$\widetilde{\mathbf{P}} = \mathbf{X}^{-1} \tag{4.97}$$

where $\mathbf{X}$ is the solution of the following Lyapunov equation:

$$(\mathbf{\Lambda}_i + \alpha\mathbb{I})\,\mathbf{X} + \mathbf{X}\,(\mathbf{\Lambda}_i + \alpha\mathbb{I})^T = \mathbf{G}\mathbf{R}^{-1}\mathbf{G}^T \tag{4.98}$$

4. Compute $\mathbf{P}_i$, $\mathbf{Q}_i$ and $\mathbf{K}_i$ as follows:

$$\begin{cases} \mathbf{P}_i = \mathbf{C}\widetilde{\mathbf{P}}\mathbf{C}^T \\ \mathbf{Q}_i = 2\alpha\,\mathbf{P}_i \\ \mathbf{K}_i = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_i = \mathbf{R}^{-1}\mathbf{G}^T\widetilde{\mathbf{P}}\mathbf{C}^T \end{cases} \tag{4.99}$$

5. Set $i = i + 1$ and $\mathbf{A}_i = \mathbf{A}_{i-1} - \mathbf{B}\,\mathbf{K}_{i-1}$. Go to step 2 if some others open-loop eigenvalues have to be shifted.

Once the loop is finished compute $\mathbf{P} = \sum_i \mathbf{P}_i$, $\mathbf{Q} = \sum_i \mathbf{Q}_i$ and $\mathbf{K} = \sum_i \mathbf{K}_i$. Gain $\mathbf{K}$ is such that eigenvalues of $\mathbf{A} - \mathbf{B}\mathbf{K}$ are located to the desired values λ_{Ki}. Furthermore $\mathbf{Q}$ is the weighting matrix for the state vector and $\mathbf{P}$ is the positive definite solution of the corresponding algebraic Riccati equation.

4.4 Frequency domain approach

4.4.1 Non optimal pole assignment

We have seen in (3.139) that thanks to the Hsu-Chen theorem the closed-loop characteristic polynomial $\det(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K})$ reads as follows:

$$\det(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}) = \det(s\mathbb{I} - \mathbf{A})\det(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}) \tag{4.100}$$

Let $D(s) = \det(s\mathbb{I} - \mathbf{A})$ be the determinant of $\mathbf{\Phi}(s)$, that is the plant characteristic polynomial, and $\mathbf{N}_{ol}(s) = \operatorname{adj}(s\mathbb{I} - \mathbf{A})\,\mathbf{B}$ be the adjugate matrix of $s\mathbb{I} - \mathbf{A}$ times matrix $\mathbf{B}$:

$$\mathbf{\Phi}(s)\mathbf{B} = (s\mathbb{I} - \mathbf{A})^{-1}\,\mathbf{B} = \frac{\operatorname{adj}(s\mathbb{I} - \mathbf{A})\,\mathbf{B}}{\det(s\mathbb{I} - \mathbf{A})} := \frac{\mathbf{N}_{ol}(s)}{D(s)} \tag{4.101}$$

Consequently (4.100) reads:

$$\det(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}) = \det(D(s)\mathbb{I} + \mathbf{K}\mathbf{N}_{ol}(s)) \tag{4.102}$$

As soon as λ_{Ki} is a desired closed-loop eigenvalue then the following relation holds:

$$\det(D(s)\mathbb{I} + \mathbf{K}\mathbf{N}_{ol}(s))|_{s=\lambda_{Ki}} = 0 \tag{4.103}$$

Consequently it is desired that matrix $D(s)\mathbb{I} + \mathbf{K}\mathbf{N}_{ol}(s)|_{s=\lambda_{Ki}}$ is singular. Let $\underline{\omega_i}$ be a vector belonging to the kernel of $D(s)\mathbb{I} + \mathbf{K}\mathbf{N}_{ol}(s)|_{s=\lambda_{Ki}}$. Thus replacing s by λ_{Ki} we can write:

$$(D(\lambda_{Ki})\mathbb{I} + \mathbf{K}\mathbf{N}_{ol}(\lambda_{Ki}))\,\underline{\omega_i} = \underline{0} \tag{4.104}$$

Actually, vector $\underline{\omega}_i \neq \underline{0}$ can be used as a design parameter.

Alternatively, when making the parallel that λ_i is an eigenvalue of matrix $\mathbf{A}$ as soon as $\det(s\mathbb{I} - \mathbf{A})|_{s=\lambda_i} = 0$, we conclude that $D(\lambda_{Ki})$ is an eigenvalue of matrix $-\mathbf{K}\mathbf{N}_{ol}(\lambda_{Ki})$, and thus $\underline{\omega}_i$ is an eigenvector of $-\mathbf{K}\mathbf{N}_{ol}(\lambda_{Ki})$ corresponding to the eigenvalue $D(\lambda_{Ki})$. This remark can be extended to the output feedback case where $\mathbf{N}_{ol}(s) = \mathbf{C}\,\mathrm{adj}\,(s\mathbb{I} - \mathbf{A})\,\mathbf{B}$.

In order to get gain $\mathbf{K}$ the preceding relation is rewritten as follows:

$$\mathbf{K}\mathbf{N}_{ol}(\lambda_{Ki})\underline{\omega}_i = -D(\lambda_{Ki})\underline{\omega}_i \tag{4.105}$$

This relation does not lead to the value of gain $\mathbf{K}$ as soon as $\mathbf{N}_{ol}(\lambda_{Ki})\underline{\omega}_i$ is a vector which is not invertible. Nevertheless assuming that n denotes the order of state matrix $\mathbf{A}$ we can apply this relation for the n desired closed-loop eigenvalues. We get:

$$\mathbf{K}\begin{bmatrix} \underline{v}_{K1} & \cdots & \underline{v}_{Kn} \end{bmatrix} = -\begin{bmatrix} \underline{p}_1 & \cdots & \underline{p}_n \end{bmatrix} \tag{4.106}$$

where vectors $\underline{v}_{Ki}$ and $\underline{p}_i$ are given by:

$$\begin{cases} \underline{v}_{Ki} = \mathbf{N}_{ol}(\lambda_{Ki})\,\underline{\omega}_i \\ \underline{p}_i = D(\lambda_{Ki})\,\underline{\omega}_i \end{cases} \tag{4.107}$$

We finally get the following expression of gain $\mathbf{K}$:

$$\mathbf{K} = -\begin{bmatrix} \underline{p}_1 & \cdots & \underline{p}_n \end{bmatrix}\begin{bmatrix} \underline{v}_{K1} & \cdots & \underline{v}_{Kn} \end{bmatrix}^{-1} \tag{4.108}$$

4.4.2 Assignment of weighting matrices Q and R

The starting point is the Kalman equality (3.144) that we recall hereafter:

$$(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(-s)\mathbf{B})^T\,\mathbf{R}\,(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}) = \mathbf{R} + (\mathbf{\Phi}(-s)\mathbf{B})^T\,\mathbf{Q}\,(\mathbf{\Phi}(s)\mathbf{B}) \tag{4.109}$$

Using the fact that $\det(\mathbf{X}\mathbf{Y}) = \det(\mathbf{X})\det(\mathbf{Y})$ leads to the following result:

$$\begin{array}{rl} \det\left((\mathbb{I} + \mathbf{K}\mathbf{\Phi}(-s)\mathbf{B})^T\right)\,\det(\mathbf{R}) & \det\left((\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B})\right) \\ & = \det\left(\mathbf{R} + (\mathbf{\Phi}(-s)\mathbf{B})^T\,\mathbf{Q}\,(\mathbf{\Phi}(s)\mathbf{B})\right) \\ \Leftrightarrow \det\left((\mathbb{I} + \mathbf{K}\mathbf{\Phi}(-s)\mathbf{B})^T\right) & \det\left((\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B})\right) \\ & = \det\left(\mathbb{I} + \mathbf{R}^{-1}\,(\mathbf{\Phi}(-s)\mathbf{B})^T\,\mathbf{Q}\,(\mathbf{\Phi}(s)\mathbf{B})\right) \end{array} \tag{4.110}$$

On the other hand, let $D(s)$ be the open-loop characteristic polynomial and $\beta(s)$ be the closed-loop characteristic polynomial:

$$\begin{cases} D(s) = \det(s\mathbb{I} - \mathbf{A}) \\ \beta(s) = \det(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}) \end{cases} \tag{4.111}$$

As in the previous section, let $\mathbf{N}_{ol}(s)$ be the following polynomial matrix:

$$\mathbf{N}_{ol}(s) := \mathrm{adj}\,(s\mathbb{I} - \mathbf{A})\,\mathbf{B} \tag{4.112}$$

Then we get:

$$\mathbf{\Phi}(s) := (s\mathbb{I} - \mathbf{A})^{-1} \Rightarrow \mathbf{\Phi}(s)\,\mathbf{B} = (s\mathbb{I} - \mathbf{A})^{-1}\,\mathbf{B} := \frac{\mathbf{N}_{ol}(s)}{D(s)} \tag{4.113}$$

Furthermore the Hsu-Chen equality (3.139) reads as follows with those notations:

$$\begin{array}{l} \det\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\right) = \det\left(s\mathbb{I} - \mathbf{A}\right)\det\left(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}\right) \\ \qquad \Leftrightarrow \det\left(\mathbb{I} + \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}\right) = \dfrac{\det\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\right)}{\det\left(s\mathbb{I} - \mathbf{A}\right)} := \dfrac{\beta(s)}{D(s)} \end{array} \tag{4.114}$$

Finally, using the fact that $\det\left(\mathbf{X}^T\right) = \det\left(\mathbf{X}\right)$, relation (4.110) becomes:

$$\frac{\beta(-s)}{D(-s)}\,\frac{\beta(s)}{D(s)} = \det\left(\mathbb{I} + \mathbf{R}^{-1}\,\left(\mathbf{\Phi}(-s)\mathbf{B}\right)^T \mathbf{Q}\left(\mathbf{\Phi}(s)\mathbf{B}\right)\right) \tag{4.115}$$

We finally get the following result where $\det\left(\mathbb{I} + \mathbf{R}^{-1}\left(\mathbf{\Phi}(-s)\mathbf{B}\right)^T \mathbf{Q}\left(\mathbf{\Phi}(s)\mathbf{B}\right)\right)$ is a rational fraction whose denominator is $D(s)\,D(-s)$, that is the denominator of $\mathbf{\Phi}(-s)\mathbf{\Phi}(s)$:

$$\begin{aligned} \beta(s)\,\beta(-s) &= D(s)\,D(-s)\det\left(\mathbb{I} + \mathbf{R}^{-1}\left(\mathbf{\Phi}(-s)\mathbf{B}\right)^T \mathbf{Q}\left(\mathbf{\Phi}(s)\mathbf{B}\right)\right) \\ &= D(s)\,D(-s)\det\left(\mathbb{I} + \mathbf{R}^{-1}\left(\tfrac{\mathbf{N}_{ol}(-s)}{D(-s)}\right)^T \mathbf{Q}\,\tfrac{\mathbf{N}_{ol}(s)}{D(s)}\right) \\ &= \det\left(D(s)\,D(-s)\,\mathbb{I} + \mathbf{R}^{-1}\mathbf{N}_{ol}(-s)^T\,\mathbf{Q}\,\mathbf{N}_{ol}(s)\right) \end{aligned} \tag{4.116}$$

Thus the closed-loop eigenvalues are the roots λ_{Ki} with negative real part such that:

$$\det\left(D(s)\,D(-s)\,\mathbb{I} + \mathbf{R}^{-1}\mathbf{N}_{ol}(-s)^T\,\mathbf{Q}\,\mathbf{N}_{ol}(s)\right)\Big|_{s=\lambda_{Ki}} = 0 \tag{4.117}$$

Relation (4.117) indicates there exists eigenvectors $\underline{\omega}_i \neq \underline{0}$ such that for a given closed-loop eigenvalue λ_{Ki} the following relation holds[7]:

$$\boxed{\left(D(-\lambda_{Ki})D(\lambda_{Ki})\mathbb{I} + \mathbf{R}^{-1}\left(\mathbf{N}_{ol}(-\lambda_{Ki})\right)^T \mathbf{Q}\,\mathbf{N}_{ol}(\lambda_{Ki})\right)\underline{\omega}_i = \underline{0}} \tag{4.118}$$

Let n be the order of state matrix $\mathbf{A}$. Once eigenvector $\underline{\omega}_i \neq \underline{0}$ has been obtained for each λ_{Ki}, relation (4.108) can be used to compute the optimal gain $\mathbf{K}$ as follows:

$$\boxed{\mathbf{K} = -\left[\begin{array}{ccc} \underline{p}_1 & \cdots & \underline{p}_n \end{array}\right]\left(\left[\begin{array}{ccc} \underline{v}_{K1} & \cdots & \underline{v}_{Kn} \end{array}\right]\right)^{-1}} \tag{4.119}$$

where vectors $\underline{v}_{Ki}$ and $\underline{p}_i$ are given as in the non optimal pole assignment problem:

$$\boxed{\left\{\begin{array}{l} \underline{v}_{Ki} = \mathbf{N}_{ol}(\lambda_{Ki})\,\underline{\omega}_i \\ \underline{p}_i = D(\lambda_{Ki})\,\underline{\omega}_i \end{array}\right.} \tag{4.120}$$

[7] L.S. Shieh, H.M. Dib, R.E. Yates, Sequential design of linear quadratic state regulators via the optimal root-locus techniques, IEE Proceedings D - Control Theory and Applications, Volume: 135 , Issue: 4, July 1988, DOI: 10.1049/ip-d.1988.0040

It is worth noticing that if the open-loop eigenvalue λ_i is desired to be kept in the closed-loop, then applying (4.118) with relations $D(\lambda_i) = 0$ and $\lambda_i = \lambda_{Ki}$ imply that $\underline{p}_i = \underline{0}$ and that $\underline{v}_{Ki}$ shall be chosen such that $\mathbf{Q}\,\underline{v}_{Ki} = \underline{0}$. Indeed:

$$\boxed{\begin{array}{l} \left\{ \begin{array}{l} \left(D(-\lambda_{Ki})D(\lambda_{Ki})\mathbb{I} + \mathbf{R}^{-1}\left(\mathbf{N}_{ol}(-\lambda_{Ki})\right)^T \mathbf{Q}\,\mathbf{N}_{ol}(\lambda_{Ki})\right)\underline{\omega}_i = \underline{0} \\ D(\lambda_i) = 0 \\ \lambda_i = \lambda_{Ki} \end{array} \right. \\ \qquad \qquad \Rightarrow \left\{ \begin{array}{l} \underline{p}_i = D(\lambda_{Ki})\,\underline{\omega}_i = D(\lambda_i)\,\underline{\omega}_i = \underline{0} \\ \mathbf{Q}\,\mathbf{N}_{ol}(\lambda_i)\,\underline{\omega}_i = \mathbf{Q}\,\underline{v}_{Ki} = \underline{0} \end{array} \right. \end{array}} \tag{4.121}$$

On the other hand, if λ_{Ki} and $\underline{\omega}_i$ are set, then $\mathbf{Q}$ and $\mathbf{R}$ shall be chosen such that (4.118) holds $\forall\, i$. Once matrix $\mathbf{R} = \mathbf{R} > 0$ has been set, matrix $\mathbf{Q}$ can be assumed to be a real diagonal matrix whose coefficients q_i shall be computed to comply with (4.118):

$$\mathbf{Q} = \mathbf{Q}^T = \begin{bmatrix} q_1 & & \\ & \ddots & \\ & & q_n \end{bmatrix} \in \mathbb{R}^n \tag{4.122}$$

Nevertheless, take care that the computed matrix $\mathbf{Q} = \mathbf{Q}^T$ may *not* be positive semi-definite in that case.

Finally, for *single input* system, $\underline{\omega}_i := \omega_i \neq 0$ and $\mathbf{R} := R > 0$ are scalars and (4.118) reduces as follows:

$$\omega_i \neq 0 \in \mathbb{R} \Rightarrow D(-\lambda_{Ki})D(\lambda_{Ki}) + \left(\mathbf{N}_{ol}(-\lambda_{Ki})\right)^T \frac{\mathbf{Q}}{R}\mathbf{N}_{ol}(\lambda_{Ki}) = 0 \quad \forall\, i \tag{4.123}$$

Example 4.1. *We consider the following state equation:*

$$\dot{\underline{x}} = \begin{bmatrix} 0 & 1 \\ 10 & -9 \end{bmatrix}\underline{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}\underline{u} \tag{4.124}$$

We wish to design an optimal state feedback controller such that the closed-loop poles are located at $\{\lambda_{K1} = -10,\ \lambda_{K2} = -2\}$.

To solve this problem, we first observe that the eigenvalues of $\mathbf{A}$ *are* $\{\lambda_1 = -10,\ \lambda_2 = 1\}$. *Thus the problem consists in preserving* $\lambda_1 = -10$ *in the state feedback loop while shifting* $\lambda_2 = 1$ *towards* $\lambda_{K2} = -2$. *Because we are looking for an optimal state feedback controller, we have to select matrix* $\mathbf{Q} = \mathbf{Q}^T \geq 0$ *and* $R > 0$ *to achieve those specifications.*

The characteristic polynomial $D(s)$ *of state matrix* $\mathbf{A}$ *reads:*

$$D(s) = det(s\mathbb{I} - \mathbf{A}) = s^2 + 9s - 10 \Rightarrow \left\{ \begin{array}{l} D(\lambda_{K1}) = D(-10) = 0 \\ D(\lambda_{K2}) = D(-2) = -24 \end{array} \right. \tag{4.125}$$

In addition, let $\mathbf{N}_{ol}(s)$ *be the following polynomial matrix:*

$$\mathbf{N}_{ol}(s) := \text{adj}\,(s\mathbb{I} - \mathbf{A})\,\mathbf{B} = \begin{bmatrix} s+9 & 1 \\ 10 & s \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ s \end{bmatrix}$$
$$\Rightarrow \left\{ \begin{array}{l} \mathbf{N}_{ol}(\lambda_{K1}) = \mathbf{N}_{ol}(-10) = \begin{bmatrix} 1 & -10 \end{bmatrix}^T \\ \mathbf{N}_{ol}(-\lambda_{K1}) = \mathbf{N}_{ol}(10) = \begin{bmatrix} 1 & 10 \end{bmatrix}^T \\ \mathbf{N}_{ol}(\lambda_{K2}) = \mathbf{N}_{ol}(-2) = \begin{bmatrix} 1 & -2 \end{bmatrix}^T \\ \mathbf{N}_{ol}(-\lambda_{K2}) = \mathbf{N}_{ol}(2) = \begin{bmatrix} 1 & 2 \end{bmatrix}^T \end{array} \right. \tag{4.126}$$

Because we focus on a single input system, we use relation (4.123) *to select* $\mathbf{Q} = \mathbf{Q}^T \geq 0$ *and* $R > 0$*. Furthermore we will assume that* $\dfrac{\mathbf{Q}}{R}$ *is a diagonal matrix:*

$$\frac{\mathbf{Q}}{R} := \begin{bmatrix} q_{r1} & 0 \\ 0 & q_{r2} \end{bmatrix} \tag{4.127}$$

We get:

$$\begin{array}{l} D(-\lambda_{Ki})D(\lambda_{Ki}) + (\mathbf{N}_{ol}(-\lambda_{Ki}))^T \, \dfrac{\mathbf{Q}}{R} \mathbf{N}_{ol}(\lambda_{Ki}) = 0 \\ \Leftrightarrow \left\{ \begin{array}{l} q_{r1} - 100\, q_{r2} = 0 \\ q_{r1} - 4\, q_{r2} - 288 = 0 \end{array} \right. \Leftrightarrow \begin{bmatrix} 1 & -100 \\ 1 & -4 \end{bmatrix} \begin{bmatrix} q_{r1} \\ q_{r2} \end{bmatrix} = \begin{bmatrix} 0 \\ 288 \end{bmatrix} \end{array} \tag{4.128}$$

We finally get:

$$\begin{array}{l} \begin{bmatrix} q_{r1} \\ q_{r2} \end{bmatrix} = \begin{bmatrix} 1 & -100 \\ 1 & -4 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 288 \end{bmatrix} = \begin{bmatrix} 300 \\ 3 \end{bmatrix} \\ \Rightarrow \dfrac{\mathbf{Q}}{R} := \begin{bmatrix} q_{r1} & 0 \\ 0 & q_{r2} \end{bmatrix} = \begin{bmatrix} 300 & 0 \\ 0 & 3 \end{bmatrix} \end{array} \tag{4.129}$$

■

4.5 Poles assignment in optimal regulator through matrix inequalities

In this section a method for designing linear quadratic regulator with prescribed closed-loop pole is presented.

Let $\underline{\Lambda}_{cl} = \{\lambda_1, \lambda_2, \cdots, \lambda_n\}$ be a set of prescribed closed-loop eigenvalues, where $Re(\lambda_i) < 0$ and $\lambda_i \in \underline{\Lambda}_{cl}$ implies that the complex conjugate of λ_i, which is denoted λ_i^*, belongs also to $\underline{\Lambda}_{cl}$. The problem consists in finding a state feedback controller $u = -\mathbf{K}\underline{x}$ such that the eigenvalues of $\mathbf{A} - \mathbf{BK}$, which are denoted $\lambda\,(\mathbf{A} - \mathbf{BK})$, belongs to $\underline{\Lambda}_{cl}$:

$$\lambda\,(\mathbf{A} - \mathbf{BK}) = \underline{\Lambda}_{cl} \tag{4.130}$$

while minimizing the quadratic performance index $J(u(t))$ for some $\mathbf{Q} > 0$ and $\mathbf{R} > 0$.

$$J(u(t)) = \frac{1}{2}\int_0^\infty \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + u^T(t)\mathbf{R}u(t)\right)\,dt \tag{4.131}$$

We provide in that section the material written by He, Cai and Han[8]. Assume that $(\mathbf{A}, \mathbf{B})$ is controllable. Then, the pole assignment problem is solvable if and only if there exist two matrices $\mathbf{X}_1 \in \mathbb{R}^{n\times n}$ and $\mathbf{X}_2 \in \mathbb{R}^{n\times n}$ such that the following matrix inequalities are satisfied:

$$\left\{\begin{array}{l} \mathbf{F}^T\mathbf{X}_2^T\mathbf{X}_1 + \mathbf{X}_1^T\mathbf{X}_2\mathbf{F} + \mathbf{X}_2^T\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{X}_2 \leq 0 \\ \mathbf{X}_1^T\mathbf{X}_2 = \mathbf{X}_2^T\mathbf{X}_1 > 0 \end{array}\right. \tag{4.132}$$

where $\mathbf{F}$ is any matrix such that $\lambda(\mathbf{F}) = \underline{\Lambda}_{cl}$ and $(\mathbf{X}_1, \mathbf{X}_2)$ satisfies the following generalized Sylvester matrix equation[9]:

$$\mathbf{A}\mathbf{X}_1 - \mathbf{X}_1\mathbf{F} = \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{X}_2 \tag{4.133}$$

If $(\mathbf{X}_1, \mathbf{X}_2)$ is a feasible solution to the above two inequalities, then the weighting matrix $\mathbf{Q}$ in the quadratic performance index $J(u(t))$ can be chosen as follows:

$$\mathbf{Q} = -\mathbf{A}^T\mathbf{X}_2\mathbf{X}_1^{-1} - \mathbf{X}_2\mathbf{F}\mathbf{X}_1^{-1} \tag{4.134}$$

In addition the solution the corresponding Riccati Algebraic equation reads:

$$\mathbf{P} = \mathbf{X}_2\mathbf{X}_1^{-1} \tag{4.135}$$

The starting point to get this result is the fact that there must exist an eigenvector matrix $\mathbf{X}$ such that the following formula involving Hamiltonian matrix $\mathbf{H}$ holds:

$$\mathbf{H}\mathbf{X} = \mathbf{X}\mathbf{F} \tag{4.136}$$

Splitting the $2n \times n$ matrix $\mathbf{X}$ into 2 square $n \times n$ matrices $\mathbf{X}_1$ and $\mathbf{X}_2$ and using the expression of the $2n \times 2n$ Hamiltonian matrix $\mathbf{H}$ leads to the following relation:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix}\begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix}\mathbf{F} \tag{4.137}$$

The preceding relation is expanded as follows:

$$\left\{\begin{array}{l} \mathbf{A}\mathbf{X}_1 - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{X}_2 = \mathbf{X}_1\mathbf{F} \\ -\mathbf{Q}\mathbf{X}_1 - \mathbf{A}^T\mathbf{X}_2 = \mathbf{X}_2\mathbf{F} \end{array}\right. \Leftrightarrow \left\{\begin{array}{l} \mathbf{A}\mathbf{X}_1 - \mathbf{X}_1\mathbf{F} = \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{X}_2 \\ \mathbf{Q} = -\mathbf{A}^T\mathbf{X}_2\mathbf{X}_1^{-1} - \mathbf{X}_2\mathbf{F}\mathbf{X}_1^{-1} \end{array}\right. \tag{4.138}$$

[8]Hua-Feng He, Guang-Bin Cai and Xiao-Jun Han, Optimal Pole Assignment of Linear Systems by the Sylvester Matrix Equations, Hindawi Publishing Corporation, Abstract and Applied Analysis, Volume 2014, Article ID 301375, http://dx.doi.org/10.1155/2014/301375

[9]An explicit solution to right factorization with application in eigenstructure assignment, Bin Zhou, Guangren Duan, Journal of Control Theory and Applications 08/2005; 3(3):275-279. DOI: 10.1007/s11768-005-0049-7

Since $\mathbf{X}_1$ is nonsingular matrix $\mathbf{Q}$ is positive definite if and only if $\mathbf{X}_1^T\mathbf{Q}\mathbf{X}_1$ is positive definite. Using the first equation of (4.138) into the second one we get:

$$\begin{array}{ll} \mathbf{X}_1^T\mathbf{Q}\mathbf{X}_1 & = -\mathbf{X}_1^T\mathbf{A}^T\mathbf{X}_2 - \mathbf{X}_1^T\mathbf{X}_2\mathbf{F} \\ & = -(\mathbf{A}\mathbf{X}_1)^T\mathbf{X}_2 - \mathbf{X}_1^T\mathbf{X}_2\mathbf{F} \\ & = -(\mathbf{X}_1\mathbf{F} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{X}_2)^T\mathbf{X}_2 - \mathbf{X}_1^T\mathbf{X}_2\mathbf{F} \\ & = -(\mathbf{F}^T\mathbf{X}_2^T\mathbf{X}_1 + \mathbf{X}_1^T\mathbf{X}_2\mathbf{F} + \mathbf{X}_2^T\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{X}_2) \end{array} \tag{4.139}$$

We recall the Schur's formula:

$$\begin{array}{ll} \det\left(\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}\right) & = \det(\mathbf{A}_{22})\det(\mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21}) \\ & = \det(\mathbf{A}_{11})\det(\mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}) \end{array} \tag{4.140}$$

Using the Schur's formula and denoting $\mathbf{S} = \mathbf{X}_1^T\mathbf{X}_2$ we finally get:

$$\mathbf{X}_1^T\mathbf{Q}\mathbf{X}_1 \geq 0 \Leftrightarrow \begin{bmatrix} \mathbf{F}^T\mathbf{S}^T + \mathbf{S}\mathbf{F} & \mathbf{X}_2^T\mathbf{B} \\ \mathbf{B}^T\mathbf{X}_2 & -\mathbf{R}^{-1} \end{bmatrix} \leq 0 \tag{4.141}$$

4.6 Model matching

4.6.1 Cross-term in the performance index

Assume that the output $\underline{z}(t)$ of interest is expressed as a linear combination of state vector $\underline{x}(t)$ and control $\underline{u}(t)$: $\underline{z}(t) = \mathbf{N}\,\underline{x}(t) + \mathbf{D}\,u(t)$. Thus the cost to be minimized reads:

$$\begin{array}{l} \left\{ \begin{array}{l} J(\underline{u}(t)) = \frac{1}{2}\int_0^\infty \underline{z}^T(t)\underline{z}(t) + \underline{u}^T(t)\mathbf{R}_1\underline{u}(t)\,dt \\ \underline{z}(t) = \mathbf{N}\,\underline{x}(t) + \mathbf{D}\,\underline{u}(t) \\ \mathbf{R}_1 = \mathbf{R}_1^T > 0 \end{array} \right. \\ \Rightarrow J(\underline{u}(t)) = \frac{1}{2}\int_0^\infty \left(\underline{x}^T(t)\mathbf{N}^T + \underline{u}^T(t)\mathbf{D}^T\right)(\mathbf{N}\,\underline{x}(t) + \mathbf{D}\,\underline{u}(t)) + \underline{u}^T(t)\mathbf{R}_1\underline{u}(t)\,dt \end{array} \tag{4.142}$$

Then we get a more general form of the quadratic performance index. Indeed the quadratic performance index can be rewritten as:

$$\begin{array}{ll} J(u(t)) & = \frac{1}{2}\int_0^\infty \begin{bmatrix} \underline{x} \\ \underline{u} \end{bmatrix}^T \begin{bmatrix} \mathbf{Q} & \mathbf{S} \\ \mathbf{S}^T & \mathbf{R} \end{bmatrix} \begin{bmatrix} \underline{x} \\ \underline{u} \end{bmatrix} dt \\ & = \frac{1}{2}\int_0^\infty \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right) + 2\underline{x}^T(t)\mathbf{S}\underline{u}(t)\,dt \end{array} \tag{4.143}$$

where:

$$\left\{ \begin{array}{l} \mathbf{Q} = \mathbf{Q}^T := \mathbf{N}^T\mathbf{N} \geq 0 \\ \mathbf{R} = \mathbf{R}^T := \mathbf{D}^T\mathbf{D} + \mathbf{R}_1 > 0 \\ \mathbf{S} := \mathbf{N}^T\mathbf{D} \end{array} \right. \tag{4.144}$$

It can be seen that:

$$\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t) + 2\underline{x}^T(t)\mathbf{S}\underline{u}(t) = \underline{x}^T(t)\mathbf{Q}_m\underline{x}(t) + \underline{v}^T(t)\mathbf{R}\underline{v}(t) \tag{4.145}$$

where:

$$\left\{ \begin{array}{l} \mathbf{Q}_m = \mathbf{Q} - \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T \\ \underline{v}(t) = \underline{u}(t) + \mathbf{R}^{-1}\mathbf{S}^T\underline{x}(t) \end{array} \right. \tag{4.146}$$

Hence cost (4.143) can be rewritten as:

$$J(u(t)) = \frac{1}{2}\int_0^{\infty} \underline{x}^T(t)\mathbf{Q}_m\underline{x}(t) + \underline{v}^T(t)\mathbf{R}\underline{v}(t)\,dt \tag{4.147}$$

Moreover the plant dynamics $\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)$ is modified as follows:

$$\begin{array}{rl} \dot{\underline{x}} & = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ & = \mathbf{A}\underline{x}(t) + \mathbf{B}\left(\underline{v}(t) - \mathbf{R}^{-1}\mathbf{S}^T\underline{x}(t)\right) \\ & = \mathbf{A}_m\underline{x}(t) + \mathbf{B}\underline{v}(t) \\ \text{where} & \mathbf{A}_m = \mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^T \end{array} \tag{4.148}$$

Assuming that $\mathbf{Q}_m$ (which is symmetric) is positive definite, we then get a standard LQR problem for which the optimal state feedback control law is given from (3.9):

$$\left\{ \begin{array}{l} \underline{v}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}\underline{x}(t) \\ \underline{v}(t) = \underline{u}(t) + \mathbf{R}^{-1}\mathbf{S}^T\underline{x}(t) \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \underline{u}(t) = -\mathbf{K}\underline{x}(t) \\ \mathbf{K} = \mathbf{R}^{-1}\left(\mathbf{P}\mathbf{B} + \mathbf{S}\right)^T \end{array} \right. \tag{4.149}$$

Where matrix $\mathbf{P}$ is the positive definite matrix which solves the following *algebraic Riccati equation* (see (3.8)):

$$\mathbf{P}\mathbf{A}_m + \mathbf{A}_m^T\mathbf{P} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q}_m = \mathbf{0} \tag{4.150}$$

It is worth noticing that robustness properties of the LQ state feedback are lost if the cost to be minimized contains a state-control cross-term as it is the case here.

4.6.2 Implicit reference model

Let $\mathbf{A}_r$ be the desired closed-loop state matrix of the system and $\underline{e}(t)$ be the following error vector:

$$\underline{e}(t) := \dot{\underline{x}}(t) - \mathbf{A}_r\underline{x}(t) \tag{4.151}$$

In that section we consider the problem to find control $u(t)$ which minimizes the following performance index:

$$\begin{array}{rl} J(\underline{u}(t)) & = \frac{1}{2}\int_0^{\infty} \underline{e}^T(t)\underline{e}(t)\,dt \\ & = \frac{1}{2}\int_0^{\infty} \left(\dot{\underline{x}}(t) - \mathbf{A}_r\underline{x}(t)\right)^T \left(\dot{\underline{x}}(t) - \mathbf{A}_r\underline{x}(t)\right)\,dt \end{array} \tag{4.152}$$

Expanding $\dot{\underline{x}}(t)$ we get:

$$\begin{array}{rl} J(\underline{u}(t)) & = \frac{1}{2}\int_0^{\infty} \left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) - \mathbf{A}_r\underline{x}(t)\right)^T \left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) - \mathbf{A}_r\underline{x}(t)\right)\,dt \\ & = \frac{1}{2}\int_0^{\infty} \left(\left(\mathbf{A} - \mathbf{A}_r\right)\underline{x}(t) + \mathbf{B}\underline{u}(t)\right)^T \left(\left(\mathbf{A} - \mathbf{A}_r\right)\underline{x}(t) + \mathbf{B}\underline{u}(t)\right)\,dt \end{array} \tag{4.153}$$

We get a cost to be minimized which contains a state-control cross-term:

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^{\infty} \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right) + 2\underline{x}^T(t)\mathbf{N}\underline{u}(t)\,dt \tag{4.154}$$

Where:

$$\begin{cases} \mathbf{Q} = (\mathbf{A} - \mathbf{A}_r)^T (\mathbf{A} - \mathbf{A}_r) \\ \mathbf{R} = \mathbf{B}^T\mathbf{B} \\ \mathbf{N} = (\mathbf{A} - \mathbf{A}_r)^T \mathbf{B} \end{cases} \tag{4.155}$$

Then we can re-use the results of section 4.6.1. Let $\mathbf{P}$ be the positive definite matrix which solves the following *algebraic Riccati* equation:

$$\mathbf{P}\mathbf{A}_m + \mathbf{A}_m^T\mathbf{P} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q}_m = \mathbf{0} \tag{4.156}$$

Where:

$$\begin{cases} \mathbf{Q}_m = \mathbf{Q} - \mathbf{N}\mathbf{R}^{-1}\mathbf{N}^T \\ \mathbf{A}_m = \mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{N}^T \\ \underline{v}(t) = \underline{u}(t) + \mathbf{R}^{-1}\mathbf{N}^T\underline{x}(t) \end{cases} \tag{4.157}$$

The stabilizing control $u(t)$ is then defined in a similar fashion than (4.149):

$$\begin{cases} \underline{v}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}\underline{x}(t) \\ \underline{v}(t) = u(t) + \mathbf{R}^{-1}\mathbf{N}^T\underline{x}(t) \end{cases} \Rightarrow \begin{cases} \underline{u}(t) = -\mathbf{K}\underline{x}(t) \\ \mathbf{K} = \mathbf{R}^{-1}(\mathbf{P}\mathbf{B} + \mathbf{N})^T \end{cases} \tag{4.158}$$

It is worth noticing that robustness properties of the LQ state feedback are lost because the cost to be minimized contains a state-control cross-term here.

Furthermore let $\mathbf{V}$ be the change of basis matrix to the Jordan form $\mathbf{\Lambda}_r$ of the desired closed-loop state matrix $\mathbf{A}_r$:

$$\mathbf{\Lambda}_r = \mathbf{V}^{-1}\mathbf{A}_r\mathbf{V} \tag{4.159}$$

Let $\mathbf{A}_{cl}$ be the state matrix of the closed-loop which is written using matrix $\mathbf{V}$ as follows :

$$\dot{\underline{x}}(t) = \mathbf{A}_{cl}\underline{x}(t) = \mathbf{V}\mathbf{\Lambda}_{cl}\mathbf{V}^{-1}\underline{x}(t) \tag{4.160}$$

Assuming that the desired Jordan form $\mathbf{\Lambda}_r$ is a diagonal matrix and using the fact $\mathbf{V}^{-1} = \mathbf{V}^T$ the product $e^T(t)e(t)$ in (4.152) reads as follows:

$$\begin{array}{ll} \underline{e}^T(t)\underline{e}(t) & = \underline{x}^T(t)(\mathbf{A}_{cl} - \mathbf{A}_r)^T(\mathbf{A}_{cl} - \mathbf{A}_r)\underline{x}(t) \\ & = \underline{x}^T(t)\mathbf{V}(\mathbf{\Lambda}_{cl} - \mathbf{\Lambda}_r)^T(\mathbf{\Lambda}_{cl} - \mathbf{\Lambda}_r)\mathbf{V}^T\underline{x}(t) \end{array} \tag{4.161}$$

From the preceding equation it is clear that minimizing the cost $J(\underline{u}(t)) = \frac{1}{2}\int_0^\infty \underline{e}^T(t)\underline{e}(t)\,dt$ consists in finding the control $\underline{u}(t)$ which minimizes the gap between the desired eigenvalues (which are set in $\mathbf{\Lambda}_r$) and the actual eigenvalues of the closed-loop.

4.7 Optimal output feedback

4.7.1 Reformulation of the state feedback optimal control problem

We consider in this section the following plant:

$$\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \tag{4.162}$$

We which minimizes the following performance index J where $\mathbf{Q} = \mathbf{Q}^T \geq 0$ and $\mathbf{R} = \mathbf{R}^T > 0$:

$$J = \int_0^\infty \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t) \right) \, dt \tag{4.163}$$

We will assume that we seek for a stabilizing static *state* feedback gain $\mathbf{K}$:

$$\underline{u}(t) = -\mathbf{K}\underline{x}(t) \tag{4.164}$$

Using the relation $\underline{u}(t) = -\mathbf{K}\underline{x}(t)$, the performance index J reads:

$$\begin{array}{ll} J & = \int_0^\infty \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + (\mathbf{K}\underline{x}(t))^T \mathbf{R}\mathbf{K}\underline{x}(t) \right) \, dt \\ & = \int_0^\infty \underline{x}^T(t) \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) \underline{x}(t) \, dt \end{array} \tag{4.165}$$

Moreover, the dynamics of the plant where $\underline{u}(t) = -\mathbf{K}\underline{x}(t)$ reads:

$$\underline{u}(t) = -\mathbf{K}\underline{x}(t) \Rightarrow \dot{\underline{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K}) \, \underline{x}(t) \tag{4.166}$$

Thus, after integration, and denoting by $\underline{x}(0)$ the initial value of $\underline{x}(t)$, we get:

$$\underline{x}(t) = e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t}\underline{x}(0) \tag{4.167}$$

Consequently, the performance index J reads:

$$\begin{array}{ll} J & = \int_0^\infty \underline{x}^T(t) \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) \underline{x}(t) \, dt \\ & = \int_0^\infty \underline{x}(0)^T e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,t} \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t}\underline{x}(0) \, dt \end{array} \tag{4.168}$$

Using the property of trace, i.e. $\mathrm{tr}\,(\mathbf{XYZ}) = \mathrm{tr}\,(\mathbf{YZX})$, the performance index J is written as follows:

$$\begin{array}{ll} J & = \int_0^\infty \underline{x}(0)^T e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,t} \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t}\underline{x}(0) \, dt \\ & = \mathrm{tr}\left(\int_0^\infty \underline{x}(0)^T e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,t} \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t}\underline{x}(0) \, dt \right) \\ & = \mathrm{tr}\left(\int_0^\infty e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,t} \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t}\underline{x}(0)\underline{x}(0)^T \, dt \right) \end{array} \tag{4.169}$$

Because the initial value $\underline{x}(0)$ of the state vector is usually unknown, the product $\underline{x}(0)\underline{x}(0)^T$ is removed from the performance index J. We get:

$$J = \mathrm{tr}\,(\mathbf{P}) \tag{4.170}$$

where matrix $\mathbf{P}$ is the so-called grammian:

$$\mathbf{P} = \int_0^\infty e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,t} \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t} \, dt = \mathbf{P}^T > 0 \tag{4.171}$$

When multiplying $\mathbf{P}$ by $e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,t_0}$ on the left and by $e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t_0}$ on the right, we get $\forall\ t_0 \in \mathbb{R}$:

$$\begin{array}{l} e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,t_0}\,\mathbf{P}\,e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t_0} \\ \qquad = \int_0^\infty e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,(t+t_0)} \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,(t+t_0)} \, dt \\ \qquad = \int_{t_0}^\infty e^{(\mathbf{A}-\mathbf{B}\mathbf{K})^T\,t} \left(\mathbf{Q} + \mathbf{K}^T\mathbf{R}\mathbf{K} \right) e^{(\mathbf{A}-\mathbf{B}\mathbf{K})\,t} \, dt \end{array} \tag{4.172}$$

Differentiation with respect to t_0 and using the facts that matrix $\mathbf{A} - \mathbf{BK}$ is assumed to be stable (that is $\lim_{t\to\infty} e^{(\mathbf{A}-\mathbf{BK})\,t} = \lim_{t\to\infty} e^{(\mathbf{A}-\mathbf{BK})^T\,t} = \mathbf{0}$) and that $g(x) = \int_{a(x)}^{b(x)} f(\tau)\,d\tau \Rightarrow g'(x) = f(b(x))\,b'(x) - f(a(x))\,a'(x)$ yields:

$$\begin{array}{l} (\mathbf{A} - \mathbf{BK})^T\, e^{(\mathbf{A}-\mathbf{BK})^T\,t_0}\,\mathbf{P}\,e^{(\mathbf{A}-\mathbf{BK})\,t_0} \\ \qquad + e^{(\mathbf{A}-\mathbf{BK})^T\,t_0}\,\mathbf{P}\,e^{(\mathbf{A}-\mathbf{BK})\,t_0}\,(\mathbf{A} - \mathbf{BK}) \\ \qquad\qquad = -e^{(\mathbf{A}-\mathbf{BK})^T\,t_0}\left(\mathbf{Q} + \mathbf{K}^T\mathbf{RK}\right)e^{(\mathbf{A}-\mathbf{BK})\,t_0} \end{array} \tag{4.173}$$

Finally setting $t_0 = 0$ leads to the following Lyapunov equation:

$$(\mathbf{A} - \mathbf{BK})^T\;\mathbf{P} + \mathbf{P}\;(\mathbf{A} - \mathbf{BK}) = -\left(\mathbf{Q} + \mathbf{K}^T\mathbf{RK}\right) \tag{4.174}$$

Alternatively, and following Lewis & al.[10], assume that there exists a positive definite matrix $\mathbf{P} = \mathbf{P}^T > 0$ such that the following equality holds:

$$\frac{d}{dt}\left(\underline{x}^T(t)\mathbf{P}\underline{x}(t)\right) = -\underline{x}^T(t)\;\left(\mathbf{Q} + \mathbf{K}^T\mathbf{RK}\right)\;\underline{x}(t) \tag{4.175}$$

Then the performance index J defined in (4.165) reads:

$$\begin{array}{ll} J & = \int_0^\infty \underline{x}^T(t)\left(\mathbf{Q} + \mathbf{K}^T\mathbf{RK}\right)\underline{x}(t)\,dt \\ & = -\int_0^\infty \frac{d}{dt}\left(\underline{x}^T(t)\mathbf{P}\underline{x}(t)\right)\,dt \\ & = -\left.\underline{x}^T(t)\mathbf{P}\underline{x}(t)\right|_{t=0}^{t\to\infty} \\ & = \underline{x}^T(0)\,\mathbf{P}\,\underline{x}(0) - \lim_{t\to\infty}\underline{x}^T(t)\,\mathbf{P}\,\underline{x}(t) \end{array} \tag{4.176}$$

Assuming that the closed-loop is stable so that $\underline{x}(t)$ vanishes with time, we get the following relation where $\mathrm{tr}\,(\mathbf{X})$ denotes the trace of matrix $\mathbf{X}$:

$$\lim_{t\to\infty}\underline{x}(t) = \underline{0} \Rightarrow J = \underline{x}^T(0)\,\mathbf{P}\,\underline{x}(0) = \mathrm{tr}\left(\mathbf{P}\,\underline{x}(0)\,\underline{x}^T(0)\right) \tag{4.177}$$

Furthermore, when using (4.162) and (4.163) in (4.175), we can write:

$$\begin{array}{ll} -\underline{x}^T(t) & \left(\mathbf{Q} + \mathbf{K}^T\mathbf{RK}\right)\underline{x}(t) = \frac{d}{dt}\left(\underline{x}^T(t)\mathbf{P}\underline{x}(t)\right) \\ & = \dot{\underline{x}}^T(t)\mathbf{P}\underline{x}(t) + \underline{x}^T(t)\mathbf{P}\dot{\underline{x}}(t) \\ & = (\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t))^T\,\mathbf{P}\underline{x}(t) + \underline{x}^T(t)\mathbf{P}\,(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)) \\ & = ((\mathbf{A} - \mathbf{BK})\,\underline{x}(t))^T\,\mathbf{P}\underline{x}(t) + \underline{x}^T(t)\mathbf{P}\,((\mathbf{A} - \mathbf{BK})\,\underline{x}(t)) \\ & = \underline{x}^T(t)\left((\mathbf{A} - \mathbf{BK})^T\,\mathbf{P} + \mathbf{P}\,(\mathbf{A} - \mathbf{BK})\right)\underline{x}(t) \end{array} \tag{4.178}$$

Since this relation shall hold for all value of $\underline{x}(t)$, we shall have:

$$-\left(\mathbf{Q} + \mathbf{K}^T\mathbf{RK}\right) = (\mathbf{A} - \mathbf{BK})^T\,\mathbf{P} + \mathbf{P}\,(\mathbf{A} - \mathbf{BK}) \tag{4.179}$$

We retrieve the Lyapunov equation (4.174).

Consequently, the *dynamic* optimization control problem can be converted into the following equivalent *static* optimization control problem:

$$\boxed{\begin{array}{c} \text{Find } \mathbf{P} = \mathbf{P}^T > 0 \text{ and } \mathbf{K} \\ \text{which minimizes } \mathrm{tr}\,(\mathbf{P}) \\ \text{under the constraint } \mathbf{A}_{cl}^T\mathbf{P} + \mathbf{P}\mathbf{A}_{cl} + \left(\mathbf{Q} + \mathbf{K}^T\mathbf{RK}\right) = \mathbf{0} \\ \text{where } \mathbf{A}_{cl} := \mathbf{A} - \mathbf{BK} \end{array}} \tag{4.180}$$

[10] Lewis F., Vrabie D., Syrmos V., Optimal Control, John Wiley & Sons, 3rd Edition, 2012

The constraint to be satisfied is simply the algebraic Riccati equation. Indeed by adding and subtracting terms $\mathbf{PBK}$ and $(\mathbf{BK})^T\mathbf{P}$ within the algebraic Riccati equation we get:

$$\begin{array}{l}
\mathbf{A}^T\mathbf{P} + \mathbf{PA} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \\
\Leftrightarrow (\mathbf{A} - \mathbf{BK})^T\mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BK}) + (\mathbf{BK})^T\mathbf{P} + \mathbf{PBK} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \\
\Leftrightarrow (\mathbf{A} - \mathbf{BK})^T\mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BK}) + \mathbf{K}^T\mathbf{B}^T\mathbf{P} + \mathbf{PBK} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \\
\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \Rightarrow (\mathbf{A} - \mathbf{BK})^T\mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BK}) + \mathbf{K}^T\mathbf{B}^T\mathbf{P} + \cancel{\mathbf{PBK}} - \cancel{\mathbf{PBK}} + \mathbf{Q} = \mathbf{0} \\
\mathbf{B}^T\mathbf{P} = \mathbf{RK} \Rightarrow (\mathbf{A} - \mathbf{BK})^T\mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BK}) + \left(\mathbf{Q} + \mathbf{K}^T\mathbf{RK}\right) = \mathbf{0}
\end{array} \tag{4.181}$$

4.7.2 Output feedback optimal control problem

The preceding result can be extended to *output* feedback optimal control problem. Indeed, consider the following plant:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\,\underline{x}(t) \end{cases} \tag{4.182}$$

We wish to find a stabilizing static *output* feedback gain $\mathbf{K}$, $\underline{u}(t) = -\mathbf{K}\underline{y}(t)$, which minimizes the following performance index J where $\mathbf{Q} = \mathbf{Q}^T \geq 0$ and $\mathbf{R} = \mathbf{R}^T > 0$:

$$\begin{cases} J = \int_0^\infty \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right) dt \\ \underline{u}(t) = -\mathbf{K}\underline{y}(t) = -\mathbf{KC}\underline{x}(t) \end{cases} \tag{4.183}$$

Let:

$$\boxed{\mathbf{Q}_K = \mathbf{Q} + \mathbf{C}^T\mathbf{K}^T\mathbf{RKC} = \mathbf{Q}_K^T \geq 0} \tag{4.184}$$

Then the performance index J reads:

$$\begin{array}{rl} J & = \int_0^\infty \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + (\mathbf{KC}\underline{x}(t))^T\mathbf{RKC}\underline{x}(t)\right) dt \\ & = \int_0^\infty \underline{x}^T(t)\left(\mathbf{Q} + \mathbf{C}^T\mathbf{K}^T\mathbf{RKC}\right)\underline{x}(t)\,dt \\ & = \int_0^\infty \underline{x}^T(t)\,\mathbf{Q}_K\,\underline{x}(t)\,dt \end{array} \tag{4.185}$$

Following Lewis & al.[10], assume that there exists a positive definite matrix $\mathbf{P} = \mathbf{P}^T > 0$ such that the following equality holds:

$$\frac{d}{dt}\left(\underline{x}^T(t)\mathbf{P}\underline{x}(t)\right) = -\underline{x}^T(t)\,\mathbf{Q}_K\,\underline{x}(t) \tag{4.186}$$

The performance index J now reads:

$$\begin{array}{rl} J & = -\int_0^\infty \frac{d}{dt}\left(\underline{x}^T(t)\mathbf{P}\underline{x}(t)\right) dt \\ & = -\left.\underline{x}^T(t)\mathbf{P}\underline{x}(t)\right|_{t=0}^{t\to\infty} \\ & = \underline{x}^T(0)\,\mathbf{P}\,\underline{x}(0) - \lim_{t\to\infty}\underline{x}^T(t)\,\mathbf{P}\,\underline{x}(t) \end{array} \tag{4.187}$$

Assuming that the closed-loop is stable so that $\underline{x}(t)$ vanishes with time, we get the following relation where $\mathrm{tr}\,(\mathbf{X})$ denotes the trace of matrix $\mathbf{X}$:

$$\lim_{t\to\infty}\underline{x}(t) = \underline{0} \Rightarrow J = \underline{x}^T(0)\,\mathbf{P}\,\underline{x}(0) = \mathrm{tr}\left(\mathbf{P}\,\underline{x}(0)\,\underline{x}^T(0)\right) \tag{4.188}$$

Furthermore, when using (4.182) and (4.183) in (4.186), we can write:

$$\begin{aligned}
-\underline{x}^T(t)\,\mathbf{Q}_K\,\underline{x}(t) &= \tfrac{d}{dt}\left(\underline{x}^T(t)\mathbf{P}\underline{x}(t)\right) \\
&= \dot{\underline{x}}^T(t)\mathbf{P}\underline{x}(t) + \underline{x}^T(t)\mathbf{P}\dot{\underline{x}}(t) \\
&= \left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right)^T \mathbf{P}\underline{x}(t) + \underline{x}^T(t)\mathbf{P}\left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right) \\
&= \left(\left(\mathbf{A} - \mathbf{BKC}\right)\underline{x}(t)\right)^T \mathbf{P}\underline{x}(t) + \underline{x}^T(t)\mathbf{P}\left(\left(\mathbf{A} - \mathbf{BKC}\right)\underline{x}(t)\right) \\
&= \underline{x}^T(t)\left(\left(\mathbf{A} - \mathbf{BKC}\right)^T\mathbf{P} + \mathbf{P}\left(\mathbf{A} - \mathbf{BKC}\right)\right)\underline{x}(t)
\end{aligned} \tag{4.189}$$

Since this relation shall hold for all value of $\underline{x}(t)$, we shall have:

$$-\mathbf{Q}_K = \left(\mathbf{A} - \mathbf{BKC}\right)^T\mathbf{P} + \mathbf{P}\left(\mathbf{A} - \mathbf{BKC}\right) \tag{4.190}$$

Consequently, the *dynamical* optimization control problem can be converted into the following equivalent *static* optimization control problem:

$$\boxed{\begin{array}{c}
\text{Find } \mathbf{P} = \mathbf{P}^T > 0 \text{ and } \mathbf{K} \\
\text{which minimizes } \operatorname{tr}(\mathbf{P}) \\
\text{under the constraint } \left(\mathbf{A} - \mathbf{BKC}\right)^T\mathbf{P} + \mathbf{P}\left(\mathbf{A} - \mathbf{BKC}\right) + \mathbf{Q}_K = \mathbf{0}
\end{array}} \tag{4.191}$$

4.7.3 Solution of the output feedback optimal control problem

First, we recall the following property: let $\mathbf{XYZ}$ be a square matrix where matrices $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ are of appropriate dimension. Then the following properties hold[10]:

$$\left\{\begin{array}{l}
\frac{\partial\operatorname{tr}(\mathbf{XY})}{\partial\mathbf{Y}} = \mathbf{X}^T \\
\frac{\partial\operatorname{tr}(\mathbf{XYZ})}{\partial\mathbf{Y}} = \mathbf{X}^T\mathbf{Z}^T \\
\frac{\partial\operatorname{tr}(\mathbf{XY}^T\mathbf{Z})}{\partial\mathbf{Y}} = \frac{\partial\operatorname{tr}(\mathbf{Z}^T\mathbf{YX}^T)}{\partial\mathbf{Y}} = \mathbf{ZX}
\end{array}\right. \tag{4.192}$$

Example 4.2. *In order to illustrate the first relation, we consider the following matrices:*

$$\left\{\begin{array}{l}
\mathbf{X} = \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix} \\
\mathbf{Y} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix}
\end{array}\right. \tag{4.193}$$

Then:

$$\begin{array}{l}
\operatorname{tr}(\mathbf{XY}) = \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix}\begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} = \begin{bmatrix} 2y_{11} + 3y_{21} & 2y_{12} + 3y_{22} \\ 4y_{11} + 5y_{21} & 4y_{12} + 5y_{22} \end{bmatrix} \\
\Rightarrow \operatorname{tr}(\mathbf{XY}) = 2y_{11} + 3y_{21} + 4y_{12} + 5y_{22}
\end{array} \tag{4.194}$$

Thus:

$$\frac{\partial\operatorname{tr}(\mathbf{XY})}{\partial\mathbf{Y}} = \begin{bmatrix} \frac{\partial\operatorname{tr}(\mathbf{XY})}{\partial y_{11}} & \frac{\partial\operatorname{tr}(\mathbf{XY})}{\partial y_{12}} \\ \frac{\partial\operatorname{tr}(\mathbf{XY})}{\partial y_{21}} & \frac{\partial\operatorname{tr}(\mathbf{XY})}{\partial y_{22}} \end{bmatrix} = \begin{bmatrix} 2 & 4 \\ 3 & 5 \end{bmatrix} = \mathbf{X}^T \tag{4.195}$$

■

Now, we are in position to solve the output feedback optimal control problem thanks to the Lagrange multiplier approach. We define the (scalar) Hamiltonian $\mathcal{H}$ as follows where $\mathbf{\Lambda} = \mathbf{\Lambda}^T$ is a $n \times n$ diagonal matrix of Lagrange multipliers to be determined:

$$\mathcal{H} = \operatorname{tr}(\mathbf{P}) + \operatorname{tr}\left(\mathbf{\Lambda}\left((\mathbf{A} - \mathbf{BKC})^T \mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BKC}) + \mathbf{Q}_K\right)\right) \tag{4.196}$$

The necessary conditions to solve this optimization problem with respect to matrices $\mathbf{K}$, $\mathbf{\Lambda}$ and $\mathbf{P}$ read as follows:

$$\begin{cases} \frac{\partial \mathcal{H}}{\partial \mathbf{K}} = \mathbf{0} \Rightarrow 2\left(\mathbf{RKC\Lambda C}^T - \mathbf{B}^T\mathbf{P\Lambda C}^T\right) = \mathbf{0} \\ \frac{\partial \mathcal{H}}{\partial \mathbf{\Lambda}} = \mathbf{0} \Rightarrow (\mathbf{A} - \mathbf{BKC})^T \mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BKC}) + \mathbf{Q}_K = \mathbf{0} \\ \frac{\partial \mathcal{H}}{\partial \mathbf{P}} = \mathbf{0} \Rightarrow \mathbb{I} + (\mathbf{A} - \mathbf{BKC})^T \mathbf{\Lambda} + \mathbf{\Lambda}(\mathbf{A} - \mathbf{BKC}) = \mathbf{0} \end{cases} \tag{4.197}$$

From the first equation, and assuming that $\mathbf{C\Lambda C}^T$ is nonsingular, the static *output* feedback gain $\mathbf{K}$ can be computed as a function of Lagrange multipliers $\mathbf{\Lambda}$:

$$\frac{\partial \mathcal{H}}{\partial \mathbf{K}} = \mathbf{0} \Rightarrow \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P\Lambda C}^T\left(\mathbf{C\Lambda C}^T\right)^{-1} \tag{4.198}$$

It is worth noticing that for the static *state* feedback case where $\mathbf{C} = \mathbb{I}$, the static *state* feedback $\mathbf{K}$ no more depends of Lagrange multipliers $\mathbf{\Lambda}$:

$$\mathbf{C} = \mathbb{I} \Rightarrow \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \tag{4.199}$$

Moreover, in the *state* feedback case the Lyapunov equation specifying the constraint turns to be the algebraic Riccati equation:

$$\begin{array}{rl} \mathbf{C} = \mathbb{I} & \Rightarrow \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \Rightarrow \mathbf{Q}_K = \mathbf{Q} + \mathbf{K}^T\mathbf{RK} = \mathbf{Q} + \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} \\ \Rightarrow \mathbf{0} & = (\mathbf{A} - \mathbf{BKC})^T \mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BKC}) + \mathbf{Q}_K \\ & = (\mathbf{A} - \mathbf{BK})^T \mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BK}) + \mathbf{Q} + \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} \\ & = \mathbf{A}^T\mathbf{P} + \mathbf{PA} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} \end{array} \tag{4.200}$$

4.7.4 Poles placement in a specified region

In this section, we still wish to find a stabilizing static *output* feedback gain $\mathbf{K}$, $\underline{u}(t) = -\mathbf{K}\underline{y}(t)$, which minimizes the performance index J defined in (4.183). We add the constraint that closed-loop poles are situated in the sector region $C(\alpha, \theta)$ shown in Figure 4.3.

Let $\mathbf{Q}_K$ be defined as in (4.184) and let $\mathbf{A}_{cl}$ be the closed-loop state matrix:

$$\mathbf{A}_{cl} = \mathbf{A} - \mathbf{BKC} \tag{4.201}$$

We have seen in the previous section that if there exists a positive definite matrix $\mathbf{P} = \mathbf{P}^T > 0$ such that the following equality holds:

$$\mathbf{A}_{cl}^T\mathbf{P} + \mathbf{PA}_{cl} + \mathbf{Q}_K = \mathbf{0} \tag{4.202}$$

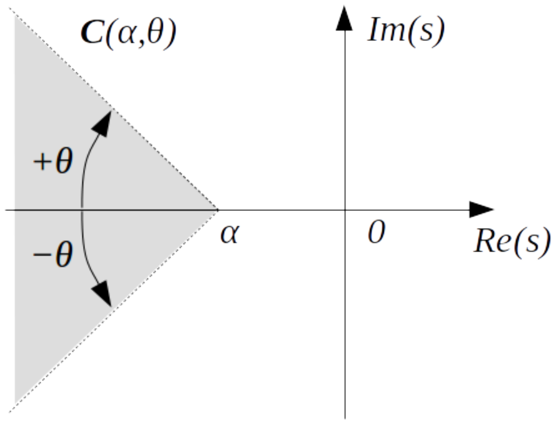


Figure 4.3: Sector region $C(\alpha, \theta)$

Then the minimum cost J^* reads as follows where $\text{tr}\,(\mathbf{X})$ denotes the trace of matrix $\mathbf{X}$:

$$J^* = \underline{x}^T(0)\,\mathbf{P}\,\underline{x}(0) = \text{tr}\left(\mathbf{P}\,\underline{x}(0)\,\underline{x}^T(0)\right) \tag{4.203}$$

Following Yuan & al.[11], let matrix $\widetilde{\mathbf{A}}$ be defined as follows, where $\otimes$ denotes the Kronecker product:

$$\left\{ \begin{array}{l} \mathbf{A}_\alpha = \mathbf{A} - \alpha\mathbb{I} \qquad \text{where } \alpha \leq 0 \\ \widetilde{\mathbf{A}} = \begin{bmatrix} \sin(\theta)\mathbf{A}_\alpha & \cos(\theta)\mathbf{A}_\alpha \\ -\cos(\theta)\mathbf{A}_\alpha & \sin(\theta)\mathbf{A}_\alpha \end{bmatrix} := \begin{bmatrix} \sin(\theta) & \cos(\theta) \\ -\cos(\theta) & \sin(\theta) \end{bmatrix} \otimes \mathbf{A}_\alpha \end{array} \right. \tag{4.204}$$

Then it can be shown[11] that all the eigenvalues of matrix $\mathbf{A}$ will be situated inside the sector region $C(\alpha, \theta)$ shown in Figure 4.3 if and only if all the eigenvalues of matrix $\widetilde{\mathbf{A}}$ are situated in the left half plane. Denoting by $\Psi(s)$ the polynomial whose companion matrix is $\widetilde{\mathbf{A}}$, this can be tested for example with the Routh-Hurwitz criterion.

This result can be extended to the case where we add the constraint that all the eigenvalues of state matrix $\mathbf{A}$ shall have a real part greater than α_m. This will be the case if and only if all the eigenvalues of the following block diagonal matrix $\widetilde{\mathbf{A}}_e$ are situated in the left half plane:

$$\widetilde{\mathbf{A}}_e = \begin{bmatrix} \widetilde{\mathbf{A}} & \mathbf{0} \\ \mathbf{0} & \alpha_m\mathbb{I} - \mathbf{A} \end{bmatrix} \tag{4.205}$$

Moreover let:

$$\left\{ \begin{array}{l} \begin{bmatrix} \widetilde{\mathbf{B}}_1 & \widetilde{\mathbf{B}}_2 \end{bmatrix} := \begin{bmatrix} \sin(\theta) & \cos(\theta) \\ -\cos(\theta) & \sin(\theta) \end{bmatrix} \otimes \mathbf{B} \\ \widetilde{\mathbf{A}}_{cl} = \widetilde{\mathbf{A}} - \begin{bmatrix} \widetilde{\mathbf{B}}_1 & \widetilde{\mathbf{B}}_2 \end{bmatrix} \mathbf{KC} \end{array} \right. \tag{4.206}$$

[11] Yuan L., Achenie L., Jiang W., Linear Quadratic Optimal Output Feedback Control For Systems With Poles In A Specified Region, International Journal of Control, Vol. 64(6), pages = 1151-1164, 1996

Then the optimal control problem for poles placement in a specified region reads as follows[11]:

$$\begin{array}{c} \text{Find } \mathbf{P} = \mathbf{P}^T > 0 \text{ and } \mathbf{K} \\ \text{which minimize } \operatorname{tr}(\mathbf{P}) \text{ under the constraints :} \\ \left\{ \begin{array}{c} (\mathbf{A} - \mathbf{BKC})^T \mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{BKC}) + \mathbf{Q}_K = \mathbf{0} \\ \widetilde{\mathbf{A}}_{cl}^T \mathbf{P} + \mathbf{P}\widetilde{\mathbf{A}}_{cl} < 0 \end{array} \right. \end{array} \tag{4.207}$$

where:

$$\widetilde{\mathbf{A}}_{cl} = \begin{bmatrix} \sin(\theta) & \cos(\theta) \\ -\cos(\theta) & \sin(\theta) \end{bmatrix} \otimes (\mathbf{A} - \alpha\mathbb{I} - \mathbf{BKC}) \tag{4.208}$$

4.8 Frequency shaped LQ control

Often system performances are specified in the frequency domain. The purpose of this section is to shift the time domain nature of the LQR problem in the frequency domain as proposed by Gupta in 1980[12]. This is done thanks to Parseval's theorem which enables to write the performance index J to be minimized as follows, where w represents frequency (in *rad/sec*):

$$\begin{array}{ll} J & = \int_0^\infty \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + u^T(t)\mathbf{R}u(t) \right) dt \\ & = \frac{1}{2\pi} \int_0^\infty \underline{x}^T(-j\omega)\mathbf{Q}\underline{x}(j\omega) + u^T(-j\omega)\mathbf{R}u(j\omega) dw \end{array} \tag{4.209}$$

Then constant weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ are modified to be function of the frequency w in order to place distinct penalties on the state and control cost at various frequencies:

$$\left\{ \begin{array}{l} \mathbf{Q} = \mathbf{Q}(w) = \mathbf{W}_q^T(-j\omega)\mathbf{W}_q(j\omega) \\ \mathbf{R} = \mathbf{R}(w) = \mathbf{W}_r^T(-j\omega)\mathbf{W}_r(j\omega) \end{array} \right. \tag{4.210}$$

For the existence of the solution of the LQ regulator, matrix $\mathbf{R}(w)$ shall be of full rank. Since we seek to minimize the quadratic cost J, then large terms in the integrand incur greater penalties than small terms and more effort is exerted to make then small. Thus if there is for example an high frequency region where the model of the plant presents unmodeled dynamics and if the control weight $\mathbf{W}_r(j\omega)$ is chosen to have large magnitude over this region then the resulting controller would not exert substantial energy in this region. This in turn would limit the controller bandwidth.

Let us define the following vectors to carry out the dynamics of the weights in the frequency domain, where s denotes the Laplace variable:

$$\left\{ \begin{array}{l} \underline{z}(s) = \mathbf{W}_q(s)\,\underline{x}(s) \\ v(s) = \mathbf{W}_r(s)\,u(s) \end{array} \right. \tag{4.211}$$

In order to simplify the process of selecting useful weights, it is common to choose weighting matrices to be scalar functions multiplying the identity matrix:

$$\left\{ \begin{array}{l} \mathbf{W}_q(s) = w_q(s)\mathbb{I} \\ \mathbf{W}_r(s) = w_r(s)\mathbb{I} \end{array} \right. \tag{4.212}$$

[12] Frequency-Shaped Cost Functionals: Extension of Linear Quadratic Gaussian Design Methods, Narendra K. Gupta, Journal of Guidance Control and Dynamics. 11/1980; 3(6):529-535. DOI: 10.2514/3.19722

The performance index J to be minimized in (4.209) turns to be:

$$J = \frac{1}{2\pi} \int_0^\infty \underline{z}^T(-j\omega)\underline{z}(j\omega) + v^T(-j\omega)v(j\omega)dw \tag{4.213}$$

Using Parseval's theorem we get in the time domain:

$$J = \int_0^\infty \underline{z}^T(t)\underline{z}(t) + v^T(t)v(t)\, dt \tag{4.214}$$

Let the state space model of the first equation of (4.211) be the following, where $\underline{z}(t)$ is the output and $\underline{x}(t)$ the input of the following MIMO system:

$$\begin{array}{l} \left\{ \begin{array}{l} \dot{\chi}_q(t) = \mathbf{A}_q \chi_q(t) + \mathbf{B}_q \underline{x}(t) \\ \underline{z}(t) = \mathbf{N}_q \chi_q(t) + \mathbf{D}_q \underline{x}(t) \end{array} \right. \\ \Rightarrow \underline{z}(s) = \left(\mathbf{N}_q \left(sI - \mathbf{A}_q \right)^{-1} \mathbf{B}_q + \mathbf{D}_q \right) \underline{x}(s) = \mathbf{W}_q(s)\underline{x}(s) \end{array} \tag{4.215}$$

Similarly, let the state space model of the second equation of (4.211) be the following, where $v(t)$ is the output and $u(t)$ the input of the following MIMO system:

$$\begin{array}{l} \left\{ \begin{array}{l} \dot{\chi}_r(t) = \mathbf{A}_r \chi_r(t) + \mathbf{B}_r u(t) \\ v(t) = \mathbf{N}_r \chi_r(t) + \mathbf{D}_r u(t) \end{array} \right. \\ \Rightarrow v(s) = \left(\mathbf{N}_r \left(sI - \mathbf{A}_r \right)^{-1} \mathbf{B}_r + \mathbf{D}_r \right) u(s) = \mathbf{W}_r(s)u(s) \end{array} \tag{4.216}$$

Then it can be shown from (4.215) and (4.216) that:

$$\begin{aligned} \underline{z}^T(t)\underline{z}(t) + v^T(t)v(t) = & \left(\mathbf{N}_q \chi_q(t) + \mathbf{D}_q \underline{x}(t) \right)^T \left(\mathbf{N}_q \chi_q(t) + \mathbf{D}_q \underline{x}(t) \right) \\ & + \left(\mathbf{N}_r \chi_r(t) + \mathbf{D}_r u(t) \right)^T \left(\mathbf{N}_r \chi_r(t) + \mathbf{D}_r u(t) \right) \end{aligned} \tag{4.217}$$

That is:

$$\begin{aligned} \underline{z}^T(t)\underline{z}(t) + v^T(t)v(t) = & \begin{bmatrix} \underline{x}^T(t) & \chi_q^T(t) & \chi_r(t)^T \end{bmatrix} \mathbf{Q}_f \begin{bmatrix} \underline{x}(t) \\ \chi_q(t) \\ \chi_r(t) \end{bmatrix} \\ & + 2 \begin{bmatrix} \underline{x}^T(t) & \chi_q^T(t) & \chi_r(t)^T \end{bmatrix} \mathbf{N}_f u(t) + u^T(t)\mathbf{R}_f u(t) \end{aligned} \tag{4.218}$$

Where:

$$\left\{ \begin{array}{c} \mathbf{Q}_f = \begin{bmatrix} \mathbf{D}_q^T \mathbf{D}_q & \mathbf{D}_q^T \mathbf{N}_q & 0 \\ \mathbf{N}_q^T \mathbf{D}_q & \mathbf{N}_q^T \mathbf{N}_q & 0 \\ 0 & 0 & \mathbf{N}_r^T \mathbf{N}_r \end{bmatrix} \\ \mathbf{N}_f = \begin{bmatrix} 0 \\ 0 \\ \mathbf{N}_r^T \mathbf{D}_r \end{bmatrix} \\ \mathbf{R}_f = \mathbf{D}_r^T \mathbf{D}_r \end{array} \right. \tag{4.219}$$

Then define the augmented state vector $\underline{x}_a(t)$:

$$\underline{x}_a(t) = \begin{bmatrix} \underline{x}(t) \\ \chi_q(t) \\ \chi_r(t) \end{bmatrix} \tag{4.220}$$

And the augmented state space model:

$$\underline{\dot{x}}_a(t) = \frac{d}{dt}\begin{bmatrix} \underline{x}(t) \\ \chi_q(t) \\ \chi_r(t) \end{bmatrix} = \mathbf{A}_a \underline{x}_a(t) + \mathbf{B}_a u(t) \tag{4.221}$$

Where:

$$\begin{cases} \mathbf{A}_a = \begin{bmatrix} \mathbf{A} & 0 & 0 \\ \mathbf{B}_q & \mathbf{A}_q & 0 \\ 0 & 0 & \mathbf{A}_r \end{bmatrix} \\ \mathbf{B}_a = \begin{bmatrix} \mathbf{B} \\ 0 \\ \mathbf{B}_r \end{bmatrix} \end{cases} \tag{4.222}$$

Using (4.218) the performance index J defined in (4.214) is written as follows:

$$J = \int_0^\infty \underline{x}_a^T(t)\mathbf{Q}_f\underline{x}_a(t) + 2\underline{x}_a(t)\mathbf{N}_f u(t) + u(t)^T\mathbf{R}_f u(t)\, dt \tag{4.223}$$

As far as cross-term in the performance index J appears results obtained in section 4.6.1 will be used: The algebraic Riccati equation (4.150) reads as follows:

$$\mathbf{P}\mathbf{A}_m + \mathbf{A}_m^T\mathbf{P} - \mathbf{P}\mathbf{B}_a\mathbf{R}^{-1}\mathbf{B}_a^T\mathbf{P} + \mathbf{Q}_m = \mathbf{0} \tag{4.224}$$

Where:

$$\begin{cases} \mathbf{Q}_m = \mathbf{Q}_f - \mathbf{N}_f\mathbf{R}_f^{-1}\mathbf{N}_f^T \\ \mathbf{A}_m = \mathbf{A}_a - \mathbf{B}_a\mathbf{R}_f^{-1}\mathbf{N}_f^T \end{cases} \tag{4.225}$$

Denoting by $\mathbf{P}$ the positive definite matrix which solves the *algebraic Riccati* equation (4.224), the stabilizing control $u(t)$ is then defined in a similar fashion than (4.149):

$$\begin{cases} u(t) = -\mathbf{K}\underline{x}(t) \\ \mathbf{K} = \mathbf{R}_f^{-1}\left(\mathbf{P}\mathbf{B}_a + \mathbf{N}_f\right)^T \end{cases} \tag{4.226}$$

4.9 Optimal transient stabilization

We provide in that section the material written by L. Qiu and K. Zhou[13]. Let's consider the feedback system for stabilization system in Figure 4.4 where $F(s)$ is the plant and $K(s)$ is the controller: The known transfer function $F(s)$ of the plant is assumed to be strictly proper with a *monic polynomial* on the denominator (a *monic polynomial* is a polynomial in which the leading coefficient (the nonzero coefficient of highest degree) is equal to 1):

$$F(s) = \frac{N(s)}{D(s)} = \frac{b_{n-1}s^{n-1} + \cdots b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0} \tag{4.227}$$

[13]Preclassical Tools for Postmodern Control, An optimal And Robust Control Theory For Undergraduate Education, Li Qiu and Kemin Zhou, IEEE Control Systems Magazine, August 2013

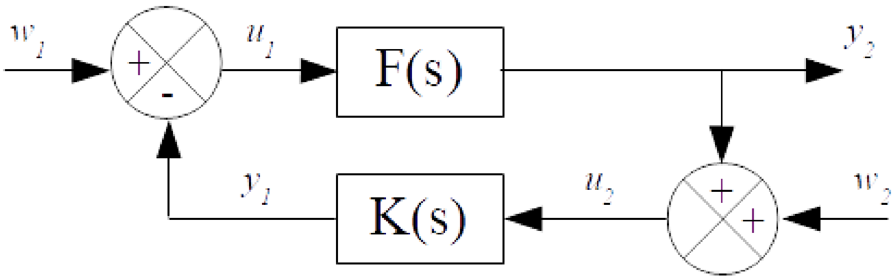


Figure 4.4: Feedback system for stabilization

Similarly the unknown transfer function $K(s)$ of the controller is assumed to be strictly proper with a monic polynomial on the denominator:

$$K(s) = \frac{q(s)}{p(s)} = \frac{q_{m-1}s^{m-1} + \cdots q_1 s + q_0}{s^m + p_{m-1}s^{m-1} + \cdots + p_1 s + p_0} \tag{4.228}$$

The closed-loop characteristic polynomial $\beta(s)$ is:

$$\begin{array}{ll} \beta(s) & = N(s)q(s) + D(s)p(s) \\ & = s^{n+m} + \beta_{n+m-1}\, s^{n+m-1} + \cdots + \beta_1\, s + \beta_0 \end{array} \tag{4.229}$$

For given coprime polynomials $N(s)$ and $D(s)$ as well as an arbitrarily chosen closed-loop characteristic polynomial $\beta(s)$ the expression of $p(s)$ and $q(s)$ amounts to solving the following Diophantine equation:

$$\beta(s) = N(s)q(s) + D(s)p(s) \tag{4.230}$$

This linear equation in the coefficients of $p(s)$ and $q(s)$ has solution for arbitrary $\beta(s)$ if and only if $m \geq n$. The solution is unique if and only if $m = n$. Now consider the following performance measure $J(\rho, \mu)$ where ρ and μ are positive number to give relative weights to outputs $y_1(t)$ and $y_2(t)$ and to inputs $w_1(t)$ and $w_2(t)$ respectively and $\delta(t)$ is the Dirac delta function:

$$\begin{aligned} J(\rho,\mu) = \frac{1}{2}\int_0^\infty y_1^2(t) + \rho y_2^2(t)\, dt \Bigg|_{\substack{w_1(t)=\mu\delta(t) \\ w_2(t)=0}} \\ + \frac{1}{2}\int_0^\infty y_1^2(t) + \rho y_2^2(t)\, dt \Bigg|_{\substack{w_1(t)=0 \\ w_2(t)=\delta(t)}} \end{aligned} \tag{4.231}$$

The design procedure to obtain the controller which minimizes performance measure $J(\rho, \mu)$ is the following:

– Find polynomial $d_\mu(s)$ (also called *spectral factor*) which is formed with the n roots with negative real parts of $D(s)D(-s) + \mu^2 N(s)N(-s)$:

$$D(s)D(-s) + \mu^2 N(s)N(-s) = d_\mu(s)d_\mu(-s) \tag{4.232}$$

- Find polynomial $d_\rho(s)$ (also called *spectral factor*) which is formed with the n roots with negative real parts of $D(s)D(-s) + \rho N(s)N(-s)$:

$$D(s)D(-s) + \rho N(s)N(-s) = d_\rho(s)d_\rho(-s) \tag{4.233}$$

- Then the optimal controller $K(s) = q(s)/p(s)$ is the unique n^{th} order strictly proper transfer function such that:

$$D(s)p(s) + N(s)q(s) = d_\mu(s)d_\rho(s) \tag{4.234}$$

Chapter 5

Linear Quadratic Tracker (LQT)

5.1 Introduction

The regulator problem that has been tackled in the previous chapters is in fact a spacial case of a wider class of problems where the outputs of the system are required to follow a desired trajectory in some optimal sense. As underlined in the book of *Anderson and Moore* trajectory following problems can be conveniently separated into three different problems which depend on the nature of the desired output trajectory:

- If the plant outputs are to follow a class of desired trajectories, for example all polynomials up to certain order, the problem is referred to as a servo (servomechanism) problem;
- When the plant outputs are to follow the response of another plant (or model) the problem is referred to as model following problems;
- If the desired output trajectory is a particular prescribed function of time, the problem is called a tracking problem.

This chapter is devoted to the presentation of some results common to all three of these problems, with a particular attention being given on the tracking problem.

5.2 Control with feedforward

We will consider in this section the following linear system, where $\underline{x}(t)$ is the state-vector, $\underline{u}(t)$ the control and $\underline{y}(t)$ the controlled output (that is the output of interest):

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\,\underline{x}(t) \end{cases} \tag{5.1}$$

Control with feedforward gain allows set point regulation. We will assume that control $\underline{u}(t)$ has the following expression where $\mathbf{F}$ is the feedforward gain and where $\underline{r}(t)$ is the commanded value for the output $\underline{y}(t)$:

$$\underline{u}(t) = -\mathbf{K}\,\underline{x}(t) + \mathbf{F}\underline{r}(t) \tag{5.2}$$

The optimal control problem is then split into two separate problems which are solved individually to form the suboptimal control:

– First the commanded value $\underline{r}(t)$ is set to zero and the gain $\mathbf{K}$ is computed to solve the Linear Quadratic Regulator (LQR) problem;

– Then the feedforward gain $\mathbf{F}$ is computed such that the steady-state value of output $\underline{y}(t)$ is equal to the commanded value $\underline{r}(t) := \underline{y}_c$.

$$\underline{r}(t) := \underline{y}_c \tag{5.3}$$

Using the expression (5.2) of the control $\underline{u}(t)$ within the state space realization (5.1) of the linear system leads to:

$$\begin{cases} \underline{\dot{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) = (\mathbf{A} - \mathbf{BK})\,\underline{x}(t) + \mathbf{BF}\underline{y}_c \\ \underline{y}(t) = \mathbf{C}\,\underline{x}(t) \end{cases} \tag{5.4}$$

Then matrix $\mathbf{F}$ is computed such that the steady-state value of output $\underline{y}(t)$ is $\underline{y}_c$. Assuming that $\underline{\dot{x}} = \underline{0}$, which corresponds to the steady-state, the preceding equations becomes:

$$\begin{cases} \underline{0} = (\mathbf{A} - \mathbf{BK})\,\underline{x} + \mathbf{BF}\underline{y}_c \Leftrightarrow \underline{x} = -(\mathbf{A} - \mathbf{BK})^{-1}\mathbf{BF}\underline{y}_c \\ \underline{y} = \mathbf{C}\,\underline{x} \end{cases} \tag{5.5}$$

That is:

$$\underline{y} = -\mathbf{C}(\mathbf{A} - \mathbf{BK})^{-1}\mathbf{BF}\underline{y}_c \tag{5.6}$$

Setting $\underline{y}$ to $\underline{y}_c$ and assuming that the size of the output vector $\underline{y}(t)$ is the same than the size of the control vector $\underline{u}$ (square plant) leads to the following expression of the feedforward gain $\mathbf{F}$:

$$\underline{y} = \underline{y}_c \Rightarrow \mathbf{F} = -\left(\mathbf{C}\,(\mathbf{A} - \mathbf{BK})^{-1}\,\mathbf{B}\right)^{-1} \tag{5.7}$$

For a square plant the feedforward gain $\mathbf{F}$ is nothing than the inverse of the closed-loop static gain (the closed-loop static gain is obtained by setting the Laplace variable s to 0 in the expression of the closed-loop transfer function).

5.3 Finite horizon Linear Quadratic Tracker

We will consider in this section the following linear system, where $\underline{x}(t)$ is the state-vector, $\underline{u}(t)$ the control and $\underline{y}(t)$ the measured output:

$$\begin{cases} \underline{\dot{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\,\underline{x}(t) \end{cases} \tag{5.8}$$

It is now desired to find an optimal control law in such a way that the controlled output $\underline{y}(t)$ tracks or follows a reference output $\underline{r}(t)$. Hence the performance index is defined as:

$$J(\underline{u}(t)) = \frac{1}{2}\underline{e}^T(t_f)\mathbf{S}\underline{e}(t_f) + \frac{1}{2}\int_0^{t_f} \underline{e}^T(t)\mathbf{Q}\underline{e}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\,dt \tag{5.9}$$

Where $\underline{e}(t)$ is the trajectory error defined as:

$$\underline{e}(t) := \underline{y}(t) - \underline{r}(t) = \mathbf{C}\,\underline{x}(t) - \underline{r}(t) \tag{5.10}$$

The Hamiltonian H is then defined as:

$$H(\underline{x}, \underline{u}, \underline{\lambda}) = \frac{1}{2}\underline{e}^T(t)\mathbf{Q}\,\underline{e}(t) + \frac{1}{2}\underline{u}^T(t)\mathbf{R}\underline{u}(t) + \underline{\lambda}^T(t)\left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right) \tag{5.11}$$

The optimality condition (1.72) yields:

$$\frac{\partial H}{\partial \underline{u}} = \underline{0} = \mathbf{R}\underline{u}(t) + \mathbf{B}^T\underline{\lambda}(t) \Rightarrow \underline{u}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \tag{5.12}$$

Equation (1.69) yields:

$$\begin{array}{rl} \dot{\underline{\lambda}}(t) & = -\frac{\partial H}{\partial \underline{x}} \\ & = -\left(\frac{\partial \underline{e}^T(t)}{\partial \underline{x}}\mathbf{Q}\,\underline{e}(t) + \mathbf{A}^T\underline{\lambda}(t)\right) \\ & = -\left(\mathbf{C}^T\mathbf{Q}\left(\mathbf{C}\,\underline{x}(t) - \underline{r}(t)\right) + \mathbf{A}^T\underline{\lambda}(t)\right) \\ \Leftrightarrow \dot{\underline{\lambda}}(t) & = -\mathbf{A}^T\underline{\lambda}(t) - \mathbf{C}^T\mathbf{Q}\mathbf{C}\,\underline{x}(t) + \mathbf{C}^T\mathbf{Q}\,\underline{r}(t) \end{array} \tag{5.13}$$

With the terminal condition (1.70):

$$\begin{array}{rl} \underline{\lambda}(t_f) & = \frac{\partial \frac{1}{2}\underline{e}^T(t_f)\mathbf{S}\underline{e}(t_f)}{\partial \underline{x}(t_f)} \\ & = \frac{\partial\left(\mathbf{C}\underline{x}(t_f) - \underline{r}(t_f)\right)^T}{\partial \underline{x}(t_f)}\mathbf{S}\underline{e}(t_f) \\ & = \mathbf{C}^T\mathbf{S}\left(\mathbf{C}\underline{x}(t_f) - \underline{r}(t_f)\right) \end{array} \tag{5.14}$$

In order to get the closed-loop control law, expression (2.25) is modified through a feedforward term $\underline{g}(t)$ to be determined:

$$\boxed{\underline{\lambda}(t) = \mathbf{P}(t)\underline{x}(t) - \underline{g}(t)} \tag{5.15}$$

Using (5.15) the terminal conditions (5.14) can be written as:

$$\mathbf{P}(t_f)\underline{x}(t_f) - \underline{g}(t_f) = \mathbf{C}^T\mathbf{S}\left(\mathbf{C}\underline{x}(t_f) - \underline{r}(t_f)\right) \tag{5.16}$$

Which implies by identification:

$$\left\{ \begin{array}{l} \mathbf{P}(t_f) = \mathbf{C}^T\mathbf{S}\mathbf{C} \\ \underline{g}(t_f) = \mathbf{C}^T\mathbf{S}\underline{r}(t_f) \end{array} \right. \tag{5.17}$$

Furthermore from (5.12) and (5.15) the control law reads:

$$\begin{array}{rl} \underline{u}(t) & = -\mathbf{R}^{-1}\mathbf{B}^T\underline{\lambda}(t) \\ & = -\mathbf{R}^{-1}\mathbf{B}^T\left(\mathbf{P}(t)\underline{x}(t) - \underline{g}(t)\right) \\ & = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)\,\underline{x}(t) + \mathbf{R}^{-1}\mathbf{B}^T\,\underline{g}(t) \end{array} \tag{5.18}$$

From the preceding equation it is clear that the optimal control is the sum of two components:

– a state-feedback component: $-\mathbf{K}(t)\,\underline{x}(t)$ where $\mathbf{K}(t) = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t)$;

– and a feedforward component: $\underline{v}(t) := \mathbf{R}^{-1}\mathbf{B}^T\underline{g}(t)$

In addition, differentiating (5.15) yields:

$$\dot{\underline{\lambda}}(t) = \dot{\mathbf{P}}(t)\underline{x}(t) + \mathbf{P}(t)\dot{\underline{x}}(t) - \dot{\underline{g}}(t) \tag{5.19}$$

Using (5.13) we get:

$$\begin{aligned} &-\mathbf{A}^T\underline{\lambda}(t) - \mathbf{C}^T\mathbf{Q}\mathbf{C}\,\underline{x}(t) + \mathbf{C}^T\mathbf{Q}\,\underline{r}(t) \\ &\qquad = \dot{\mathbf{P}}(t)\underline{x}(t) + \mathbf{P}(t)\left(\mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)\right) - \dot{\underline{g}}(t) \end{aligned} \tag{5.20}$$

Using (5.8), (5.15) and (5.18) to express $\underline{u}(t)$ as a function of $\underline{x}(t)$ and $\underline{g}(t)$ we finally get:

$$\begin{aligned} &\left(\dot{\mathbf{P}}(t) + \mathbf{A}^T\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A} - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t) + \mathbf{C}^T\mathbf{Q}\mathbf{C}\right)\underline{x}(t) \\ &\qquad - \dot{\underline{g}}(t) - \left(\mathbf{A}^T - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\right)\underline{g}(t) - \mathbf{C}^T\mathbf{Q}\,\underline{r}(t) = \underline{0} \end{aligned} \tag{5.21}$$

The solution of (5.21) can be obtained by solving the preceding differential equation with final conditions (5.17) as two separate problems

$$\begin{cases} -\dot{\mathbf{P}}(t) = \mathbf{A}^T\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A} - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}(t) + \mathbf{C}^T\mathbf{Q}\mathbf{C} \\ \mathbf{P}(t_f) = \mathbf{C}^T\mathbf{S}\mathbf{C} \end{cases} \tag{5.22}$$

and:

$$\begin{cases} -\dot{\underline{g}}(t) &= \left(\mathbf{A}^T - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\right)\underline{g}(t) + \mathbf{C}^T\mathbf{Q}\,\underline{r}(t) \\ &:= \left(\mathbf{A} - \mathbf{B}\mathbf{K}(t)\right)^T\underline{g}(t) + \mathbf{C}^T\mathbf{Q}\,\underline{r}(t) \\ \underline{g}(t_f) &= \mathbf{C}^T\mathbf{S}\underline{r}(t_f) \end{cases} \tag{5.23}$$

Thus the implementation of the tracker (5.18) in real-time involves a standard optimal feedback regulator and a feedforward controller:

– The feedback regulator term requires the backward-in-time solution of the differential Riccati equation (5.22). This differential Riccati equation is independent of the reference signal $\underline{r}(t)$ and its solution has been studied in section 2.5.

– For particular applications where the reference signal $\underline{r}(t)$ is known a priori, the feedforward term $\underline{g}(t)$ can also be computed off-line by integrating, backwards in time, the differential equation (5.23). Backward integration is achieved when the time is reversed, that is by setting $\tau = t_f - t$ (thus the minus signs to the left of equality (5.23) is omitted). Then the initial value of the feedforward term $\underline{g}(0)$ is known and during the actual control run $\underline{g}(0)$ can be used to solve a forward differential equation instead[1].

[1]Rocio Alba-Flores and Enrique Barbieri, Real-time Infinite Horizon Linear-Quadratic Tracking Controller for Vibration Quenching in Flexible Beams, 2006 IEEE Conference on Systems, Man, and Cybernetics, October 8-11, 2006, Taipei, Taiwan

5.4 Infinite horizon Linear Quadratic Tracker

5.4.1 General result

When infinite horizon is considered, the performance index (5.9) is changed as follows where the tracking error $\underline{e}(t)$ is defined in (5.10):

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^{\infty} \underline{e}^T(t)\mathbf{Q}\underline{e}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\,dt \tag{5.24}$$

Assuming that $(\mathbf{A}, \mathbf{B})$ is detectable and $(\mathbf{A}, \sqrt{\mathbf{Q}}\,\mathbf{C})$ is detectable, there exists a unique steady-state solution of equations (5.22) obtained through the corresponding algebraic Riccati equation. Assuming that we want to achieve a perfect tracking of $\underline{r}(t)$, control law (5.18) can be written as:

$$\underline{u}(t) = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}\underline{x}(t) + \mathbf{R}^{-1}\mathbf{B}^T\underline{g}(t) \tag{5.25}$$

Matrix $\mathbf{P}$ is the positive definite solution of the following algebraic Riccati equation which is derived from (5.22) by setting $\dot{\mathbf{P}} = \mathbf{0}$:

$$\dot{\mathbf{P}} = \mathbf{0} \Rightarrow \mathbf{0} = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{C}^T\mathbf{Q}\mathbf{C} \tag{5.26}$$

On the other hand, feedforward term $\underline{g}(t)$ is derived from (5.23):

$$\begin{cases} \dot{\underline{g}}(t) = -\left(\mathbf{A} - \mathbf{B}\mathbf{K}\right)^T \underline{g}(t) - \mathbf{C}^T\mathbf{Q}\,\underline{r}(t) \\ \lim\limits_{t_f\to\infty} \underline{g}(t_f) = \mathbf{C}^T\mathbf{S}\underline{r}(t_f) \end{cases} \tag{5.27}$$

However, it is worth noticing that all the eigenvalues of state matrix $-\left(\mathbf{A} - \mathbf{B}\mathbf{K}\right)^T$ are situated in the right half plane (thus unstable) because gain $\mathbf{K}$ is such that all the eigenvalues of $(\mathbf{A} - \mathbf{B}\mathbf{K})$ are stable. Thus feedforward term $\underline{g}(t)$ is not bounded in general. Nevertheless the infinite horizon tracker can be approximated over a finite control interval $[0, t_f]$ by using the steady-state gain $\mathbf{K}$ and the auxiliary function $\underline{g}(t)$ where t_f is large enough.

5.4.2 Asymptotically stable linear reference model

We will assume hereafter that the reference signal $\underline{r}(t)$ is given as the output of the following asymptotically stable linear reference model where $\mathbf{A}_r$ is known and has all its eigenvalues in the left-half plane:

$$\dot{\underline{r}}(t) = \mathbf{A}_r\underline{r}(t) \tag{5.28}$$

Then by combining (5.8) and (5.28) the following augmented plant can be build with state-vector $\underline{x}_a(t) := \begin{bmatrix} \underline{x}(t) \\ \underline{r}(t) \end{bmatrix}$. We will denote $\mathbf{A}_a$, $\mathbf{B}_a$ and $\mathbf{C}_a$ the related matrices corresponding to the state-space representation of this augmented system:

$$\underline{x}_a(t) := \begin{bmatrix} \underline{x}(t) \\ \underline{r}(t) \end{bmatrix} \Rightarrow \begin{cases} \begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\underline{r}}(t) \end{bmatrix} & = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_r \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{r}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \underline{u}(t) \\ & := \mathbf{A}_a\underline{x}_a(t) + \mathbf{B}_a\underline{u}(t) \\ \underline{e}(t) & = \underline{y}(t) - \underline{r}(t) = \begin{bmatrix} \mathbf{C} & -\mathbb{I} \end{bmatrix} \underline{x}_a(t) \\ & := \mathbf{C}_a\underline{x}_a(t) \end{cases} \tag{5.29}$$

Then minimization of performance index (5.24) is achieved by applying classical results on LQR problems and control $\underline{u}(t)$ reads as follows[2]:

$$\underline{u}(t) = -\mathbf{K}_a \begin{bmatrix} \underline{x}(t) \\ \underline{r}(t) \end{bmatrix} \text{ where } \mathbf{K}_a = \mathbf{R}^{-1}\mathbf{B}_a^T\mathbf{P}_a \tag{5.30}$$

where $\mathbf{P}_a$ is the positive definite solution of the following algebraic Riccati equation:

$$\mathbf{0} = \mathbf{A}_a^T\mathbf{P}_a + \mathbf{P}_a\mathbf{A}_a - \mathbf{P}_a\mathbf{B}_a\mathbf{R}^{-1}\mathbf{B}_a^T\mathbf{P}_a + \mathbf{C}_a^T\mathbf{Q}\mathbf{C}_a \tag{5.31}$$

5.4.3 Constant reference tracking

Finally, let's assume a constant value for $\underline{r}(t)$, which will be denoted $\underline{r}_{ss}$. Then at steady-state (5.8) reads as follows, where $\underline{x}_{ss}$ denotes the steady-state of $\underline{x}(t)$:

$$\begin{cases} \underline{0} = \mathbf{A}\underline{x}_{ss} + \mathbf{B}\underline{u}_{ss} \\ \underline{y}_{ss} = \mathbf{C}\,\underline{x}_{ss} \end{cases} \tag{5.32}$$

If we impose $\underline{y}_{ss} := \underline{r}_{ss}$, the preceding relations read:

$$\underline{y}_{ss} := \underline{r}_{ss} \Rightarrow \begin{bmatrix} \underline{0} \\ \underline{r}_{ss} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \underline{x}_{ss} \\ \underline{u}_{ss} \end{bmatrix} \tag{5.33}$$

Assuming that matrix $\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{bmatrix}$ is square and invertible, we get:

$$\begin{aligned} \begin{bmatrix} \underline{x}_{ss} \\ \underline{u}_{ss} \end{bmatrix} &= \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \underline{0} \\ \underline{r}_{ss} \end{bmatrix} \\ &:= \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} \begin{bmatrix} \underline{0} \\ \underline{r}_{ss} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{M}_{12} \\ \mathbf{M}_{22} \end{bmatrix} \underline{r}_{ss} \end{aligned} \tag{5.34}$$

Then let $\underline{\widetilde{x}}(t)$ the error between the actual state-vector $\underline{x}(t)$ and its steady-state value $\underline{x}_{ss}$ and $\underline{\widetilde{u}}(t)$ the error between the actual control $\underline{u}(t)$ and its steady-state value $\underline{u}_{ss}$:

$$\begin{cases} \underline{\widetilde{x}}(t) := \underline{x}(t) - \underline{x}_{ss} \\ \underline{\widetilde{u}}(t) := \underline{u}(t) - \underline{u}_{ss} \end{cases} \tag{5.35}$$

Then using (5.35) the dynamics of $\underline{\widetilde{x}}(t)$ reads:

$$\begin{aligned} \dot{\underline{\widetilde{x}}}(t) &= \dot{\underline{x}}(t) \\ &= \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ &= \mathbf{A}\left(\underline{\widetilde{x}}(t) + \underline{x}_{ss}\right) + \mathbf{B}\left(\underline{\widetilde{u}}(t) + \underline{u}_{ss}\right) \end{aligned} \tag{5.36}$$

It is clear from (5.33) that $\mathbf{A}\underline{x}_{ss} + \mathbf{B}\underline{u}_{ss} = \underline{0}$. We finally get:

$$\mathbf{A}\underline{x}_{ss} + \mathbf{B}\underline{u}_{ss} = \underline{0} \Rightarrow \dot{\underline{\widetilde{x}}}(t) = \mathbf{A}\underline{\widetilde{x}}(t) + \mathbf{B}\underline{\widetilde{u}}(t) \tag{5.37}$$

[2] Hamidreza Modares, Frank L. Lewis, Online Solution to the Linear Quadratic Tracking Problem of Continuous-time Systems using Reinforcement Learning, 52nd IEEE Conference on Decision and Control, December 10-13, 2013. Florence, Italy

In addition, the tracking error defined in (5.10) becomes:

$$\underline{e}(t) := \underline{y}(t) - \underline{r}_{ss} = \mathbf{C}\,\underline{x}(t) - \mathbf{C}\,\underline{x}_{ss} = \mathbf{C}\,\widetilde{\underline{x}}(t) \tag{5.38}$$

Then we consider the minimization of following performance index :

$$J(\widetilde{\underline{u}}(t)) = \frac{1}{2}\int_0^{\infty} \underline{e}^T(t)\mathbf{Q}\underline{e}(t) + \widetilde{\underline{u}}^T(t)\mathbf{R}\widetilde{\underline{u}}(t)\,dt \tag{5.39}$$

The minimization of $J(\widetilde{\underline{u}}(t))$ is achieved by applying classical results on LQR problems and control $\widetilde{\underline{u}}(t)$ reads as follows:

$$\widetilde{\underline{u}}(t) = -\mathbf{K}\widetilde{\underline{x}}(t) \text{ where } \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \tag{5.40}$$

Matrix $\mathbf{P}$ is the positive definite solution of the following algebraic Riccati equation:

$$\mathbf{0} = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{C}^T\mathbf{Q}\mathbf{C} \tag{5.41}$$

The actual control is finally obtained thanks to (5.35):

$$\underline{u}(t) = \widetilde{\underline{u}}(t) + \underline{u}_{ss} = -\mathbf{K}\,\widetilde{\underline{x}}(t) + \underline{u}_{ss} = -\mathbf{K}\,(\underline{x}(t) - \underline{x}_{ss}) + \underline{u}_{ss} \tag{5.42}$$

That is, using (5.34):

$$\begin{array}{ll} \underline{u}(t) & = -\mathbf{K}\,\underline{x}(t) + \mathbf{K}\,\underline{x}_{ss} + \underline{u}_{ss} \\ & = -\mathbf{K}\,\underline{x}(t) + \begin{bmatrix} \mathbf{K} & \mathbb{I} \end{bmatrix} \begin{bmatrix} \underline{x}_{ss} \\ \underline{u}_{ss} \end{bmatrix} \\ & = -\mathbf{K}\,\underline{x}(t) + \begin{bmatrix} \mathbf{K} & \mathbb{I} \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \underline{0} \\ \underline{r}_{ss} \end{bmatrix} \\ & = -\mathbf{K}\,\underline{x}(t) + (\mathbf{K}\,\mathbf{M}_{12} + \mathbf{M}_{22})\,\underline{r}_{ss} \\ & = -\mathbf{K}\,\underline{x}(t) + \mathbf{F}\,\underline{r}_{ss} \text{ where } \mathbf{F} := \mathbf{K}\,\mathbf{M}_{12} + \mathbf{M}_{22} \end{array} \tag{5.43}$$

5.5 Plant augmented with integrator

5.5.1 Integral augmentation

An alternative to make the steady-state error exactly equal to zero in response to a step for the commanded value $\underline{r}(t) = \underline{y}_c$ is to replace the feedforward gain $\mathbf{F}$ by an integrator which will cancel the steady-state error whatever the input step (the system's type is augmented to be of type 1). The advantage of adding an integrator is that it eliminates the need to determine the feedforward gain $\mathbf{F}$ which could be difficult because of the uncertainty in the model.

By augmenting the system with the integral error the LQR routine will choose the value of the integral gain automatically.

The integrator is denoted $\mathbf{T}/s$, where $\mathbf{T} \neq \mathbf{0}$ is a constant which may be used to increase the response of the closed-loop system. Let $\underline{x}_i$ be the additional component of the state-vector which is proportional to the integral of the error $\underline{e}(t) = \underline{r}(t) - \underline{y}(t)$. Adding an integrator augments the system's dynamics as follows:

$$\begin{cases} \underline{\dot{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\,\underline{x}(t) \\ \underline{\dot{x}}_i(t) = \mathbf{T}\underline{e}(t) = \mathbf{T}\left(\underline{r}(t) - \underline{y}(t)\right) = \mathbf{T}\underline{r}(t) - \mathbf{TC}\,\underline{x}(t) \end{cases}$$
$$\Leftrightarrow \begin{cases} \frac{d}{dt}\begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{TC} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \underline{u}(t) + \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix} \underline{r}(t) \\ \underline{y}(t) = \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} \end{cases} \tag{5.44}$$

Then, the suboptimal control is found by solving the LQR regulation problem where $\underline{r} = \underline{0}$:

- The augmented state space model reads:

$$\frac{d}{dt}\begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} = \underline{\dot{x}}_a(t) = \mathbf{A}_a\underline{x}_a(t) + \mathbf{B}_a\underline{u}(t) \text{ where } \begin{cases} \mathbf{A}_a = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{TC} & \mathbf{0} \end{bmatrix} \\ \mathbf{B}_a = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \end{cases} \tag{5.45}$$

- The performance index $J(\underline{u}(t))$ to be minimized is the following:

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^\infty \underline{x}_a^T(t)\mathbf{Q}_a\underline{x}_a(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\,dt \tag{5.46}$$

  Where, denoting by $\mathbf{N}_a$ a design matrix, matrix $\mathbf{Q}_a$ is defined as follows:

$$\mathbf{Q}_a = \mathbf{N}_a^T\mathbf{N}_a \tag{5.47}$$

  Note that design matrix $\mathbf{N}_a$ shall be chosen such pair $(\mathbf{A}_a, \mathbf{N}_a)$ is detectable.

Assuming that pair $(\mathbf{A}_a, \mathbf{B}_a)$ is stabilizable and pair $(\mathbf{A}_a, \mathbf{N}_a)$ is detectable the algebraic Riccati equation can be solved. This leads to the following expression of the control $\underline{u}(t)$ (here feedforward gain $\mathbf{F}$ no more exists):

$$\begin{aligned} \underline{u}(t) &= -\mathbf{K}_a\,\underline{x}_a(t) \\ &= -\mathbf{R}^{-1}\mathbf{B}_a^T\mathbf{P}\,\underline{x}_a(t) \\ &= -\mathbf{R}^{-1}\begin{bmatrix} \mathbf{B}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} \\ &= -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_{11}\,\underline{x}(t) - \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_{12}\,\underline{x}_i(t) \\ &:= -\mathbf{K}_p\,\underline{x}(t) - \mathbf{K}_i\,\underline{x}_i(t) \end{aligned} \tag{5.48}$$

Obviously, term $\mathbf{K}_p = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_{11}$ represents the *proportional* gain of the controller whereas term $\mathbf{K}_i = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}_{12}$ represents the *integral* gain of the controller.

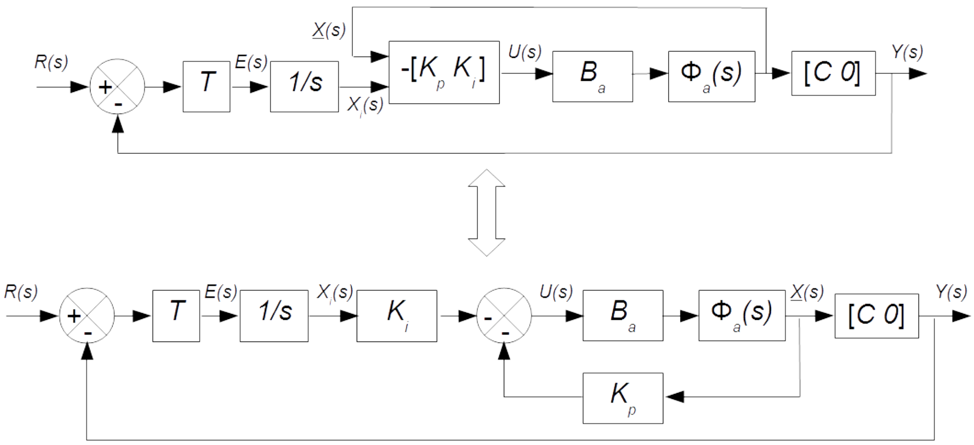


Figure 5.1: Plant augmented with integrator

The state space equation of closed-loop system is obtained by setting $\underline{u}(t) = -\mathbf{K}_a\,\underline{x}_a(t) = -\mathbf{K}_p\,\underline{x}(t) - \mathbf{K}_i\,\underline{x}_i(t)$ in (5.44):

$$\left\{\begin{array}{ll} \frac{d}{dt}\begin{bmatrix}\underline{x}(t)\\ \underline{x}_i(t)\end{bmatrix} & = (\mathbf{A}_a - \mathbf{B}_a\mathbf{K}_a)\,\underline{x}_a(t) + \begin{bmatrix}\mathbf{0}\\ \mathbf{T}\end{bmatrix}\underline{r}(t) \\ & = \begin{bmatrix}\mathbf{A}-\mathbf{B}\mathbf{K}_p & -\mathbf{B}\mathbf{K}_i\\ -\mathbf{T}\mathbf{C} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\underline{x}(t)\\ \underline{x}_i(t)\end{bmatrix} + \begin{bmatrix}\mathbf{0}\\ \mathbf{T}\end{bmatrix}\underline{r}(t) \\ \underline{y}(t) & = \begin{bmatrix}\mathbf{C} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\underline{x}(t)\\ \underline{x}_i(t)\end{bmatrix} \\ \underline{x}_i(t) & = \mathbf{T}\int_0^t \left(\underline{r}(\tau) - \underline{y}(\tau)\right)\,d\tau \end{array}\right. \tag{5.49}$$

The corresponding bloc diagram is shown in Figure 5.1 where $\Phi_a(s) = (s\mathbb{I} - \mathbf{A}_a)^{-1}$.

5.5.2 Proof of the cancellation of the steady-state error through integral augmentation

In order to proof that integrator cancels the steady-state error when $\underline{r}(t)$ is a step input, let us compute the final value of the error $\underline{e}(t)$ using the final value theorem where s denotes the Laplace variable:

$$\lim_{t\to\infty} \underline{e}(t) = \lim_{s\to 0} s\underline{E}(s) \tag{5.50}$$

When $\underline{r}(t)$ is a step input with amplitude one, we have:

$$\underline{r}(t) = 1\ \forall\ t \geq 0 \Rightarrow \underline{R}(s) = \frac{1}{s} \tag{5.51}$$

Using the feedback $\underline{u} = -\mathbf{K}_a\underline{x}_a$ the dynamics of the closed-loop system is:

$$\begin{array}{rl} \dot{\underline{x}}_a & = (\mathbf{A}_a - \mathbf{B}_a\mathbf{K}_a)\,\underline{x}_a + \begin{bmatrix}\mathbf{0}\\ \mathbf{T}\end{bmatrix}\underline{r}(t) \\ \Rightarrow \underline{e}(t) & = \mathbf{T}\left(\underline{r}(t) - \underline{y}(t)\right) \\ & = \mathbf{T}\left(\underline{r}(t) - \begin{bmatrix}\mathbf{C} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\underline{x}\\ \underline{x}_i\end{bmatrix}\right) \\ & = \mathbf{T}\left(\underline{r}(t) - \begin{bmatrix}\mathbf{C} & \mathbf{0}\end{bmatrix}\underline{x}_a\right) \end{array} \tag{5.52}$$

Using the Laplace transform, and denoting by $\mathbb{I}$ the identity matrix, we get:

$$\begin{cases} \underline{X}_a(s) = (s\mathbb{I} - \mathbf{A}_a + \mathbf{B}_a\mathbf{K}_a)^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix} \underline{R}(s) \\ \underline{E}(s) = \mathbf{T}\left(\underline{R}(s) - \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} \underline{X}_a(s)\right) \end{cases} \tag{5.53}$$

Inserting (5.51) in (5.53) we get:

$$\underline{E}(s) = \mathbf{T}\left(\mathbb{I} - \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} (s\mathbb{I} - \mathbf{A}_a + \mathbf{B}_a\mathbf{K}_a)^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix}\right) \frac{1}{s} \tag{5.54}$$

Then the final value theorem (5.50) takes the following expression:

$$\begin{aligned} \lim_{t\to\infty} \underline{e}(t) &= \lim_{s\to 0} s\underline{E}(s) \\ &= \lim_{s\to 0} \mathbf{T}\left(\mathbb{I} - \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} (s\mathbb{I} - \mathbf{A}_a + \mathbf{B}_a\mathbf{K}_a)^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix}\right) \\ &= \mathbf{T}\left(\mathbb{I} - \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} (-\mathbf{A}_a + \mathbf{B}_a\mathbf{K}_a)^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix}\right) \end{aligned} \tag{5.55}$$

Let us focus on the inverse of the matrix $-\mathbf{A}_a + \mathbf{B}_a\mathbf{K}_a$. First we write $\mathbf{K}_a$ as $\mathbf{K}_a = \begin{bmatrix} \mathbf{K}_p & \mathbf{K}_i \end{bmatrix}$, where $\mathbf{K}_p$ and $\mathbf{K}_i$ represents respectively the proportional and the integral gains. Then using (5.45) we get:

$$-\mathbf{A}_a + \mathbf{B}_a\mathbf{K}_a = \begin{bmatrix} -\mathbf{A} & \mathbf{0} \\ \mathbf{TC} & \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{K}_p & \mathbf{K}_i \end{bmatrix} = \begin{bmatrix} -\mathbf{A} + \mathbf{BK}_p & \mathbf{BK}_i \\ \mathbf{TC} & \mathbf{0} \end{bmatrix} \tag{5.56}$$

Assuming that $\mathbf{X}$ is a square invertible matrix, it can be shown that the inverse of the matrix $\begin{bmatrix} \mathbf{X} & \mathbf{Y} \\ \mathbf{Z} & \mathbf{0} \end{bmatrix}$ is the following:

$$\begin{bmatrix} \mathbf{X} & \mathbf{Y} \\ \mathbf{Z} & \mathbf{0} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{0} & \mathbf{Y}(\mathbf{ZY})^{-1} \\ \mathbf{Y}^T\left(\mathbf{YY}^T\right)^{-1} & \mathbf{W} \end{bmatrix}$$
$$\text{where } \mathbf{XY}(\mathbf{ZY})^{-1} + \mathbf{YW} = \mathbf{0} \tag{5.57}$$

Thus:

$$\begin{aligned} (-\mathbf{A}_a + \mathbf{B}_a\mathbf{K}_a)^{-1} &= \begin{bmatrix} -\mathbf{A} + \mathbf{BK}_p & \mathbf{BK}_i \\ \mathbf{TC} & \mathbf{0} \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \mathbf{0} & \mathbf{BK}_i(\mathbf{TCBK}_i)^{-1} \\ (\mathbf{BK}_i)^T\left(\mathbf{BK}_i(\mathbf{BK}_i)^T\right)^{-1} & \mathbf{W} \end{bmatrix} \end{aligned} \tag{5.58}$$

And:

$$\begin{aligned} (-\mathbf{A}_a + \mathbf{B}_a\mathbf{K})^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix} &= \begin{bmatrix} \mathbf{0} & \mathbf{BK}_i(\mathbf{TCBK}_i)^{-1} \\ * & \mathbf{W} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{BK}_i(\mathbf{TCBK}_i)^{-1}\mathbf{T} \\ \mathbf{WT} \end{bmatrix} \\ \Rightarrow \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} (-\mathbf{A}_a + \mathbf{B}_a\mathbf{K})^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix} &= \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{BK}_i(\mathbf{TCBK}_i)^{-1}\mathbf{T} \\ \mathbf{WT} \end{bmatrix} \\ &= \mathbf{CBK}_i(\mathbf{TCBK}_i)^{-1}\mathbf{T} \end{aligned} \tag{5.59}$$

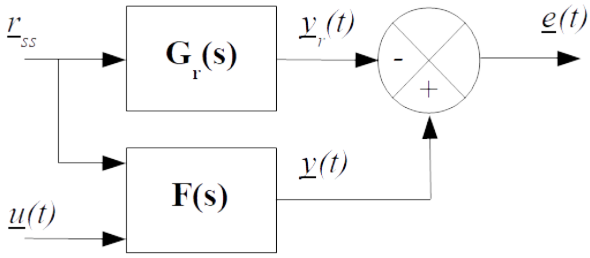


Figure 5.2: Linear Quadratic Tracker with constant reference signal $\underline{r}_{ss}$

Consequently, using (5.59) in (5.55), the final value of the error $\underline{e}(t)$ becomes:

$$\begin{array}{ll} \lim_{t\to\infty} \underline{e}(t) & = \mathbf{T}\left(\mathbb{I} - \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} (-\mathbf{A}_a + \mathbf{B}_a\mathbf{K}_a)^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix}\right) \\ & = \mathbf{T}\left(\mathbb{I} - \mathbf{CBK}_i \left(\mathbf{TCBK}_i\right)^{-1} \mathbf{T}\right) \\ & = \mathbf{T} - \mathbf{TCBK}_i \left(\mathbf{TCBK}_i\right)^{-1} \mathbf{T} \\ & = \mathbf{T} - \mathbf{T} \\ & = \mathbf{0} \end{array} \tag{5.60}$$

As a consequence, the integrator allows to cancel the steady-state error whatever the input step $\underline{r}(t)$.

5.6 Tracking with prefilter

5.6.1 Tracking without integral augmentation

We consider Figure 5.2 and the problem to design a control $\underline{u}(t)$ which minimizes the error $\underline{e}(t)$ between the output $\underline{y}_r(t)$ of the reference model represented by $\mathbf{G}_r(s)$ and the actual output $\underline{y}(t)$ of the plant represented by $\mathbf{F}(s)$.

The state space realization of $\mathbf{G}_r(s)$ and $\mathbf{F}(s)$ are assumed to read as follows, where $\underline{r}_{ss}$ is a *constant* reference signal and where input matrix $\mathbf{B}_2$ has been introduced to tackle the case where an integrator is inserted in the feedforward path of the loop, as presented in Section 5.5:

$$\left\{ \begin{array}{l} \mathbf{F}(s) : \left\{ \begin{array}{l} \dot{\underline{x}}(t) = \mathbf{A}_1\underline{x}(t) + \mathbf{B}_1\underline{u}(t) + \mathbf{B}_2\underline{r}_{ss} \\ \underline{y}(t) = \mathbf{C}_1\,\underline{x}(t) \end{array} \right. \\ \mathbf{G}_r(s) : \left\{ \begin{array}{l} \dot{\underline{x}}_r(t) = \mathbf{A}_r\underline{x}_r(t) + \mathbf{B}_r\underline{r}_{ss} \\ \underline{y}_r(t) = \mathbf{C}_r\,\underline{x}_r(t) \end{array} \right. \end{array} \right. \tag{5.61}$$

Assuming that no integrator is inserted in the feedforward path of the loop, matrices $\mathbf{A}_1$, $\mathbf{B}_1$ and $\mathbf{C}_1$ are related to the state-space representation of the actual plant:

$$\left\{ \begin{array}{l} \mathbf{A}_1 := \mathbf{A} \\ \mathbf{B}_1 := \mathbf{B} \\ \mathbf{B}_2 := \mathbf{0} \\ \mathbf{C}_1 := \mathbf{C} \end{array} \right. \tag{5.62}$$

The tracking error $\underline{e}(t)$ reads:

$$\underline{e}(t) := \underline{y}(t) - \underline{y}_r(t) = \mathbf{C}_1\, \underline{x}(t) - \mathbf{C}_r\, \underline{x}_r(t) \tag{5.63}$$

In order to solve this problem, we first compute the steady-state values imposing that at steady-state we shall have $\underline{y}_{ss} = \underline{y}_{r_{ss}}$. From (5.61) we get:

$$\begin{cases} \underline{0} = \mathbf{A}_1\, \underline{x}_{ss} + \mathbf{B}_1\, \underline{u}_{ss} + \mathbf{B}_2\, \underline{r}_{ss} \\ \underline{0} = \mathbf{A}_r\, \underline{x}_{r_{ss}} + \mathbf{B}_r\, \underline{r}_{ss} \\ \underline{y}_{ss} = \underline{y}_{r_{ss}} \Leftrightarrow \mathbf{C}_1\, \underline{x}_{ss} = \mathbf{C}_r\, \underline{x}_{r_{ss}} \end{cases}$$

$$\Leftrightarrow \begin{bmatrix} -\mathbf{B}_2 \\ -\mathbf{B}_r \\ \mathbf{0} \end{bmatrix} \underline{r}_{ss} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \mathbf{B}_1 \\ \mathbf{0} & \mathbf{A}_r & \mathbf{0} \\ \mathbf{C}_1 & -\mathbf{C}_r & \mathbf{0} \end{bmatrix} \begin{bmatrix} \underline{x}_{ss} \\ \underline{x}_{r_{ss}} \\ \underline{u}_{ss} \end{bmatrix} \tag{5.64}$$

Let matrix $\mathbf{M}$ be defined as follows:

$$\mathbf{M} := \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \mathbf{B}_1 \\ \mathbf{0} & \mathbf{A}_r & \mathbf{0} \\ \mathbf{C}_1 & -\mathbf{C}_r & \mathbf{0} \end{bmatrix} \tag{5.65}$$

Assuming that matrix $\mathbf{M}$ is invertible, we get:

$$\begin{bmatrix} \underline{x}_{ss} \\ \underline{x}_{r_{ss}} \\ \underline{u}_{ss} \end{bmatrix} = \mathbf{M}^{-1} \begin{bmatrix} -\mathbf{B}_2 \\ -\mathbf{B}_r \\ \mathbf{0} \end{bmatrix} \underline{r}_{ss} \tag{5.66}$$

Let:

$$\mathbf{M}^{-1} \begin{bmatrix} -\mathbf{B}_2 \\ -\mathbf{B}_r \\ \mathbf{0} \end{bmatrix} := \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \\ \mathbf{M}_3 \end{bmatrix} \tag{5.67}$$

Thus:

$$\boxed{\begin{bmatrix} \underline{x}_{ss} \\ \underline{x}_{r_{ss}} \\ \underline{u}_{ss} \end{bmatrix} = \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \\ \mathbf{M}_3 \end{bmatrix} \underline{r}_{ss}} \tag{5.68}$$

Note that if matrix $\mathbf{M}$ is not invertible, its pseudo inverse (Moore–Penrose inverse) $\mathbf{M}^+$ can be used instead of its inverse $\mathbf{M}^{-1}$. We recall that $\mathbf{M}^+$ is a generalization of the inverse of a matrix and is such that $\mathbf{M}\mathbf{M}^+\mathbf{M} = \mathbf{M}$ and $\mathbf{M}^+\mathbf{M}\mathbf{M}^+ = \mathbf{M}^+$. It can be computed by using the singular value decomposition (SVD) of $\mathbf{M}$: If $\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^*$ is the singular value decomposition (SVD) of $\mathbf{M}$, then $\mathbf{M}^+ = \mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^*$.

Then let $\underline{\widetilde{x}}(t)$ the error between the actual state-vector $\underline{x}(t)$ and its steady-state value $\underline{x}_{ss}$, $\underline{\widetilde{x}}_r(t)$ the error between the reference model state-vector $\underline{x}_r(t)$ and its steady-state value $\underline{x}_{r_{ss}}$ and $\underline{\widetilde{u}}(t)$ the error between the actual control $\underline{u}(t)$ and its steady-state value $\underline{u}_{ss}$:

$$\begin{cases} \underline{\widetilde{x}}(t) := \underline{x}(t) - \underline{x}_{ss} \\ \underline{\widetilde{x}}_r(t) := \underline{x}_r(t) - \underline{x}_{r_{ss}} \\ \underline{\widetilde{u}}(t) := \underline{u}(t) - \underline{u}_{ss} \end{cases} \tag{5.69}$$

Using (5.69) the dynamics of $\underline{\widetilde{x}}(t)$ reads:

$$\begin{array}{ll} \dot{\underline{\widetilde{x}}}(t) & = \dot{\underline{x}}(t) \\ & = \mathbf{A}_1\underline{x}(t) + \mathbf{B}_1\underline{u}(t) + \mathbf{B}_2\underline{r}_{ss} \\ & = \mathbf{A}_1\left(\underline{\widetilde{x}}(t) + \underline{x}_{ss}\right) + \mathbf{B}_1\left(\underline{\widetilde{u}}(t) + \underline{u}_{ss}\right) + \mathbf{B}_2\underline{r}_{ss} \\ & = \mathbf{A}_1\underline{\widetilde{x}}(t) + \mathbf{B}_1\underline{\widetilde{u}}(t) + \mathbf{A}_1\underline{x}_{ss} + \mathbf{B}_1\underline{u}_{ss} + \mathbf{B}_2\underline{r}_{ss} \end{array} \tag{5.70}$$

Similarly:

$$\begin{array}{ll} \dot{\underline{\widetilde{x}}}_r(t) & = \dot{\underline{x}}_r(t) \\ & = \mathbf{A}_r\underline{x}_r(t) + \mathbf{B}_r\underline{r}_{ss} \\ & = \mathbf{A}_r\left(\underline{\widetilde{x}}_r(t) + \underline{x}_{r_{ss}}\right) + \mathbf{B}_r\underline{r}_{ss} \\ & = \mathbf{A}_r\underline{\widetilde{x}}_r(t) + \mathbf{A}_r\underline{x}_{r_{ss}} + \mathbf{B}_r\underline{r}_{ss} \end{array} \tag{5.71}$$

It is clear from (5.64) that $\mathbf{A}_1\underline{x}_{ss}+\mathbf{B}_1\underline{u}_{ss}+\mathbf{B}_2\underline{r}_{ss} = \underline{0}$ and $\mathbf{A}_r\underline{x}_{r_{ss}}+\mathbf{B}_r\underline{r}_{ss} = \underline{0}$. We finally get the following state space equation:

$$\begin{array}{ll} \begin{bmatrix} \dot{\underline{\widetilde{x}}}(t) \\ \dot{\underline{\widetilde{x}}}_r(t) \end{bmatrix} & = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_r \end{bmatrix} \begin{bmatrix} \underline{\widetilde{x}}(t) \\ \underline{\widetilde{x}}_r(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{0} \end{bmatrix} \underline{\widetilde{u}}(t) \\ & := \mathbf{A}_a \begin{bmatrix} \underline{\widetilde{x}}(t) \\ \underline{\widetilde{x}}_r(t) \end{bmatrix} + \mathbf{B}_a\underline{\widetilde{u}}(t) \end{array} \tag{5.72}$$

In addition, the tracking error defined in (5.63) becomes:

$$\begin{array}{ll} \underline{e}(t) & := \underline{y}(t) - \underline{y}_r(t) \\ & = \mathbf{C}_1\,\underline{x}(t) - \mathbf{C}_r\,\underline{x}_r(t) \\ & = \mathbf{C}_1\,\left(\underline{\widetilde{x}}(t) + \underline{x}_{ss}\right) - \mathbf{C}_r\,\left(\underline{\widetilde{x}}_r(t) + \underline{x}_{r_{ss}}\right) \\ & = \mathbf{C}_1\,\underline{\widetilde{x}}(t) - \mathbf{C}_r\,\underline{\widetilde{x}}_r(t) + \mathbf{C}_1\underline{x}_{ss} - \mathbf{C}_r\underline{x}_{r_{ss}} \end{array} \tag{5.73}$$

Using the last equation of (5.64), we finally get the following output equation:

$$\begin{array}{ll} \mathbf{C}_1\underline{x}_{ss} = \mathbf{C}_r\underline{x}_{r_{ss}} \Rightarrow \underline{e}(t) & = \mathbf{C}_1\,\underline{\widetilde{x}}(t) - \mathbf{C}_r\,\underline{\widetilde{x}}_r(t) \\ & = \begin{bmatrix} \mathbf{C}_1 & -\mathbf{C}_r \end{bmatrix} \begin{bmatrix} \underline{\widetilde{x}}(t) \\ \underline{\widetilde{x}}_r(t) \end{bmatrix} \\ & := \mathbf{C}_a \begin{bmatrix} \underline{\widetilde{x}}(t) \\ \underline{\widetilde{x}}_r(t) \end{bmatrix} \end{array} \tag{5.74}$$

Then we consider the minimization of following performance index :

$$J(\underline{\widetilde{u}}(t)) = \frac{1}{2}\int_0^\infty \underline{e}^T(t)\mathbf{Q}\underline{e}(t) + \underline{\widetilde{u}}^T(t)\mathbf{R}\underline{\widetilde{u}}(t)\,dt \tag{5.75}$$

The minimization of $J(\underline{\widetilde{u}}(t))$ is achieved by applying classical results on LQR problems and control $\underline{\widetilde{u}}(t)$ reads as follows:

$$\underline{\widetilde{u}}(t) = -\mathbf{K}_a \begin{bmatrix} \underline{\widetilde{x}}(t) \\ \underline{\widetilde{x}}_r(t) \end{bmatrix} \text{ where } \mathbf{K}_a = \mathbf{R}^{-1}\mathbf{B}_a^T\mathbf{P} \tag{5.76}$$

Matrix $\mathbf{P}$ is the positive definite solution of the following algebraic Riccati equation:

$$\boxed{\mathbf{0} = \mathbf{A}_a^T\mathbf{P} + \mathbf{P}\mathbf{A}_a - \mathbf{P}\mathbf{B}_a\mathbf{R}^{-1}\mathbf{B}_a^T\mathbf{P} + \mathbf{C}_a^T\mathbf{Q}\mathbf{C}_a} \tag{5.77}$$

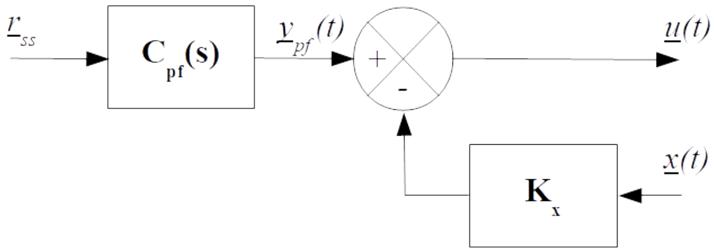


Figure 5.3: Linear Quadratic Tracker with prefilter

The actual control $\underline{u}(t)$ is finally obtained thanks to (5.69):

$$\begin{aligned} \underline{u}(t) &= \widetilde{\underline{u}}(t) + \underline{u}_{ss} \\ &= -\mathbf{K}_a \begin{bmatrix} \widetilde{\underline{x}}(t) \\ \widetilde{\underline{x}}_r(t) \end{bmatrix} + \underline{u}_{ss} \\ &= -\mathbf{K}_a \begin{bmatrix} \underline{x}(t) \\ \underline{x}_r(t) \end{bmatrix} + \mathbf{K}_a \begin{bmatrix} \underline{x}_{ss} \\ \underline{x}_{r_{ss}} \end{bmatrix} + \underline{u}_{ss} \end{aligned} \tag{5.78}$$

That is, when splitting $\mathbf{K}_a$ as $\mathbf{K}_a := \begin{bmatrix} \mathbf{K}_x & \mathbf{K}_r \end{bmatrix}$ and using (5.68):

$$\begin{aligned} \underline{u}(t) &= -\mathbf{K}_a \begin{bmatrix} \underline{x}(t) \\ \underline{x}_r(t) \end{bmatrix} + \mathbf{K}_a \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \end{bmatrix} \underline{r}_{ss} + \mathbf{M}_3\, \underline{r}_{ss} \\ &:= -\begin{bmatrix} \mathbf{K}_x & \mathbf{K}_r \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{x}_r(t) \end{bmatrix} + \mathbf{D}_{pf}\, \underline{r}_{ss} \\ &:= -\mathbf{K}_x\, \underline{x}(t) + \underline{y}_{pf}(t) \end{aligned} \tag{5.79}$$

where:

$$\begin{cases} \underline{y}_{pf}(t) = -\mathbf{K}_r\, \underline{x}_r(t) + \mathbf{D}_{pf}\, \underline{r}_{ss} \\ \mathbf{D}_{pf} := \mathbf{K}_a \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \end{bmatrix} + \mathbf{M}_3 \end{cases} \tag{5.80}$$

Consequently the actual optimal control $\underline{u}(t)$ is the sum of two components:

– a state-feedback component: $-\mathbf{K}_x\, \underline{x}(t)$;

– and a feedforward component $\underline{y}_{pf}(t)$ with is obtained as the output of the a prefilter $\mathbf{C}_{pf}(s)$ with the following realization:

$$\mathbf{C}_{pf}(s) : \begin{cases} \dot{\underline{x}}_r(t) = \mathbf{A}_r \underline{x}_r(t) + \mathbf{B}_r \underline{r}_{ss} \\ \underline{y}_{pf}(t) = -\mathbf{K}_r\, \underline{x}_r(t) + \mathbf{D}_{pf}\, \underline{r}_{ss} \end{cases} \tag{5.81}$$

This is illustrated in Figure 5.3.

5.6.2 Tracking with integral augmentation

Assuming that an integrator is inserted in the feedforward path of the loop, as presented in Section 5.5, the state vector $\underline{x}(t)$ of the plant has to be extended by adding a new component $\underline{x}_i(t)$ in the state vector:

$$\underline{x}(t) \to \begin{bmatrix} \underline{x}(t) \\ \underline{x}_i(t) \end{bmatrix} \text{ where } \dot{\underline{x}}_i(t) = \mathbf{T}\underline{e}(t) = \mathbf{T}\left(\underline{r}(t) - \underline{y}(t)\right) \tag{5.82}$$

Then using (5.44) matrices $\mathbf{A}_1$, $\mathbf{B}_1$ and $\mathbf{C}_1$ in (5.61) read as follows:

$$\begin{cases} \mathbf{A}_1 := \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{TC} & \mathbf{0} \end{bmatrix} \\ \mathbf{B}_1 := \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \\ \mathbf{B}_2 := \begin{bmatrix} \mathbf{0} \\ \mathbf{T} \end{bmatrix} \\ \mathbf{C}_1 := \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} \end{cases} \tag{5.83}$$

Nevertheless the algebraic Riccati equation (5.77) is not solvable because pair $(\mathbf{A}_a, \mathbf{C}_a)$ is no more observable. In order to tackle this point, weighting matrix $\mathbf{C}_a := \begin{bmatrix} \mathbf{C}_1 & -\mathbf{C}_r \end{bmatrix} = \begin{bmatrix} \mathbf{C} & \mathbf{0} & -\mathbf{C}_r \end{bmatrix}$ has to be changed, for example as follows:

$$\mathbf{C}_a := [\mathbf{C} \quad \underbrace{\mathbb{I}}_{\mathbf{0} \text{ becomes } \mathbb{I}} \quad -\mathbf{C}_r] \tag{5.84}$$

Finally steady-state value of the integral term $\underline{x}_i(t)$ is $\underline{x}_{i_{ss}} = \underline{0}$. Consequently, tracking with integral augmentation will not the change steady-state values and (5.68) is still valid (meaning that $\mathbf{B}_2$ is assumed to be zero to compute steady-state values). This implies that the expression of the structure of the prefilter remains unchanged. Finally (5.80) now reads as follows where matrix $\mathbf{0}$ has been added after $\mathbf{M}_1$ in the expression of $\mathbf{D}_{pf}$ to take into account the presence of integral term in the state vector of the augmented plant:

$$\begin{cases} \underline{y}_{pf}(t) = -\mathbf{K}_r\, \underline{x}_r(t) + \mathbf{D}_{pf}\, \underline{r}_{ss} \\ \mathbf{D}_{pf} := \mathbf{K}_a \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{0} \\ \mathbf{M}_2 \end{bmatrix} + \mathbf{M}_3 \end{cases} \tag{5.85}$$

Chapter 6

Linear Quadratic Gaussian (LQG) regulator

6.1 Introduction

The design of the Linear Quadratic Regulator (LQR) assumes that the whole state is available for control and that there is no noise. Those assumptions may appear unrealistic in practical applications. We will assume in this chapter that the process to be controlled is described by the following linear time invariant model where $\underline{w}(t)$ and $\underline{v}(t)$ are random processes which represents the process noise and the measurement noise, respectively:

$$\left\{ \begin{array}{l} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) + \underline{w}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t) + \underline{v}(t) \end{array} \right. \tag{6.1}$$

The preceding relation can be equivalently represented by the block diagram in Figure 6.1.

Linear Quadratic Gaussian (LQG) control deals with the design of a regulator which minimizes a quadratic cost using the available output and taking into account the noise into the process and the available output for control. More precisely the LQG control problem is to find the optimal control $\underline{u}(t)$ which minimizes the following performance index $J(\underline{u}(t))$ where $E()$ is the

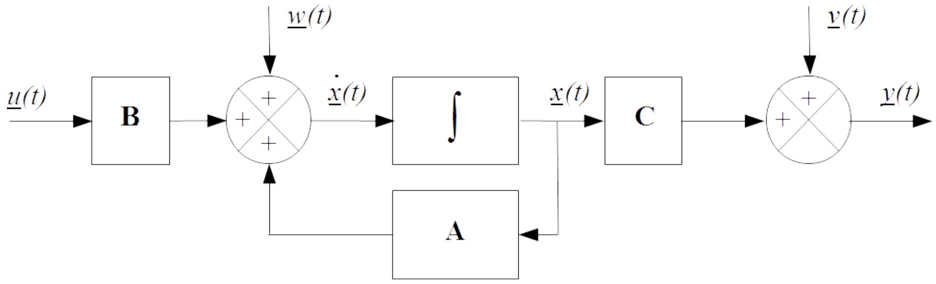


Figure 6.1: Open-loop linear system with process and measurement noises

mathematical expectation, $\mathbf{Q} = \mathbf{Q}^T \geq 0$ and $\mathbf{R} = \mathbf{R}^T > 0$:

$$J(\underline{u}(t)) = E\left(\lim_{t_f \to \infty} \frac{1}{2\,t_f} \int_0^{t_f} \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\,dt\right) \tag{6.2}$$

As far as only the output $\underline{y}(t)$ is now available for control (not the full state $\underline{x}(t)$), the separation principle will be used to design the LQG regulator. Indeed, the solution of the LQG problem can be split into two steps:

- First an estimator will be used to estimate the full state using the available output $\underline{y}(t)$
- Then an LQ controller will be designed using the state estimation in place of the true (but unknown) state $\underline{x}(t)$

6.2 Luenberger observer

Consider a process with the following state space model where $\underline{y}(t)$ denotes the measured output and $\underline{u}(t)$ the control input:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t) \end{cases} \tag{6.3}$$

We assume that $\underline{x}(t)$ cannot be measured and the goal of the observer is to estimate $\underline{x}(t)$ based on $\underline{y}(t)$. Luenberger observer (1964) provides an estimation of the state vector through the following differential equation where matrices $\mathbf{F}$, $\mathbf{J}$ and $\mathbf{L}$ have to be determined:

$$\frac{d}{dt}\widehat{\underline{x}}(t) = \mathbf{F}\widehat{\underline{x}}(t) + \mathbf{J}\underline{u}(t) + \mathbf{L}\underline{y}(t) \tag{6.4}$$

The estimation error $\underline{e}(t)$ is defined as follows:

$$\underline{e}(t) = \underline{x}(t) - \widehat{\underline{x}}(t) \tag{6.5}$$

Thus using (6.3) and (6.4) its time derivative reads:

$$\dot{\underline{e}}(t) = \dot{\underline{x}}(t) - \dot{\widehat{\underline{x}}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) - \left(\mathbf{F}\widehat{\underline{x}}(t) + \mathbf{J}\underline{u}(t) + \mathbf{L}\underline{y}(t)\right) \tag{6.6}$$

Using (6.5) and the output equation $\underline{y}(t) = \mathbf{C}\underline{x}(t)$ the preceding relation can be rewritten as follows:

$$\begin{array}{rl} \dot{\underline{e}}(t) & = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) - \mathbf{F}\left(\underline{x}(t) - \underline{e}(t)\right) - \mathbf{J}\underline{u}(t) - \mathbf{LC}\underline{x}(t) \\ & = \mathbf{F}\underline{e}(t) + \left(\mathbf{A} - \mathbf{F} - \mathbf{LC}\right)\underline{x}(t) + \left(\mathbf{B} - \mathbf{J}\right)\underline{u}(t) \end{array} \tag{6.7}$$

As soon as the purpose of the observer is to move the estimation error $\underline{e}(t)$ towards zero independently of control $\underline{u}(t)$ and *true* state vector $\underline{x}(t)$ we choose matrices $\mathbf{F}$ and $\mathbf{J}$ as follows:

$$\begin{cases} \mathbf{J} = \mathbf{B} \\ \mathbf{F} = \mathbf{A} - \mathbf{LC} \end{cases} \tag{6.8}$$

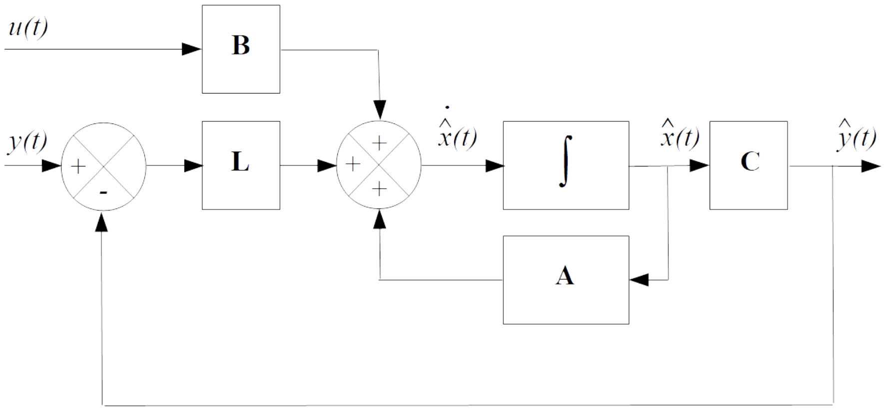


Figure 6.2: Luenberger observer

Thus the dynamics of the estimation error $\underline{e}(t)$ reduces to be:

$$\dot{\underline{e}}(t) = \mathbf{F}\underline{e}(t) = (\mathbf{A} - \mathbf{LC})\,\underline{e}(t) \tag{6.9}$$

Where matrix $\mathbf{L}$ shall be chosen such that all the eigenvalues of $\mathbf{A} - \mathbf{LC}$ are situated in the left half plane. Furthermore the Luenberger observer (6.4) can now be written as follows using (6.8):

$$\begin{aligned} \dot{\widehat{\underline{x}}}(t) &= (\mathbf{A} - \mathbf{LC})\,\widehat{\underline{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}\underline{y}(t) \\ &= \mathbf{A}\widehat{\underline{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}\left(\underline{y}(t) - \mathbf{C}\widehat{\underline{x}}(t)\right) \end{aligned} \tag{6.10}$$

Figure 6.2 shows the structure of the Luenberger observer.

6.3 White noise through Linear Time Invariant (LTI) system

6.3.1 Assumptions and definitions

Let's consider the following linear time invariant system which is fed by a random process $\underline{w}(t)$ of dimension n (which is also the dimension of the state vector $\underline{x}(t)$):

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{w}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t) \end{cases} \tag{6.11}$$

The mean of $\underline{w}(t)$ will be denoted $E\left[\underline{w}(t)\right]$ and its autocorrelation function $\mathbf{R}_w(\tau)$ will be denoted $E\left[\underline{w}(t)\underline{w}^T(t+\tau)\right]$ where E designates the expectation operator. We said that $\underline{w}(t)$ is a wide-sense stationary (WSS) random process when the two following properties hold:

- The mean $\underline{m}_w(t) := E\left[\underline{w}(t)\right]$ of $\underline{w}(t)$ is independent of t, that is constant;
- The autocorrelation function $\mathbf{R}_w(t, t+\tau)$ just depends on the time difference $\tau = (t+\tau) - t$:

$$\begin{aligned} \mathbf{R}_w(t, t+\tau) &= E\left[\left(\underline{w}(t) - \underline{m}_w(t)\right)\left(\underline{w}(t+\tau) - \underline{m}_w(t+\tau)\right)^T\right] \\ &:= \mathbf{R}_w(\tau) \end{aligned} \tag{6.12}$$

We will assume that $\underline{w}(t)$ is a white noise (which is a special case of a wide-sense stationary (WSS) random process) with zero mean Gaussian probability density function (pdf) $p(\underline{w})$. The covariance matrix of the Gaussian probability density function $p(\underline{w})$ will be denoted $\mathbf{P}_w$ and the Dirac delta function will be denoted $\delta(\tau)$:

$$\begin{cases} p(\underline{w}) = \frac{1}{(2\pi)^{n/2}\sqrt{\det(\mathbf{P}_w)}} e^{-\frac{1}{2}\underline{w}^T \mathbf{P}_w^{-1} \underline{w}} \\ E\left[\underline{w}(t)\right] = \underline{m}_w(t) = \underline{0} \\ \mathbf{R}_w(\tau) = E\left[\underline{w}(t)\underline{w}^T(t+\tau)\right] = \mathbf{P}_w\,\delta(\tau) \text{ where } \mathbf{P}_w = \mathbf{P}_w^T > 0 \end{cases} \tag{6.13}$$

Because $\underline{w}(t)$ is a stochastic process, the differential equation $\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{w}(t)$ is called a stochastic differential equation. Moreover this particular type of stochastic differential equation where $\underline{w}(t)$ comes from the derivative of a Wiener process is called Langevin equation and can be written more elegantly as the following Ornstein-Uhlenbeck process where $\underline{w}(t)$ is a vector of Wiener process, also called Brownian motion:

$$d\underline{x}(t) = \mathbf{A}\underline{x}(t)\,dt + \mathbf{B}\,d\underline{w}(t) \tag{6.14}$$

6.3.2 Mean and covariance matrix of the state vector

As far as $\underline{w}(t)$ is a random process it is clear from (6.11) that the state vector $\underline{x}(t)$ and the output vector $\underline{y}(t)$ are also a random processes. When expending the expression of the state vector obtained for deterministic signals we get:

$$\underline{x}(t) = e^{\mathbf{A}t}\underline{x}_0 + \int_0^t e^{\mathbf{A}(t-\tau)}\,\mathbf{B}\,\underline{w}(\tau)\,d\tau \tag{6.15}$$

Let $\underline{m}_x(0) = E\left[\underline{x}_0\right]$ be the mean of the initial value $\underline{x}_0$ of the state vector $\underline{x}(t)$ and $\mathbf{P}_x(0)$ the covariance matrix of the initial value $\underline{x}_0$ of the state vector. Then it can be shown that $\underline{x}(t)$ is a Gaussian random process with:

– Mean $\underline{m}_x(t)$ given by:

$$\underline{m}_x(t) = E\left[\underline{x}(t)\right] = e^{\mathbf{A}t}\underline{m}_x(0) \tag{6.16}$$

Assuming that $\underline{m}_x(0) = \underline{0}$ we get zero for the mean value of $\underline{x}(t)$::

$$\underline{m}_x(0) = \underline{0} \Rightarrow \underline{m}_x(t) = \underline{0} \tag{6.17}$$

– Covariance matrix $\mathbf{P}_x(t)$ which is defined as follows:

$$\mathbf{P}_x(t) = E\left[\left(\underline{x}(t) - \underline{m}_x(t)\right)\left(\underline{x}(t) - \underline{m}_x(t)\right)^T\right] \tag{6.18}$$

Assuming that $\underline{m}_x(0) = \underline{0}$ we get:

$$\underline{m}_x(0) = \underline{0} \Rightarrow \mathbf{P}_x(t) = E\left[\underline{x}(t)\,\underline{x}(t)^T\right] \tag{6.19}$$

Finally, assuming that $\underline{m}_x(0) = \underline{0}$ and the input random process $\underline{w}(t)$ is a zero mean white noise with autocorrelation function $\mathbf{R}_w(\tau) = \mathbf{P}_w\,\delta(\tau)$ and is uncorrelated with the initial value $\underline{x}_0$ of the state vector, that is $E\left[\underline{x}_0\,\underline{w}^T(\tau)\right] = \mathbf{0}$, then matrix $\mathbf{P}_x(t)$ reads as follows:

$$\begin{array}{ll}\mathbf{P}_x(t) & = E\left[\underline{x}(t)\,\underline{x}(t)^T\right] \\ & = E\left[\left(e^{\mathbf{A}t}\underline{x}_0 + \int_0^t e^{\mathbf{A}(t-\tau)}\,\mathbf{B}\,\underline{w}(\tau)\,d\tau\right)\left(e^{\mathbf{A}t}\underline{x}_0 + \int_0^t e^{\mathbf{A}(t-\tau)}\,\mathbf{B}\,\underline{w}(\tau)\,d\tau\right)^T\right] \\ & = e^{\mathbf{A}t}\mathbf{P}_x(0)e^{\mathbf{A}^Tt} + E\left[\left(\int_0^t e^{\mathbf{A}(t-\tau_1)}\,\mathbf{B}\,\underline{w}(\tau_1)\,d\tau_1\right)\left(\int_0^t e^{\mathbf{A}(t-\tau_2)}\,\mathbf{B}\,\underline{w}(\tau_2)\,d\tau_2\right)^T\right] \\ & = e^{\mathbf{A}t}\mathbf{P}_x(0)e^{\mathbf{A}^Tt} + \int_0^t\int_0^t e^{\mathbf{A}(t-\tau)}\,\mathbf{B}\,E\left[\underline{w}(\tau)\underline{w}^T(\tau_1)\right]\mathbf{B}^T e^{\mathbf{A}^T(t-\tau_1)}\,d\tau\,d\tau_1 \\ & = e^{\mathbf{A}t}\mathbf{P}_x(0)e^{\mathbf{A}^Tt} + \int_0^t\int_0^t e^{\mathbf{A}(t-\tau)}\,\mathbf{B}\,\mathbf{P}_w\,\delta(\tau_1-\tau)\,\mathbf{B}^T e^{\mathbf{A}^T(t-\tau_1)}\,d\tau\,d\tau_1\end{array} \tag{6.20}$$

Using the fact that $\int_0^t g(\tau_1)\,\delta(\tau_1-\tau)\,d\tau_1 = g(\tau)$, we finally obtain:

$$\mathbf{P}_x(t) = e^{\mathbf{A}t}\mathbf{P}_x(0)e^{\mathbf{A}^Tt} + \int_0^t e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{P}_w\mathbf{B}^T e^{\mathbf{A}^T(t-\tau)}d\tau \tag{6.21}$$

Because the evaluation of the preceding integral is difficult, we take the derivative of $\mathbf{P}_x(t)$ to get the following Lyapunov matrix differential equation where $\mathbf{P}_x(t) = \mathbf{P}_x^T(t) \geq 0$:

$$\dot{\mathbf{P}}_x(t) = \mathbf{A}\mathbf{P}_x(t) + \mathbf{P}_x(t)\mathbf{A}^T + \mathbf{B}\mathbf{P}_w\mathbf{B}^T \tag{6.22}$$

Assuming that the system is *stable* (i.e. all the eigenvalues of the state matrix $\mathbf{A}$ have negative real part) the random process $\underline{x}(t)$ will become stationary after a certain amount of time: its mean $\underline{m}_x(t)$ will be zero whereas the value of its covariance matrix $\mathbf{P}_x(t)$ turns to be a constant matrix $\mathbf{P}_x = \mathbf{P}_x^T \geq 0\,\forall\,t$ which solves the following matrix algebraic Lyapunov equation:

$$\mathbf{A}\mathbf{P}_x + \mathbf{P}_x\mathbf{A}^T + \mathbf{B}\mathbf{P}_w\mathbf{B}^T = \mathbf{0} \tag{6.23}$$

Thus after a certain amount of time the state vector $\underline{x}(t)$ as well as the output vector $\underline{y}(t)$ are wide-sense stationary (WSS) random processes.

6.3.3 Autocorrelation function of the stationary output vector

As in the previous section, we will assume in the following that $\underline{m}_x(0) = \underline{0}$. Let $\mathbf{R}_x(\tau)$ be the autocorrelation function of the stationary state vector $\underline{x}(t)$:

$$\mathbf{R}_x(\tau) = E\left[\underline{x}(t)\,\underline{x}(t+\tau)^T\right] \tag{6.24}$$

The autocorrelation function $\mathbf{R}_y(\tau)$ (which may be a matrix for vector signal) of the output vector $\underline{y}(t) = \mathbf{C}\,\underline{x}(t)$ is defined as follows:

$$\mathbf{R}_y(\tau) = E\left[\underline{y}(t)\,\underline{y}(t+\tau)^T\right] = \mathbf{C}\mathbf{R}_x(\tau)\mathbf{C}^T \tag{6.25}$$

It is clear from the definition of the autocorrelation function $\mathbf{R}_y(\tau)$ that the stationary value of the covariance matrix $\mathbf{P}_y$ of $\underline{y}(t)$ is equal to the value of the autocorrelation function $\mathbf{R}_y(\tau)$ at $\tau = 0$:

$$\mathbf{P}_y = E\left[\underline{y}(t)\,\underline{y}(t)^T\right] = \mathbf{C}\mathbf{P}_x\mathbf{C}^T = \left.\mathbf{R}_y(\tau)\right|_{\tau=0} \tag{6.26}$$

The power spectral density (psd) $\mathbf{S}_y(f)$ of a stationary process $\underline{y}(t)$ is given by the Fourier transform of its autocorrelation function $\mathbf{R}_y(\tau)$:

$$\mathbf{S}_y(f) = \int_{-\infty}^{+\infty} \mathbf{R}_y(\tau) e^{-j2\pi f\tau} d\tau \tag{6.27}$$

Then we will see in Section 6.3.4 that the following result holds:

$$\boxed{\mathbf{S}_y(f) = \left.\mathbf{F}(-s)\,\mathbf{P}_w\,\mathbf{F}^T(s)\right|_{s=j2\pi f}} \tag{6.28}$$

where $\mathbf{F}(s)$ is the transfer function of the linear system, which is assumed to be *stable*:

$$\mathbf{F}(s) = \mathbf{C}\left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B} \tag{6.29}$$

Relation (6.28) indicates that the power spectral density (psd) $\mathbf{S}_y(f)$ of $\underline{y}(t)$ can be obtained thanks to the transfer function $\mathbf{F}(s)$ of the *stable* linear system and the spectral density matrix $\mathbf{P}_w$ of the exciting white noise $\underline{w}(t)$.

Let $\mathbf{S}_y(s)$ be the (one-sided) Laplace transform of the autocorrelation function $\mathbf{R}_y(\tau)$:

$$\mathbf{S}_y(s) = \mathcal{L}\left[\mathbf{R}_y(\tau)\right] = \int_{0}^{+\infty} \mathbf{R}_y(\tau) e^{-s\tau} d\tau \tag{6.30}$$

It can be seen that the power spectral density (psd) $\mathbf{S}_y(f)$ of $\underline{y}(t)$ can be obtained thanks to the (one-sided) Laplace transform $\mathbf{S}_y(s)$ of $\mathbf{R}_y(\tau)$ as:

$$\mathbf{S}_y(f) \;\; = \left.\mathbf{S}_y(-s)\right|_{s=j2\pi f} + \left.\mathbf{S}_y(s)\right|_{s=j2\pi f} \tag{6.31}$$

Indeed we can write:

$$\begin{array}{ll} \mathbf{S}_y(f) & = \int_{-\infty}^{+\infty} \mathbf{R}_y(\tau) e^{-j2\pi f\tau} d\tau \\ & = \int_{-\infty}^{0} \mathbf{R}_y(\tau) e^{-j2\pi f\tau} d\tau + \int_{0}^{+\infty} \mathbf{R}_y(\tau) e^{-j2\pi f\tau} d\tau \\ & = \left.\int_{-\infty}^{0} \mathbf{R}_y(\tau) e^{-s\tau} d\tau\right|_{s=j2\pi f} + \left.\int_{0}^{+\infty} \mathbf{R}_y(\tau) e^{-s\tau} d\tau\right|_{s=j2\pi f} \\ & = \left.\int_{0}^{+\infty} \mathbf{R}_y(-\tau) e^{s\tau} d\tau\right|_{s=j2\pi f} + \left.\int_{0}^{+\infty} \mathbf{R}_y(\tau) e^{-s\tau} d\tau\right|_{s=j2\pi f} \end{array} \tag{6.32}$$

As far as $\mathbf{R}_y(\tau)$ is an even function we get:

$$\begin{array}{c} \mathbf{R}_y(-\tau) = \mathbf{R}_y(\tau) \\ \Rightarrow \mathbf{S}_y(f) = \left.\int_{0}^{+\infty} \mathbf{R}_y(\tau) e^{s\tau} d\tau\right|_{s=j2\pi f} + \left.\int_{0}^{+\infty} \mathbf{R}_y(\tau) e^{-s\tau} d\tau\right|_{s=j2\pi f} \end{array} \tag{6.33}$$

The preceding equations reads:

$$\mathbf{S}_y(f) \;\; = \left.\mathbf{S}_y(-s)\right|_{s=j2\pi f} + \left.\mathbf{S}_y(s)\right|_{s=j2\pi f} \tag{6.34}$$

Then, using (6.28), we can write:

$$\mathbf{S}_y(-s) + \mathbf{S}_y(s) = \mathbf{F}(-s)\,\mathbf{P}_w\,\mathbf{F}^T(s) \tag{6.35}$$

When identifying the *stable* transfer function $\mathbf{S}_y(s)$ in the preceding relation, we get the autocorrelation function $\mathbf{R}_y(\tau)\,\forall\tau \geq 0$ thank to the inverse (one-sided) Laplace transform of $\mathbf{S}_y(s)$:

$$\boxed{\begin{array}{c} \mathbf{S}_y(-s) + \mathbf{S}_y(s) = \mathbf{F}(-s)\,\mathbf{P}_w\,\mathbf{F}^T(s) \\ \Rightarrow \mathbf{R}_y(\tau) = \mathcal{L}^{-1}\left[\mathbf{S}_y(s)\right]\,\forall\tau \geq 0 \text{ where } \mathbf{S}_y(s) \textit{ stable} \end{array}} \tag{6.36}$$

Finally using the initial value theorem on the (one-sided) Laplace transform $\mathbf{S}_y(s)$ we get the following result:

$$\mathbf{P}_y = \mathbf{R}_y(\tau)|_{\tau=0} = \lim_{s\to\infty} s\,\mathbf{S}_y(s) \tag{6.37}$$

Example 6.1. *Let $\mathbf{F}(s)$ be a first order system with time constant a and let $\underline{w}(t)$ be a white noise with covariance $\mathbf{P}_w$:*

$$\left\{\begin{array}{l} \mathbf{F}(s) = \frac{1}{1+as} \\ \mathbf{R}_w(\tau) = E\left[\underline{w}(t)\underline{w}^T(t+\tau)\right] = \mathbf{P}_w\,\delta(\tau) \textit{ where } \mathbf{P}_w = \mathbf{P}_w^T > 0 \end{array}\right. \tag{6.38}$$

One realization of transfer function $\mathbf{F}(s)$ is the following:

$$\left\{\begin{array}{l} \dot{x}(t) = -\frac{1}{a}x(t) + \underline{w}(t) \\ y(t) = \frac{1}{a}x(t) \end{array}\right. \tag{6.39}$$

That is:

$$\left\{\begin{array}{l} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{w}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t) \end{array}\right. \tag{6.40}$$

Where:

$$\left\{\begin{array}{l} \mathbf{A} = -\frac{1}{a} \\ \mathbf{B} = 1 \\ \mathbf{C} = \frac{1}{a} \end{array}\right. \Rightarrow \mathbf{F}(s) = \mathbf{C}(s\mathbb{I} - \mathbf{A})^{-1}\mathbf{B} \tag{6.41}$$

As far as $a > 0$ the system is stable. The covariance matrix $\mathbf{P}_x(t)$ is defined as follows:

$$\mathbf{P}_x(t) = E\left[\left(\underline{x}(t) - \underline{m}_x(t)\right)\left(\underline{x}(t) - \underline{m}_x(t)\right)^T\right] \tag{6.42}$$

Where matrix $\mathbf{P}_x(t)$ is the solution of the following Lyapunov differential equation:

$$\dot{\mathbf{P}}_x(t) = \mathbf{A}\mathbf{P}_x(t) + \mathbf{P}_x(t)\mathbf{A}^T + \mathbf{B}\mathbf{P}_w\mathbf{B}^T = -\frac{2}{a}\mathbf{P}_x(t) + \mathbf{P}_w \tag{6.43}$$

We get:

$$\mathbf{P}_x(t) = \frac{a}{2}\mathbf{P}_w + \left(\mathbf{P}_x(0) - \frac{a}{2}\mathbf{P}_w\right)e^{-\frac{2t}{a}} \tag{6.44}$$

The stationary value $\mathbf{P}_x$ of the covariance matrix $\mathbf{P}_x(t)$ of the state vector $\underline{x}(t)$ is obtained as $t \to \infty$:

$$\mathbf{P}_x = \lim_{t\to\infty}\mathbf{P}_x(t) = \frac{a}{2}\mathbf{P}_w \tag{6.45}$$

Consequently the stationary value $\mathbf{P}_y$ of the covariance matrix of the output vector $\underline{y}(t)$ reads:

$$\mathbf{P}_y = \mathbf{C}\mathbf{P}_x\mathbf{C}^T = \frac{1}{a^2} \times \frac{a}{2}\mathbf{P}_w = \frac{\mathbf{P}_w}{2a} \tag{6.46}$$

This result can be retrieved thanks to the power spectral density (psd) of the output vector $\underline{y}(t)$. Indeed let's compute the power spectral density (psd) $\mathbf{S}_y(f)$ of the output stationary process $\underline{y}(t)$ of the system:

$$\mathbf{S}_y(f) = \int_{-\infty}^{+\infty} \mathbf{R}_y(\tau)e^{-j2\pi f\tau}d\tau = \mathbf{F}(-s)\,\mathbf{P}_w\,\mathbf{F}^T(s)\big|_{s=j2\pi f} \tag{6.47}$$

We get:

$$\begin{array}{l} \mathbf{F}(-s)\,\mathbf{P}_w\,\mathbf{F}^T(s) = \frac{\mathbf{P}_w}{(1+as)(1-as)} = \frac{\mathbf{P}_w}{1-(as)^2} \\ \Rightarrow \mathbf{S}_y(f) = \mathbf{F}(-s)\,\mathbf{P}_w\,\mathbf{F}^T(s)\big|_{s=j2\pi f} = \frac{\mathbf{P}_w}{1+(2\pi f a)^2} \end{array} \tag{6.48}$$

Furthermore let's decompose $\frac{\mathbf{P}_w}{1-(as)^2}$ as the sum $\mathbf{S}_y(-s) + \mathbf{S}_y(s)$:

$$\frac{\mathbf{P}_w}{1-(as)^2} = \frac{\mathbf{P}_w}{2}\frac{1}{1-as} + \frac{\mathbf{P}_w}{2}\frac{1}{1+as} = \mathbf{S}_y(-s) + \mathbf{S}_y(s) \tag{6.49}$$

Thus by identification we get for the stable *transfer function $\mathbf{S}_y(s)$:*

$$\mathbf{S}_y(s) = \frac{\mathbf{P}_w}{2}\frac{1}{1+as} \tag{6.50}$$

The autocorrelation function $\mathbf{R}_y(\tau)$ is given by the inverse Laplace transform of $\mathbf{S}_y(s)$:

$$\mathbf{R}_y(\tau) = \mathcal{L}^{-1}\left[\mathbf{S}_y(s)\right] = \mathcal{L}^{-1}\left[\frac{\mathbf{P}_w}{2a}\frac{1}{1/a+s}\right] = \frac{\mathbf{P}_w}{2a}e^{-\frac{\tau}{a}} \;\forall\; \tau \geq 0 \tag{6.51}$$

As far as $\mathbf{R}_y(\tau)$ is an even function we get:

$$\mathbf{R}_y(\tau) = \frac{\mathbf{P}_w}{2a}e^{\frac{\tau}{a}} \;\forall\; \tau \leq 0 \tag{6.52}$$

Thus the autocorrelation function $\mathbf{R}_y(\tau)$ for $\tau \in \mathbb{R}$ reads:

$$\mathbf{R}_y(\tau) = \frac{\mathbf{P}_w}{2a}e^{-\frac{|\tau|}{a}} \;\forall\; \tau \in \mathbb{R} \tag{6.53}$$

Finally we use the initial value theorem on the (one-sided) Laplace transform $\mathbf{S}_y(s)$ to get the following result:

$$\mathbf{P}_y = \mathbf{R}_y(\tau)\big|_{\tau=0} = \lim_{s\to\infty} s\,\mathbf{S}_y(s) = \frac{\mathbf{P}_w}{2a} \tag{6.54}$$

■

6.3.4 Proof of the expression of the autocorrelation function

The autocorrelation function $\mathbf{R}_y(t, t+\tau)$ of a non-stationary process reads as follows:

$$
\begin{array}{ll}
\mathbf{R}_y(t, t+\tau) & = E\left[\underline{y}(t)\, \underline{y}^T(t+\tau)\right] \\
& = E\left[\mathbf{C}\, \underline{x}(t)\, \underline{x}^T(t+\tau) \mathbf{C}^T\right] \\
& = \mathbf{C}\, E\left[\underline{x}(t)\, \underline{x}^T(t+\tau)\right]\, \mathbf{C}^T \\
& = \mathbf{C}\, E\left[\left(\int_0^t e^{\mathbf{A}(t-\tau_1)}\, \mathbf{B}\, \underline{w}(\tau_1)\, d\tau_1\right) \left(\int_0^{t+\tau} e^{\mathbf{A}(t+\tau-\tau_2)}\, \mathbf{B}\, \underline{w}(\tau_2)\, d\tau_2\right)^T\right] \mathbf{C}^T \\
& = \mathbf{C} \left(\int_0^t \int_0^{t+\tau} e^{\mathbf{A}(t-\tau_1)}\, \mathbf{B}\, E\left[\underline{w}(\tau_1)\, \underline{w}^T(\tau_2)\right]\, \mathbf{B}^T\, e^{\mathbf{A}^T(t+\tau-\tau_2)}\, d\tau_1\, d\tau_2\right) \mathbf{C}^T
\end{array}
\tag{6.55}
$$

Using the facts that $\underline{w}(t)$ is a white noise, that is $E\left[\underline{w}(\tau_1)\, \underline{w}^T(\tau_2)\right] = \mathbf{P}_w\, \delta(\tau_2 - \tau_1)$, and that $\int_0^t g(\tau_1)\, \delta(\tau_2 - \tau_1)\, d\tau_1 = g(\tau_2)$, we get:

$$
\begin{array}{ll}
\mathbf{R}_y(t, t+\tau) & = \mathbf{C} \left(\int_0^t \int_0^{t+\tau} e^{\mathbf{A}(t-\tau_1)}\, \mathbf{B}\, \mathbf{P}_w\, \delta(\tau_2 - \tau_1)\, \mathbf{B}^T\, e^{\mathbf{A}^T(t+\tau-\tau_2)}\, d\tau_1\, d\tau_2\right) \mathbf{C}^T \\
& = \mathbf{C} \left(\int_0^t e^{\mathbf{A}(t-\tau_2)}\, \mathbf{B}\, \mathbf{P}_w\, \mathbf{B}^T\, e^{\mathbf{A}^T(t+\tau-\tau_2)}\, d\tau_2\right) \mathbf{C}^T
\end{array}
\tag{6.56}
$$

Let $\xi := t - \tau_2 \Rightarrow d\xi = -d\tau_2$. Then:

$$
\begin{array}{ll}
\mathbf{R}_y(t, t+\tau) & = -\mathbf{C} \left(\int_t^0 e^{\mathbf{A}\xi}\, \mathbf{B}\, \mathbf{P}_w\, \mathbf{B}^T\, e^{\mathbf{A}^T(\xi+\tau)}\, d\xi\right) \mathbf{C}^T \\
& = \mathbf{C} \left(\int_0^t e^{\mathbf{A}\xi}\, \mathbf{B}\, \mathbf{P}_w\, \mathbf{B}^T\, e^{\mathbf{A}^T(\xi+\tau)}\, d\xi\right) \mathbf{C}^T
\end{array}
\tag{6.57}
$$

We finally get[1]:

$$
\boxed{\mathbf{R}_y(t, t+\tau) = \int_0^t \left(\mathbf{C}\, e^{\mathbf{A}\xi}\, \mathbf{B}\right)\, \mathbf{P}_w\, \left(\mathbf{C}\, e^{\mathbf{A}(\xi+\tau)}\, \mathbf{B}\right)^T\, d\xi}
\tag{6.58}
$$

Because t is the upper limit of the integral, the preceding autocorrelation function $\mathbf{R}_y(t, t+\tau)$ is not stationary. But stationary comes at $t \to \infty$ when the process reaches steady state:

$$
\lim_{t \to \infty} \mathbf{R}_y(t, t+\tau) = \int_0^\infty \left(\mathbf{C}\, e^{\mathbf{A}\xi}\, \mathbf{B}\right)\, \mathbf{P}_w\, \left(\mathbf{C}\, e^{\mathbf{A}(\xi+\tau)}\, \mathbf{B}\right)^T\, d\xi := \mathbf{R}_y(\tau)
\tag{6.59}
$$

Of course (6.59) is valid only if the process has a steady state response, that is if the process is *stable*.

Then the power spectral density (psd) $\mathbf{S}_y(f)$ of $\underline{y}(t)$ is defined as the Fourier transform of the autocorrelation function $\mathbf{R}_y(\tau)$. We get from (6.59):

$$
\begin{array}{ll}
\mathbf{S}_y(f) & = \int_{-\infty}^{+\infty} \mathbf{R}_y(\tau) e^{-j2\pi f\tau} d\tau \\
& = \int_{-\infty}^{+\infty} \left(\int_0^\infty \left(\mathbf{C}\, e^{\mathbf{A}\xi}\, \mathbf{B}\right)\, \mathbf{P}_w\, \left(\mathbf{C}\, e^{\mathbf{A}(\xi+\tau)}\, \mathbf{B}\right)^T\, d\xi\right) e^{-j2\pi f\tau} d\tau \\
& = \int_0^\infty \left(\mathbf{C}\, e^{\mathbf{A}\xi}\, \mathbf{B}\right)\, \mathbf{P}_w \left(\int_{-\infty}^{+\infty} \left(\mathbf{C}\, e^{\mathbf{A}(\xi+\tau)}\, \mathbf{B}\right)^T\, e^{-j2\pi f\tau} d\tau\right) d\xi
\end{array}
\tag{6.60}
$$

[1] Friedland B., Control System Design: An Introduction to State-Space Methods, Dover Books on Electrical Engineering (2012)

Replacing $\xi + \tau$ by t in the bracketed integral yields:

$$\begin{array}{ll}\int_{-\infty}^{+\infty} \left(\mathbf{C}\, e^{\mathbf{A}(\xi+\tau)}\, \mathbf{B}\right)^T e^{-j2\pi f \tau} d\tau & = \int_{-\infty}^{+\infty} \left(\mathbf{C}\, e^{\mathbf{A}t}\, \mathbf{B}\right)^T e^{-j2\pi f (t-\xi)} dt \\ & = e^{j2\pi f \xi} \int_{-\infty}^{+\infty} \left(\mathbf{C}\, e^{\mathbf{A}t}\, \mathbf{B}\right)^T e^{-j2\pi f t} dt \\ & = e^{j2\pi f \xi}\, \mathbf{F}^T(j2\pi f) \end{array} \tag{6.61}$$

where $\mathbf{F}(j2\pi f)$ is defined as follows:

$$\mathbf{F}(j2\pi f) = \int_{-\infty}^{+\infty} \mathbf{C}\, e^{\mathbf{A}t}\, \mathbf{B}\, e^{-j2\pi f t} dt \tag{6.62}$$

It is worth noticing that the time response of a *causal* system is zero $\forall t < 0$. So we recognize in $\mathbf{F}(j2\pi f)$ the transfer function of the linear system when $s = j2\pi f$. Indeed for a linear and causal system we have:

$$\begin{array}{ll}\mathbf{F}(j2\pi f) & = \int_{-\infty}^{+\infty} \mathbf{C}\, e^{\mathbf{A}t}\, \mathbf{B}\, e^{-j2\pi f t} dt \\ & = \int_{0}^{+\infty} \mathbf{C}\, e^{\mathbf{A}t}\, \mathbf{B}\, e^{-j2\pi f t} dt \\ & = \left.\int_{0}^{+\infty} \mathbf{C}\, e^{\mathbf{A}t}\, \mathbf{B}\, e^{-st} dt\right|_{s=j2\pi f} \\ & = \left.\int_{0}^{+\infty} \mathbf{C}\, e^{-(s\mathbb{I}-\mathbf{A})t}\, \mathbf{B}\, dt\right|_{s=j2\pi f} \\ & = \left.\mathbf{C}\,(s\mathbb{I}-\mathbf{A})^{-1}\,\mathbf{B}\right|_{s=j2\pi f} \\ & = \left.\mathbf{F}(s)\right|_{s=j2\pi f} \end{array} \tag{6.63}$$

Returning to (6.60), we get from the preceding results:

$$\begin{array}{ll}\mathbf{S}_y(f) & = \int_0^{\infty} \left(\mathbf{C}\, e^{\mathbf{A}\xi}\, \mathbf{B}\right) \mathbf{P}_w\, e^{j2\pi f \xi}\, \mathbf{F}^T(j2\pi f)\, d\xi \\ & = \left(\int_0^{\infty} \left(\mathbf{C}\, e^{\mathbf{A}\xi}\, \mathbf{B}\right) e^{j2\pi f \xi}\, d\xi\right) \mathbf{P}_w\, \mathbf{F}^T(j2\pi f) \\ & = \mathbf{F}(-j2\pi f)\, \mathbf{P}_w\, \mathbf{F}^T(j2\pi f) \end{array} \tag{6.64}$$

We finally retrieve result (6.28):

$$\mathbf{S}_y(f) = \left.\mathbf{F}(-s)\, \mathbf{P}_w\, \mathbf{F}^T(s)\right|_{s=j2\pi f} \tag{6.65}$$

This completes the proof. ■

6.4 Kalman-Bucy filter

6.4.1 Linear Quadratic Estimator

Let's consider the following linear time invariant model where $\underline{w}(t)$ and $\underline{v}(t)$ are random processes which represents the process noise and the measurement noise respectively:

$$\left\{\begin{array}{l}\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) + \underline{w}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t) + \underline{v}(t)\end{array}\right. \tag{6.66}$$

The Kalman-Bucy filter is a state estimator that is optimal in the sense that it minimizes the covariance of the estimated error $\underline{e}(t) = \underline{x}(t) - \widehat{\underline{x}}(t)$ when the following conditions are met:

– Random vectors $\underline{w}(t)$ and $\underline{v}(t)$ are zero mean Gaussian noise. Let $p(\underline{w})$ and $p(\underline{v})$ be the probability density function (pdf) of random processes $\underline{w}(t)$ and $\underline{v}(t)$. Then:

$$\begin{cases} p(\underline{w}) = \frac{1}{(2\pi)^{n/2}\sqrt{\det(\mathbf{P}_w)}} e^{-\frac{1}{2}\underline{w}^T \mathbf{P}_w^{-1} \underline{w}} \\ p(\underline{v}) = \frac{1}{(2\pi)^{p/2}\sqrt{\det(\mathbf{P}_v)}} e^{-\frac{1}{2}v^T \mathbf{P}_v^{-1} v} \end{cases} \tag{6.67}$$

– Random vectors $\underline{w}(t)$ and $\underline{v}(t)$ are white noise (i.e. uncorrelated). The covariance matrices of $\underline{w}(t)$ and $\underline{v}(t)$ will be denoted $\mathbf{P}_w$ and $\mathbf{P}_v$ respectively:

$$\begin{cases} E\left[\underline{w}(t)\underline{w}^T(t+\tau)\right] = \mathbf{P}_w\delta(\tau) \text{ where } \mathbf{P}_w = \mathbf{P}_w^T > 0 \\ E\left[\underline{v}(t)v^T(t+\tau)\right] = \mathbf{P}_v\delta(\tau) \text{ where } \mathbf{P}_v = \mathbf{P}_v^T \geq 0 \end{cases} \tag{6.68}$$

– The cross correlation between $\underline{w}(t)$ and $\underline{v}(t)$ is zero:

$$\begin{cases} E\left[\underline{w}(t)\underline{v}^T(t+\tau)\right] = \mathbf{0} \\ E\left[\underline{v}(t)\underline{w}^T(t+\tau)\right] = \mathbf{0} \end{cases} \tag{6.69}$$

The Kalman-Bucy filter is a special form of the Luenberger observer (6.10):

$$\dot{\underline{\hat{x}}}(t) = \mathbf{A}\underline{\hat{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}(t)\left(\underline{y}(t) - \mathbf{C}\underline{\hat{x}}(t)\right) \tag{6.70}$$

Where the time dependent observer gain $\mathbf{L}(t)$, also-called Kalman gain, is given by:

$$\mathbf{L}(t) = \mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1} \tag{6.71}$$

where matrix $\mathbf{Y}(t)$ is the solution of the following *differential* Riccati equation:

$$\dot{\mathbf{Y}}(t) = \mathbf{A}\mathbf{Y}(t) + \mathbf{Y}(t)\mathbf{A}^T - \mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y}(t) + \mathbf{P}_w \tag{6.72}$$

The suboptimal observer gain $\mathbf{L} = \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1}$ is obtained thanks to the positive definite *steady state* solution $\mathbf{Y} = \mathbf{Y}^T > 0$ of the following *algebraic* Riccati equation:

$$\begin{cases} \mathbf{L} = \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1} \\ \mathbf{0} = \mathbf{A}\mathbf{Y} + \mathbf{Y}\mathbf{A}^T - \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y} + \mathbf{P}_w \end{cases} \tag{6.73}$$

For *discrete time systems*, the following discrete time algebraic Riccati equation has be be solved to get the suboptimal observer gain, as shown in section 3.8:

$$\mathbf{Y} + \mathbf{A}\mathbf{Y}\mathbf{C}^T\left(\mathbf{P}_v + \mathbf{C}\mathbf{Y}\mathbf{C}^T\right)^{-1}\mathbf{C}\mathbf{Y}\mathbf{A}^T - \mathbf{A}\mathbf{Y}\mathbf{A}^T - \mathbf{P}_w = \mathbf{0} \tag{6.74}$$

Kalman gain shall be tuned when the covariance matrices $\mathbf{P}_w$ and $\mathbf{P}_v$ are not known:

– When measurements $\underline{y}(t)$ are very noisy the coefficients of covariance matrix $\mathbf{P}_v$ are high and Kalman gain will be quite small;

– On the other hand when we do not trust very much the linear time invariant model of the process the coefficients of covariance matrix $\mathbf{P}_w$ are high and Kalman gain will be quite high.

From a practical point of view matrices $\mathbf{P}_w$ and $\mathbf{P}_v$ are design parameters which are tuned to achieve the desired properties of the closed-loop.

Moreover, when the Riccati equation (6.72) related to the Kalman-Bucy filter is identified to the Riccati equation related the Linear-Quadratic-Regulator (LQR) we get:

$$\begin{array}{ll} \dot{\mathbf{Y}}(t) & = \mathbf{A}\mathbf{Y}(t) + \mathbf{Y}(t)\mathbf{A}^T - \mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y}(t) + \mathbf{P}_w \\ & := \mathbf{A}^T\mathbf{Y}(t) + \mathbf{Y}(t)\mathbf{A} - \mathbf{Y}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{Y}(t) + \mathbf{Q} \\ \Rightarrow & \left\{ \begin{array}{l} \mathbf{Q} := \mathbf{P}_w \geq 0 \\ \mathbf{R} := \mathbf{P}_v > 0 \\ \mathbf{B} \to \mathbf{C}^T \\ \mathbf{A} \to \mathbf{A}^T \end{array} \right. \end{array} \tag{6.75}$$

6.4.2 Sketch of the proof

To get this result let's consider the following estimation error $\underline{e}(t)$:

$$\underline{e}(t) = \underline{x}(t) - \widehat{\underline{x}}(t) \tag{6.76}$$

Thus using (6.66) and (6.70) its time derivative reads:

$$\begin{array}{ll} \dot{\underline{e}}(t) & = \dot{\underline{x}}(t) - \dot{\widehat{\underline{x}}}(t) \\ & = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) + \underline{w}(t) - \left(\mathbf{A}\widehat{\underline{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}(t)\left(\underline{y}(t) - \mathbf{C}\widehat{\underline{x}}(t)\right)\right) \\ & = \mathbf{A}\underline{e}(t) + \underline{w}(t) - \mathbf{L}(t)\left(\mathbf{C}\underline{x}(t) + \underline{v}(t) - \mathbf{C}\widehat{\underline{x}}(t)\right) \\ & = \left(\mathbf{A} - \mathbf{L}(t)\mathbf{C}\right)\underline{e}(t) + \underline{w}(t) - \mathbf{L}(t)\underline{v}(t) \end{array} \tag{6.77}$$

Since $\underline{v}(t)$ and $\underline{w}(t)$ are zero mean white noise their weighted sum $\underline{n}(t) = \underline{w}(t) - \mathbf{L}(t)\underline{v}(t)$ is also a zero mean white noise. We get:

$$\underline{n}(t) = \underline{w}(t) - \mathbf{L}(t)\underline{v}(t) \Rightarrow \dot{\underline{e}}(t) = \left(\mathbf{A} - \mathbf{L}(t)\mathbf{C}\right)\underline{e}(t) + \underline{n}(t) \tag{6.78}$$

The covariance matrix $\mathbf{P}_n$ of $\underline{n}(t)$ reads:

$$\begin{array}{ll} \mathbf{P}_n & = E\left[\underline{n}(t)n^T(t)\right] \\ & = E\left[\left(\underline{w}(t) - \mathbf{L}(t)\underline{v}(t)\right)\left(\underline{w}(t) - \mathbf{L}(t)\underline{v}(t)\right)^T\right] \\ & = \mathbf{P}_w + \mathbf{L}(t)\mathbf{P}_v\mathbf{L}^T(t) \end{array} \tag{6.79}$$

Then the covariance matrix $\mathbf{Y}(t)$ of $\underline{e}(t)$ is obtained thanks to (6.22):

$$\begin{array}{ll} \dot{\mathbf{Y}}(t) & = \left(\mathbf{A} - \mathbf{L}(t)\mathbf{C}\right)\mathbf{Y}(t) + \mathbf{Y}(t)\left(\mathbf{A} - \mathbf{L}(t)\mathbf{C}\right)^T + \mathbf{P}_n \\ & = \mathbf{A}\mathbf{Y}(t) + \mathbf{Y}(t)\mathbf{A}^T - \mathbf{L}(t)\mathbf{C}\mathbf{Y}(t) - \mathbf{Y}(t)\mathbf{C}^T\mathbf{L}(t)^T + \mathbf{P}_n \end{array} \tag{6.80}$$

By using the expression (6.79) of the covariance matrix $\mathbf{P}_n$ of $\underline{n}(t)$ we get:

$$\begin{array}{l} \dot{\mathbf{Y}}(t) = \mathbf{A}\mathbf{Y}(t) + \mathbf{Y}(t)\mathbf{A}^T + \mathbf{P}_w \\ \qquad - \mathbf{L}(t)\mathbf{C}\mathbf{Y}(t) - \mathbf{Y}(t)\mathbf{C}^T\mathbf{L}(t)^T + \mathbf{L}(t)\mathbf{P}_v\mathbf{L}^T(t) \end{array} \tag{6.81}$$

Let's complete the square of $-\mathbf{L}(t)\mathbf{C}\mathbf{Y}(t) - \mathbf{Y}(t)\mathbf{C}^T\mathbf{L}(t)^T + \mathbf{L}(t)\mathbf{P}_v\mathbf{L}^T(t)$. First we will focus on the scalar case where we try to minimize the following quadratic function $f(\mathbf{L})$ where $\mathbf{P}_v > 0$:

$$f(\mathbf{L}) = -2\mathbf{L}\mathbf{C}\mathbf{Y} + \mathbf{P}_v\mathbf{L}^2 \tag{6.82}$$

Completing the square of $f(\mathbf{L})$ means writing $f(\mathbf{L})$ as follows:

$$f(\mathbf{L}) = \mathbf{P}_v^{-1}(\mathbf{L}\mathbf{P}_v - \mathbf{Y}\mathbf{C})^2 - \mathbf{Y}^2\mathbf{C}^2\mathbf{P}_v^{-1} \tag{6.83}$$

Then it is clear that $f(\mathbf{L})$ is minimal when $\mathbf{L}\mathbf{P}_v - \mathbf{Y}\mathbf{C}$ and that the minimal value of $f(\mathbf{L})$ is $-\mathbf{Y}^2\mathbf{C}^2\mathbf{P}_v^{-1}$. This approach can be extended to the matrix case. When we complete the square of $-\mathbf{L}(t)\mathbf{C}\mathbf{Y}(t) - \mathbf{Y}(t)\mathbf{C}^T\mathbf{L}(t)^T + \mathbf{L}(t)\mathbf{P}_v\mathbf{L}^T(t)$ we get:

$$\begin{aligned}
&-\mathbf{L}(t)\mathbf{C}\mathbf{Y}(t) - \mathbf{Y}(t)\mathbf{C}^T\mathbf{L}(t)^T + \mathbf{L}(t)\mathbf{P}_v\mathbf{L}^T(t) = \\
&\qquad \left(\mathbf{L}(t)\mathbf{P}_v - \mathbf{Y}(t)\mathbf{C}^T\right)\mathbf{P}_v^{-1}\left(\mathbf{L}(t)\mathbf{P}_v - \mathbf{Y}(t)\mathbf{C}^T\right)^T \\
&\qquad\qquad - \mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y}(t)
\end{aligned} \tag{6.84}$$

Using the preceding relation within (6.81) reads:

$$\begin{aligned}
\dot{\mathbf{Y}}(t) = {} & \mathbf{A}\mathbf{Y}(t) + \mathbf{Y}(t)\mathbf{A}^T + \mathbf{P}_w \\
& + \left(\mathbf{L}(t)\mathbf{P}_v - \mathbf{Y}(t)\mathbf{C}^T\right)\mathbf{P}_v^{-1}\left(\mathbf{L}(t)\mathbf{P}_v - \mathbf{Y}(t)\mathbf{C}^T\right)^T \\
& - \mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y}(t)
\end{aligned} \tag{6.85}$$

In order to find the optimum observer gain $\mathbf{L}(t)$ which minimizes the covariance matrix $\mathbf{Y}(t)$ we choose $\mathbf{L}(t)$ such that $\mathbf{Y}(t)$ decreases by the maximum amount possible at each instant in time. This is accomplished by setting $\mathbf{L}(t)$ as follows:

$$\mathbf{L}(t)\mathbf{P}_v - \mathbf{Y}(t)\mathbf{C}^T = \mathbf{0} \Leftrightarrow \mathbf{L}(t) = \mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1} \tag{6.86}$$

Once $\mathbf{L}(t)$ is set such that $\mathbf{L}(t)\mathbf{P}_v - \mathbf{Y}(t)\mathbf{C}^T = \mathbf{0}$ the matrix differential equation (6.85) reads as follows:

$$\dot{\mathbf{Y}}(t) = \mathbf{A}\mathbf{Y}(t) + \mathbf{Y}(t)\mathbf{A}^T - \mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y}(t) + \mathbf{P}_w \tag{6.87}$$

This is Equation (6.72).

6.5 Duality principle

In the chapter dedicated to the closed-loop solution of the infinite horizon Linear Quadratic Regulator (LQR) problem we have seen that the minimization of the cost functional $J(\underline{u}(t))$:

$$J(\underline{u}(t)) = \frac{1}{2}\int_0^\infty \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)dt \tag{6.88}$$

Under the constraint

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{x}(0) = \underline{x}_0 \end{cases} \tag{6.89}$$

This leads to solving the following algebraic Riccati equation where $\mathbf{Q} = \mathbf{Q}^T \geq 0$ (thus $\mathbf{Q}$ is symmetric and positive semi-definite matrix), and $\mathbf{R} = \mathbf{R}^T > 0$ is a symmetric and positive definite matrix:

$$\mathbf{0} = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} \tag{6.90}$$

The constant suboptimal Kalman gain $\mathbf{K}$ and the suboptimal stabilizing control $\underline{u}(t)$ are then defined as follows :

$$\begin{cases} \underline{u}(t) = -\mathbf{K}\underline{x}(t) \\ \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \end{cases} \tag{6.91}$$

Then let's compare the preceding relations with the following relations which are actually those which have been seen in (6.73):

$$\begin{cases} \mathbf{L} = \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1} \\ \mathbf{0} = \mathbf{Y}\mathbf{A}^T + \mathbf{A}\mathbf{Y} - \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y} + \mathbf{P}_w \end{cases} \tag{6.92}$$

Then it is clear than the duality principle on Table 6.1 between observer and controller gains apply.

Controller	Observer
$\mathbf{A}$	$\mathbf{A}^T$
$\mathbf{B}$	$\mathbf{C}^T$
$\mathbf{C}$	$\mathbf{B}^T$
$\mathbf{K}$	$\mathbf{L}^T$
$\mathbf{P} = \mathbf{P}^T \geq 0$	$\mathbf{Y} = \mathbf{Y}^T \geq 0$
$\mathbf{Q} = \mathbf{Q}^T \geq 0$	$\mathbf{P}_w = \mathbf{P}_w^T \geq 0$
$\mathbf{R} = \mathbf{R}^T > 0$	$\mathbf{P}_v = \mathbf{P}_v^T > 0$
$\mathbf{A} - \mathbf{B}\mathbf{K}$	$\mathbf{A}^T - \mathbf{C}^T\mathbf{L}^T$

Table 6.1: Duality principle

6.6 Separation principle

Let $\underline{e}(t)$ be the state estimation error:

$$\underline{e}(t) = \underline{x}(t) - \widehat{\underline{x}}(t) \tag{6.93}$$

Using (6.109) we get the following expressions for the dynamics of the state vector $\underline{x}(t)$:

$$\begin{array}{rl} \dot{\underline{x}}(t) & = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) + \underline{w}(t) \\ & = \mathbf{A}\underline{x}(t) - \mathbf{B}\mathbf{K}\widehat{\underline{x}}(t) + \underline{w}(t) \\ & = \mathbf{A}\underline{x}(t) - \mathbf{B}\mathbf{K}\left(\underline{x}(t) - \underline{e}(t)\right) + \underline{w}(t) \\ & = \left(\mathbf{A} - \mathbf{B}\mathbf{K}\right)\underline{x}(t) + \mathbf{B}\mathbf{K}\underline{e}(t) + \underline{w}(t) \end{array} \tag{6.94}$$

In addition using (6.108) and $\underline{y}(t) = \mathbf{C}\underline{x}(t) + \underline{v}(t)$ we get the following expressions for the dynamics of the estimation error $\underline{e}(t)$:

$$
\begin{aligned}
\dot{\underline{e}}(t) &= \dot{\underline{x}}(t) - \dot{\widehat{\underline{x}}}(t) \\
&= \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) + \underline{w}(t) - \left(\mathbf{A}\widehat{\underline{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}\left(\underline{y}(t) - \mathbf{C}\widehat{\underline{x}}(t)\right)\right) \\
&= (\mathbf{A} - \mathbf{LC})\,\underline{e}(t) + \underline{w}(t) - \mathbf{L}\underline{v}(t)
\end{aligned}
\tag{6.95}
$$

Thus the closed-loop dynamics is defined as follows:

$$
\begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\underline{e}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{BK} & \mathbf{BK} \\ \mathbf{0} & \mathbf{A} - \mathbf{LC} \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{e}(t) \end{bmatrix} + \begin{bmatrix} \underline{w}(t) \\ \underline{w}(t) - \mathbf{L}\underline{v}(t) \end{bmatrix} \tag{6.96}
$$

From equations (6.96) it is clear that the $2n$ eigenvalues of the closed-loop are just the union between the n eigenvalues of the state-feedback coming from the spectrum of $\mathbf{A} - \mathbf{BK}$ and the n eigenvalues of the state estimator coming from the spectrum of $\mathbf{A} - \mathbf{LC}$. This result is called the separation principle. More precisely the separation principle states that the optimal control law is achieved by adopting the following two steps procedure:

- First assume an exact measurement of the full state to solve the deterministic Linear Quadratic (LQ) control problem which minimizes the following cost functional $J(\underline{u}(t))$:

  $$
  J(\underline{u}(t)) = \frac{1}{2}\int_0^{\infty} \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)dt \tag{6.97}
  $$

  This leads to the following stabilizing control $\underline{u}(t)$:

  $$
  \underline{u}(t) = -\mathbf{K}\underline{x}(t) \tag{6.98}
  $$

  Where the Kalman gain $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$ is obtained thanks to the positive semi-definite solution $\mathbf{P}$ of the following algebraic Riccati equation:

  $$
  \mathbf{0} = \mathbf{A}^T\mathbf{P} + \mathbf{PA} - \mathbf{PBR}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} \tag{6.99}
  $$

- Then obtain an optimal estimate of the state which minimizes the following estimated error covariance:

  $$
  E\left[\underline{e}^T(t)\underline{e}(t)\right] = E\left[\left(\underline{x}(t) - \widehat{\underline{x}}(t)\right)^T\left(\underline{x}(t) - \widehat{\underline{x}}(t)\right)\right] \tag{6.100}
  $$

  This leads to the Kalman-Bucy filter:

  $$
  \frac{d}{dt}\widehat{\underline{x}}(t) = \mathbf{A}\widehat{\underline{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}\left(\underline{y}(t) - \mathbf{C}\widehat{\underline{x}}(t)\right) \tag{6.101}
  $$

  And the stabilizing control $\underline{u}(t)$ now reads:

  $$
  \underline{u}(t) = -\mathbf{K}\widehat{\underline{x}}(t) \tag{6.102}
  $$

The observer gain $\mathbf{L}$ reads:

$$\mathbf{L} = \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1} \tag{6.103}$$

where matrix $\mathbf{Y} = \mathbf{Y}^T > 0$ is the positive definite solution of the following algebraic Riccati equation

$$\mathbf{0} = \mathbf{A}\mathbf{Y} + \mathbf{Y}\mathbf{A}^T - \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y} + \mathbf{P}_w \tag{6.104}$$

It is worth noticing that the optimal state estimate is independent of $\mathbf{Q}$ and $\mathbf{R}$. Moreover the observer dynamics much be faster than the desired state-feedback dynamics.

Furthermore the dynamics of the state vector $\underline{x}(t)$ is slightly modified when compared with an actual state-feedback control $\underline{u}(t) = -\mathbf{K}\underline{x}(t)$. Indeed we have seen in (6.94) that the dynamics of the state vector $\underline{x}(t)$ is now modified and depends on $\underline{e}(t)$ and $\underline{w}(t)$:

$$\dot{\underline{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K})\,\underline{x}(t) + \mathbf{B}\mathbf{K}\underline{e}(t) + \underline{w}(t) \tag{6.105}$$

6.7 Controller transfer function

First let's assume that a full state-feedback $\underline{u}(t) = -\mathbf{K}\underline{x}(t)$ is applied on the following system:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) + \underline{w}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t) + \underline{v}(t) \end{cases} \tag{6.106}$$

Then the dynamics of the closed-loop system is given by:

$$\underline{u}(t) = -\mathbf{K}\underline{x}(t) \Rightarrow \dot{\underline{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K})\,\underline{x}(t) + \underline{w}(t) \tag{6.107}$$

If the full state vector $\underline{x}(t)$ is assumed not to be available the control $\underline{u}(t) = -\mathbf{K}\underline{x}(t)$ cannot be computed. Then an observer has to be added. We recall the dynamics of the observer (see (6.10)):

$$\dot{\underline{\hat{x}}}(t) = \mathbf{A}\underline{\hat{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}\left(\underline{y}(t) - \mathbf{C}\underline{\hat{x}}(t)\right) \tag{6.108}$$

and the control $\underline{u}(t) = -\mathbf{K}\underline{x}(t)$ has to be changed into:

$$\underline{u}(t) = -\mathbf{K}\underline{\hat{x}}(t) \tag{6.109}$$

Gathering (6.108) and (6.109) leads to the state space representation of the controller:

$$\begin{bmatrix} \dot{\underline{\hat{x}}}(t) \\ \underline{u}(t) \end{bmatrix} \left[\begin{array}{c|c} \mathbf{A}_K & \mathbf{B}_K \\ \hline \mathbf{C}_K & \mathbf{D}_K \end{array}\right] \begin{bmatrix} \underline{\hat{x}}(t) \\ \underline{y}(t) \end{bmatrix} \tag{6.110}$$

Where:

$$\left[\begin{array}{c|c} \mathbf{A}_K & \mathbf{B}_K \\ \hline \mathbf{C}_K & \mathbf{D}_K \end{array}\right] = \left[\begin{array}{c|c} \mathbf{A} - \mathbf{B}\mathbf{K} - \mathbf{L}\mathbf{C} & \mathbf{L} \\ \hline -\mathbf{K} & \mathbf{0} \end{array}\right] \tag{6.111}$$

The controller transfer function $\mathbf{K}(s)$ is the relation between the Laplace transform of its output, $\underline{U}(s)$, and the Laplace transform of its input, $\underline{Y}(s)$. By

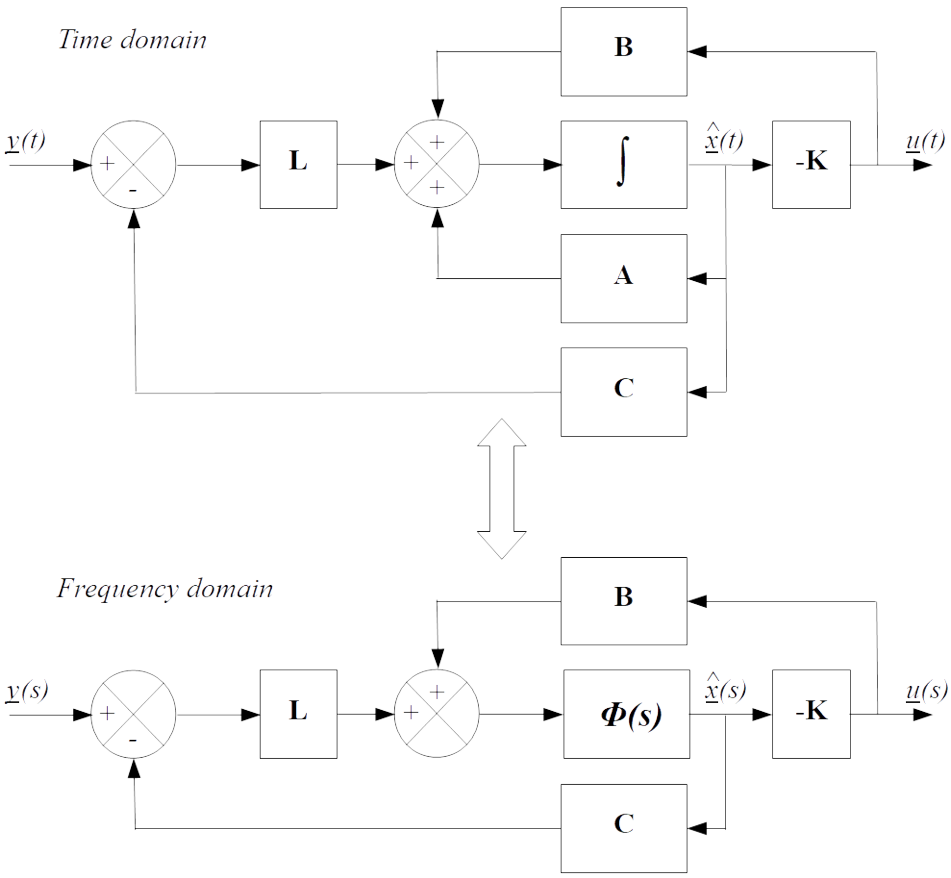


Figure 6.3: Block diagram of the controller in the time domain and the frequency domain

taking the Laplace transform of equation (6.108) and (6.109) (and assuming no initial condition) we get:

$$\begin{cases} s\underline{\widehat{X}}(s) = \mathbf{A}\underline{\widehat{X}}(s) + \mathbf{B}\underline{U}(s) + \mathbf{L}\left(\underline{Y}(s) - \mathbf{C}\underline{\widehat{X}}(s)\right) \\ \underline{U}(s) = -\mathbf{K}\underline{\widehat{X}}(s) \end{cases} \tag{6.112}$$

We finally get:

$$\underline{U}(s) = -\mathbf{K}(s)\underline{Y}(s) \tag{6.113}$$

where the controller transfer function $\mathbf{K}(s)$ reads:

$$\mathbf{K}(s) = \mathbf{K}\left(s\mathbb{I} - \mathbf{A} + \mathbf{BK} + \mathbf{LC}\right)^{-1}\mathbf{L} \tag{6.114}$$

The preceding relation can be equivalently represented in the time domain or the frequency domain by the block diagram shown in Figure 6.3 where:

$$\mathbf{\Phi}(s) := (s\mathbb{I} - \mathbf{A})^{-1} \tag{6.115}$$

6.8 Loop Transfer Recovery

6.8.1 Lack of guaranteed robustness of LQG design

According to the choice of matrix $\mathbf{L}$ which drives the dynamics of the error $\underline{e}(t)$ the closed-loop may not be stable. So far LQR is shown to have either infinite gain margin (stable open-loop plant) or at least -6 dB gain margin and at least sixty degrees phase margin. In 1978 John Doyle[2] showed that all the nice robustness properties of LQR design can be lost once the observer is added and that LQG design can exhibit arbitrarily poor stability margins. Around 1981 Doyle along with Gunter Stein followed this line by showing that the loop shape itself will, in general, change when a filter is added for estimation. Fortunately there is a way of designing the Kalman-Bucy filter so that the full state-feedback properties are recovered at the input of the plant. This is the purpose of the Loop Transfer Recovery design. The LQG/LTR design method was introduced by Doyle and Stein in 1981 before the development of H_2 and H_∞ methods which is a more general approach to directly handle many types of modeling uncertainties.

6.8.2 Doyle's seminal example

Consider the following state space realization:

$$\begin{cases} \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \underline{u}(t) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \underline{w}(t) \\ y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \underline{v}(t) \end{cases} \tag{6.116}$$

where $\underline{w}(t)$ and $\underline{v}(t)$ are Gaussian white noise with covariance matrices $\mathbf{P}_w$ and $\mathbf{P}_v$, respectively:

$$\begin{cases} \mathbf{P}_w = \sigma \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \\ \sigma > 0 \\ \mathbf{P}_v = 1 \end{cases} \tag{6.117}$$

Let $J(\underline{u}(t))$ be the following cost functional to be minimized :

$$J(\underline{u}(t)) = \frac{1}{2} \int_0^\infty \underline{x}^T(t) \mathbf{Q} \underline{x}(t) + \underline{u}^T(t) \mathbf{R} \underline{u}(t) dt \tag{6.118}$$

where:

$$\begin{cases} \mathbf{Q} = q \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \\ q > 0 \\ \mathbf{R} = 1 \end{cases} \tag{6.119}$$

Applying the separation principle the optimal control law is achieved by adopting the following two steps procedure:

[2]Doyle J.C., Guaranteed margins for LQG regulators, IEEE Transactions on Automatic Control, Volume: 23, Issue: 4, Aug 1978

- First assume an exact measurement of the full state to solve the deterministic Linear Quadratic (LQ) control problem which minimizes the following cost functional $J(\underline{u}(t))$:

$$J(\underline{u}(t)) = \frac{1}{2} \int_0^\infty \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)dt \tag{6.120}$$

This leads to the following stabilizing control $\underline{u}(t)$:

$$\underline{u}(t) = -\mathbf{K}\underline{x}(t) \tag{6.121}$$

Where the Kalman gain $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$ is obtained thanks to the positive semi-definite solution $\mathbf{P}$ of the following algebraic Riccati equation:

$$\mathbf{0} = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} \tag{6.122}$$

We get:

$$\mathbf{P} = \begin{bmatrix} * & * \\ \alpha & \alpha \end{bmatrix} \tag{6.123}$$

And:

$$\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} = \alpha \begin{bmatrix} 1 & 1 \end{bmatrix} \text{ where } \alpha = 2 + \sqrt{4+q} > 0 \tag{6.124}$$

- Then obtain an optimal estimate of the state which minimizes the following estimated error covariance :

$$E\left[\underline{e}^T(t)\underline{e}(t)\right] = E\left[\left(\underline{x}(t) - \widehat{\underline{x}}(t)\right)^T\left(\underline{x}(t) - \widehat{\underline{x}}(t)\right)\right] \tag{6.125}$$

This leads to the Kalman-Bucy filter:

$$\frac{d}{dt}\widehat{\underline{x}}(t) = \mathbf{A}\widehat{\underline{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}(t)\left(\underline{y}(t) - \mathbf{C}\widehat{\underline{x}}(t)\right) \tag{6.126}$$

And the stabilizing control $\underline{u}(t)$ now reads:

$$\underline{u}(t) = -\mathbf{K}\widehat{\underline{x}}(t) \tag{6.127}$$

The observer gain $\mathbf{L} = \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1}$ is obtained thanks to the positive semi-definite solution $\mathbf{Y}$ of the following algebraic Riccati equation:

$$\mathbf{0} = \mathbf{A}\mathbf{Y} + \mathbf{Y}\mathbf{A}^T - \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y} + \mathbf{P}_w \tag{6.128}$$

We get:

$$\mathbf{Y} = \begin{bmatrix} \beta & * \\ \beta & * \end{bmatrix} \tag{6.129}$$

And:

$$\mathbf{L} = \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1} = \beta \begin{bmatrix} 1 \\ 1 \end{bmatrix} \text{ where } \beta = 2 + \sqrt{4+\sigma} > 0 \tag{6.130}$$

Now assume that the input matrix of the plant is multiplied by a scalar gain Δ (nominally unit) :

$$\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \Delta\mathbf{B}\underline{u}(t) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \underline{w}(t) \tag{6.131}$$

In order to assess the stability of the closed-loop system we will assume no exogenous disturbance $\underline{v}(t)$ and $\underline{w}(t)$. Then the dynamics of the closed-loop system reads:

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \dot{\widehat{\underline{x}}}(t) \end{bmatrix} = \mathbf{A}_{cl} \begin{bmatrix} \underline{x}(t) \\ \widehat{\underline{x}}(t) \end{bmatrix} \tag{6.132}$$

where, using (6.108) and (6.131):

$$\mathbf{A}_{cl} = \begin{bmatrix} \mathbf{A} & -\Delta\mathbf{BK} \\ \mathbf{LC} & \mathbf{A} - \mathbf{BK} - \mathbf{LC} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & -\Delta\alpha & -\Delta\alpha \\ \beta & 0 & 1-\beta & 1 \\ \beta & 0 & -\alpha-\beta & 1-\alpha \end{bmatrix} \tag{6.133}$$

The characteristic equation of the closed-loop system is:

$$\det\left(s\mathbb{I} - \mathbf{A}_{cl}\right) = s^4 + p_3 s^3 + p_2 s^2 + p_1 s + p_0 = 0 \tag{6.134}$$

The evaluation of coefficients p_3, p_2, p_1 and p_0 is quite tedious. Nevertheless coefficient p_0 reads;

$$p_0 = 1 + (1-\Delta)\alpha\beta \tag{6.135}$$

The closed-loop system is unstable if:

$$p_0 < 0 \Leftrightarrow \Delta > 1 + \frac{1}{\alpha\beta} \tag{6.136}$$

With large values of α and β even a slight increase in the value of Δ from its nominal value will render the closed-loop system to be unstable. Thus the phase margin of the LQG control-loop can be almost 0. This example clearly shows that the robustness of the LQG control-loop to modeling uncertainty is not guaranteed.

6.8.3 Closed-loop eigenvalues and eigenvectors

Let λ be a closed-loop eigenvalue and $\underline{v}$ the corresponding eigenvector of the LQ state-feedback:

$$\begin{array}{c} \left(\lambda\mathbb{I} - (\mathbf{A} - \mathbf{BK})\right)\underline{v} = \underline{0} \text{ where } \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \\ \Leftrightarrow \lambda\underline{v} - \mathbf{A}\underline{v} + \mathbf{BR}^{-1}\mathbf{B}^T\mathbf{P}\underline{v} = \underline{0} \end{array} \tag{6.137}$$

Thus we can equivalently write:

$$\begin{bmatrix} \lambda\mathbb{I} - \mathbf{A} & \mathbf{BR}^{-1}\mathbf{B}^T \end{bmatrix} \begin{bmatrix} \underline{v} \\ \mathbf{P}\underline{v} \end{bmatrix} = \underline{0} \tag{6.138}$$

Furthermore we have seen that matrix $\mathbf{P}$ is the positive semi-definite solution of the following algebraic Riccati equation:

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} = \mathbf{0} \tag{6.139}$$

Thus, multiplying this equality by eigenvector $-\underline{v}$, and adding and subtracting $\lambda\mathbf{P}\underline{v}$, we get:

$$\begin{array}{c} \left(-\mathbf{A}^T\mathbf{P} - \mathbf{P}\mathbf{A} + \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} - \mathbf{Q}\right)\underline{v} + \lambda\mathbf{P}\underline{v} - \lambda\mathbf{P}\underline{v} = \mathbf{0} \\ \Leftrightarrow \mathbf{P}\left(\lambda\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}\right)\underline{v} - \left(\mathbf{Q} + \lambda\mathbf{P} + \mathbf{A}^T\mathbf{P}\right)\underline{v} = \mathbf{0} \end{array} \tag{6.140}$$

Inserting (6.137) within (6.140) yields:

$$\begin{array}{c} \left(\lambda\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}\right)\underline{v} = \underline{0} \Rightarrow \left(\mathbf{Q} + \lambda\mathbf{P} + \mathbf{A}^T\mathbf{P}\right)\underline{v} = \mathbf{0} \\ \Leftrightarrow \begin{bmatrix} \mathbf{Q} & \lambda\mathbb{I} + \mathbf{A}^T \end{bmatrix} \begin{bmatrix} \underline{v} \\ \mathbf{P}\underline{v} \end{bmatrix} = \underline{0} \end{array} \tag{6.141}$$

Finally, (6.138) and (6.141) together read:

$$\begin{array}{c} \begin{bmatrix} \lambda\mathbb{I} - \mathbf{A} & \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ \mathbf{Q} & \lambda\mathbb{I} + \mathbf{A}^T \end{bmatrix} \begin{bmatrix} \underline{v} \\ \mathbf{P}\underline{v} \end{bmatrix} = \underline{0} \\ \Leftrightarrow \left(\lambda\mathbb{I} - \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix}\right) \begin{bmatrix} \underline{v} \\ \mathbf{P}\underline{v} \end{bmatrix} = \underline{0} \end{array} \tag{6.142}$$

The preceding relation indicates the equivalence between any closed-loop eigenvalue λ and the corresponding eigenvector $\underline{v}$ of the LQ state-feedback correspond and any eigenvalue of the Hamiltonian matrix $\mathbf{H} := \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix}$ and the corresponding eigenvector $\begin{bmatrix} \underline{v} \\ \mathbf{P}\underline{v} \end{bmatrix}$ where matrix $\mathbf{P}$ is the positive semi-definite solution of the algebraic Riccati equation.

6.8.4 Asymptotic behavior of Riccati equation

Now, let $\mathbf{W}$ be some unitary matrix ($\mathbf{W}^T\mathbf{W} = \mathbb{I}$) and $\mathbf{M}$ be some symmetric positive definite matrix ($\mathbf{M} = \mathbf{M}^T > 0$) such that $\mathbf{R} = \epsilon^2\mathbf{M}$. Then if the transfer function $\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}$ is right invertible with no unstable zeros the following relation holds:

$$\left\{ \begin{array}{l} \mathbf{R} = \epsilon^2\mathbf{M} \\ \mathbf{Q} = \mathbf{C}^T\mathbf{C} \\ \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \end{array} \right. \Rightarrow \lim_{\epsilon\to 0} \mathbf{K} = \frac{1}{\epsilon}\mathbf{M}^{-0.5}\mathbf{W}\mathbf{C} = \mathbf{R}^{-0.5}\mathbf{W}\mathbf{C} \tag{6.143}$$

Indeed, the algebraic Riccati equation becomes in that case:

$$\begin{array}{l} \left\{ \begin{array}{l} \mathbf{R} = \epsilon^2\mathbf{M} = \left(\epsilon\mathbf{M}^{0.5}\right)^T\left(\epsilon\mathbf{M}^{0.5}\right) \\ \mathbf{Q} = \mathbf{C}^T\mathbf{C} = \mathbf{C}^T\mathbf{W}^T\mathbf{W}\mathbf{C} = (\mathbf{W}\mathbf{C})^T(\mathbf{W}\mathbf{C}) \end{array} \right. \\ \Rightarrow \mathbf{0} \;\; = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} \\ \quad\;\; = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\frac{\mathbf{M}^{-1}}{\epsilon^2}\mathbf{B}^T\mathbf{P} + (\mathbf{W}\mathbf{C})^T(\mathbf{W}\mathbf{C}) \\ \quad\;\; = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \left(\frac{\mathbf{M}^{-0.5}}{\epsilon}\mathbf{B}^T\mathbf{P}\right)^T\left(\frac{\mathbf{M}^{-0.5}}{\epsilon}\mathbf{B}^T\mathbf{P}\right) + (\mathbf{W}\mathbf{C})^T(\mathbf{W}\mathbf{C}) \end{array} \tag{6.144}$$

If the transfer function $\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}$ is right invertible with no unstable zeros then $\mathbf{P} \to \mathbf{0}$ as $\epsilon \to 0$ and the preceding equation reads as follows, where $\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A}$ has been neglected:

$$\begin{array}{c} \mathbf{0} \underset{\epsilon\to 0}{\approx} -\left(\frac{\mathbf{M}^{-0.5}}{\epsilon}\mathbf{B}^T\mathbf{P}\right)^T\left(\frac{\mathbf{M}^{-0.5}}{\epsilon}\mathbf{B}^T\mathbf{P}\right) + (\mathbf{W}\mathbf{C})^T(\mathbf{W}\mathbf{C}) \\ \Rightarrow \frac{\mathbf{M}^{-0.5}}{\epsilon}\mathbf{B}^T\mathbf{P} \underset{\epsilon\to 0}{\approx} \mathbf{W}\mathbf{C} \end{array} \tag{6.145}$$

We finally get, using the fact that $\mathbf{R} = \epsilon^2\mathbf{M}$:

$$\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} = \frac{1}{\epsilon}\mathbf{M}^{-0.5}\frac{\mathbf{M}^{-0.5}}{\epsilon}\mathbf{B}^T\mathbf{P} \underset{\epsilon\to 0}{\approx} \frac{1}{\epsilon}\mathbf{M}^{-0.5}\mathbf{W}\mathbf{C} = \mathbf{R}^{-0.5}\mathbf{W}\mathbf{C} \tag{6.146}$$

This completes the proof. ■

6.8.5 Loop Transfer Recovery (LTR) design

Linear Quadratic (LQ) controller and the Kalman-Bucy filter (KF) alone have very good robustness property. Nevertheless we have seen with Doyle's seminal example that Linear Quadratic Gaussian (LQG) control which simultaneously involves a Linear Quadratic (LQ) controller and a Kalman-Bucy filter (KF) does not have any guaranteed robustness. Therefore the LQG / LTR design tries to recover a target open-loop transfer function. The target loop transfer function is either:

– the Linear Quadratic (LQ) control open-loop transfer function, which is $\mathbf{K}\mathbf{\Phi}(s)\mathbf{B}$

– or the Kalman-Bucy filter (KF) open-loop transfer function, which is $\mathbf{C}\mathbf{\Phi}(s)\mathbf{L}$.

Let ρ be a parameter design of either design matrix $\mathbf{Q}$ or matrix $\mathbf{P}_w$ and $\mathbf{F}(s)$ the transfer function of the plant:

$$\mathbf{F}(s) = \mathbf{C}\mathbf{\Phi}(s)\mathbf{B} \tag{6.147}$$

Then two types of Loop Transfer Recovery are possible:

– Input recovery: let ω_c be the cut-off frequency (i.e. 0 dB) of the targeted dynamics. The objective is to tune ρ such that:

$$\lim_{\rho\to\infty} \mathbf{K}(s)\mathbf{F}(s)\Big|_{\substack{s=j\omega \\ \omega<\omega_c}} \approx \mathbf{K}\mathbf{\Phi}(s)\mathbf{B}\Big|_{\substack{s=j\omega \\ \omega<\omega_c}} \tag{6.148}$$

The objective of the input recovery design is shown in Figure 6.4. The corresponding objective in the state space domain is the following:

$$\left\{ \begin{array}{l} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{u}(t) = -\mathbf{K}\underline{x}(t) + \underline{r}(t) \end{array} \right. \tag{6.149}$$

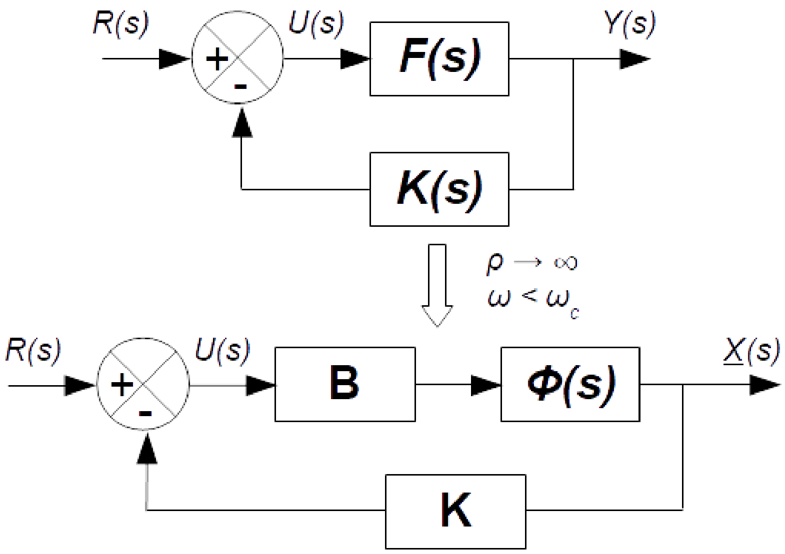


Figure 6.4: Input recovery objective

– Output recovery: let ω_c be the cut-off frequency (i.e. 0 dB) of the targeted dynamics. The objective is to tune ρ such that:

$$\lim_{\rho\to\infty} \mathbf{F}(s)\mathbf{K}(s)\Big|_{\substack{s=j\omega \\ \omega<\omega_c}} \approx \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\Big|_{\substack{s=j\omega \\ \omega<\omega_c}} \tag{6.150}$$

The objective of the output recovery design is shown in Figure 6.5[3]. The corresponding objective in the state space domain is the following:

$$\begin{cases} \dot{\widehat{\underline{x}}}(t) = \mathbf{A}\widehat{\underline{x}}(t) + \mathbf{L}\left(\underline{r}(t) - \widehat{\underline{y}}(t)\right) \\ \widehat{\underline{y}}(t) = \mathbf{C}\widehat{\underline{x}}(t) \end{cases} \tag{6.151}$$

We recall that initial design matrices $\mathbf{Q}_0$ and $\mathbf{R}_0$ are set to meet control requirements whereas initial design matrices $\mathbf{P}_{w0}$ and $\mathbf{P}_{v0}$ are set to meet observer requirements. Let ρ be a parameter design of either design matrix $\mathbf{P}_w$ or matrix $\mathbf{Q}$. Weighting parameter ρ is tuned to make a trade-off between initial performances and stability margins and is set according to the type of Loop Transfer Recovery:

– Input recovery: a new observer design with the following design matrices:

$$\begin{cases} \mathbf{P}_w = \mathbf{P}_{w0} + \rho^2\mathbf{B}\mathbf{B}^T \\ \mathbf{P}_v = \mathbf{P}_{v0} \end{cases} \tag{6.152}$$

[3]Ronaldo Waschburger and Karl Heinz Kienitz, A root locus approach to loop transfer recovery based controller design, 13th International Conference on Control Automation Robotics & Vision (ICARCV), 2014

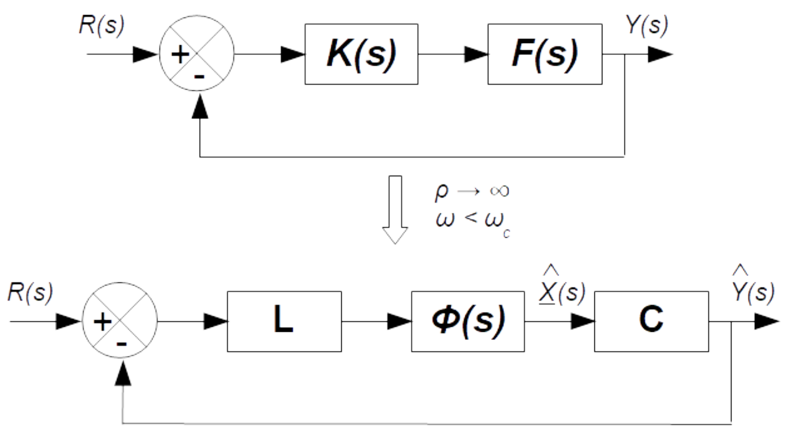


Figure 6.5: Output recovery objective

– Output recovery: a new controller is designed with the following design matrices:

$$\begin{cases} \mathbf{Q} = \mathbf{Q}_0 + \rho^2 \mathbf{C}^T \mathbf{C} \\ \mathbf{R} = \mathbf{R}_0 \end{cases} \tag{6.153}$$

The preceding relation is simply obtained by applying the duality principle.

From a practical point of view, design parameter ρ is increased until satisfactory robust properties of the loop transfer function are achieved. It is worth noticing that to apply Loop Transfer Recovery (LTR) the transfer function $\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}$ shall be minimum phase (i.e. no zero with positive real part) and square (meaning that the system has the same number of inputs and outputs).

Example 6.2. *Let's the double integrator plant:*

$$\begin{cases} \mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \\ \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \end{cases} \tag{6.154}$$

Let:

$$\begin{cases} \mathbf{K} = \begin{bmatrix} k_1 & k_2 \end{bmatrix} \\ \mathbf{L} = \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} \end{cases} \tag{6.155}$$

Then the controller transfer function is given by (6.114)*:*

$$\begin{array}{ll} \mathbf{K}(s) & = \mathbf{K}\left(s\mathbb{I} - \mathbf{A} + \mathbf{B}\mathbf{K} + \mathbf{L}\mathbf{C}\right)^{-1}\mathbf{L} \\ & = \begin{bmatrix} k_1 & k_2 \end{bmatrix} \begin{bmatrix} s + l_1 & -1 \\ k_1 + l_2 & s + k_2 \end{bmatrix}^{-1} \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} \\ & = \frac{(k_1 l_1 + k_2 l_2)s + k_1 l_2}{s^2 + (k_2 + l_1)s + k_2 l_1 + k_1 + l_2} \end{array} \tag{6.156}$$

From (6.153) *we set* $\mathbf{Q}$ *and* $\mathbf{R}$ *as follows:*

$$\left\{\begin{array}{l}\mathbf{Q}_0 := \mathbf{0} \\ \mathbf{Q} = \mathbf{Q}_0 + \rho^2\mathbf{C}^T\mathbf{C} \\ \mathbf{R} = \mathbf{R}_0 := 1\end{array}\right. \Rightarrow \mathbf{Q} = \rho^2\mathbf{C}^T\mathbf{C} = \begin{bmatrix} \rho^2 & 0 \\ 0 & 0 \end{bmatrix} \tag{6.157}$$

The Kalman gain $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$ *is then obtained thanks to the positive semi-definite solution* $\mathbf{P}$ *of the following algebraic Riccati equation:*

$$\begin{array}{l}\mathbf{0} = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} + \mathbf{Q} \Rightarrow \mathbf{P} = \begin{bmatrix} * & * \\ \rho & \sqrt{2\rho} \end{bmatrix} \\ \Rightarrow \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} = \begin{bmatrix} \rho & \sqrt{2\rho} \end{bmatrix} := \begin{bmatrix} k_1 & k_2 \end{bmatrix}\end{array} \tag{6.158}$$

Consequently:

$$\begin{array}{rl}\lim\limits_{\rho\to\infty} \mathbf{K}(s) & = \lim\limits_{\rho\to\infty} \frac{(k_1 l_1 + k_2 l_2)s + k_1 l_2}{s^2 + (k_2 + l_1)s + k_2 l_1 + k_1 + l_2} \\ & = \lim\limits_{\rho\to\infty} \frac{(\rho l_1 + \sqrt{2\rho} l_2)s + \rho l_2}{s^2 + (\sqrt{2\rho} + l_1)s + \sqrt{2\rho} l_1 + \rho + l_2} \\ & = \lim\limits_{\rho\to\infty} \frac{\rho l_1 s + \rho l_2}{\rho} \\ & = l_1 s + l_2\end{array} \tag{6.159}$$

The transfer function of the plant reads:

$$\mathbf{F}(s) = \mathbf{C}\mathbf{\Phi}(s)\mathbf{B} = \mathbf{C}\left(s\mathbb{I} - \mathbf{A}\right)^{-1}\mathbf{B} = \frac{1}{s^2} \tag{6.160}$$

Therefore:

$$\lim_{\rho\to\infty} \mathbf{K}(s)\mathbf{F}(s) = \frac{l_1 s + l_2}{s^2} \tag{6.161}$$

Note that:

$$\mathbf{C}\mathbf{\Phi}(s)\mathbf{L} = \frac{l_1 s + l_2}{s^2} \tag{6.162}$$

Therefore the loop transfer function has been recovered.

■

6.9 Proof of the Loop Transfer Recovery condition

6.9.1 Loop transfer function with observer

The Loop Transfer Recovery design procedure tries to recover a target loop transfer function, here the open-loop full state LQ control, despite the use of the observer.

The lecture of Faryar Jabbari, from the Henry Samueli School of Engineering, University of California, is the primary source of this section[4].

We will first show what happen when adding an observer-based closed-loop on the following system where $\underline{y}(t)$ is the *actual* output of the system (not the *controlled* output):

$$\left\{\begin{array}{l}\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t)\end{array}\right. \tag{6.163}$$

[4] http://mae2.eng.uci.edu/~fjabbari//me270b/chap9.pdf

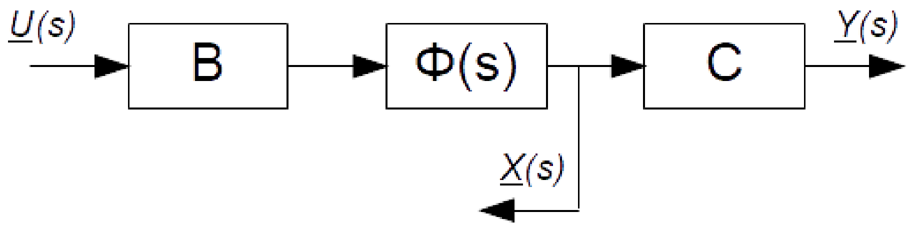


Figure 6.6: Block diagram of open-loop transfer function

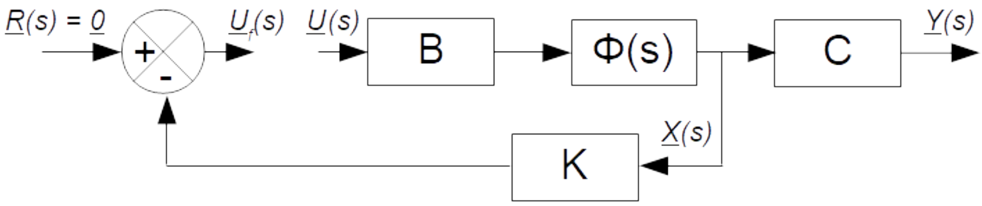


Figure 6.7: Broken state-feedback loop

Taking the Laplace transform and assuming no initial condition, we get:

$$\left\{ \begin{array}{l} s\underline{X}(s) = \mathbf{A}\underline{X}(s) + \mathbf{B}\underline{U}(s) \\ \underline{Y}(s) = \mathbf{C}\underline{X}(s) \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \underline{X}(s) = \mathbf{\Phi}(s)\mathbf{B}\underline{U}(s) \\ \underline{Y}(s) = \mathbf{C}\underline{X}(s) \end{array} \right. \tag{6.164}$$

where:

$$\mathbf{\Phi}(s) = (s\mathbb{I} - \mathbf{A})^{-1} \tag{6.165}$$

The preceding relations can be represented by the block diagram in Figure 6.6. Let $\mathbf{K}$ be a full state-feedback gain matrix such that the closed-loop system is asymptotically stable, i.e. the eigenvalues of $\mathbf{A} - \mathbf{BK}$ lie in the left half s-plane, and the open-loop transfer function when the loop is broken at the input point of the given system meets some given frequency dependent specifications. The state feedback control $\underline{u}_f$ with full state available is:

$$\underline{u}_f(t) = -\mathbf{K}\underline{x}(t) \Leftrightarrow \underline{U}_f(s) = -\mathbf{K}\underline{X}(s) \tag{6.166}$$

We will focus on the regulator problem and thus $\underline{r} = \underline{0}$. As shown in Figure 6.7 the loop transfer function is evaluated when the loop is broken at the input point of the system. The so-called target loop transfer function $\mathbf{L}_t(s)$ is defined as follows:

$$\underline{U}_f(s) = \mathbf{L}_t(s)\underline{U}(s) \text{ where } \mathbf{L}_t(s) = -\mathbf{K}\mathbf{\Phi}(s)\mathbf{B} \tag{6.167}$$

If the full state vector $\underline{x}(t)$ is assumed not to be available, the control $\underline{u}(t) = -\mathbf{K}\underline{x}(t)$ cannot be computed. We then add an observer with the following expression:

$$\dot{\widehat{\underline{x}}}(t) = \mathbf{A}\widehat{\underline{x}}(t) + \mathbf{B}\underline{u}(t) + \mathbf{L}\left(\underline{y}(t) - \mathbf{C}\widehat{\underline{x}}(t)\right) \tag{6.168}$$

The observer-based state-feedback control $\underline{u}_o$ is:

$$\underline{u}_o(t) = -\mathbf{K}\widehat{\underline{x}}(t) \tag{6.169}$$

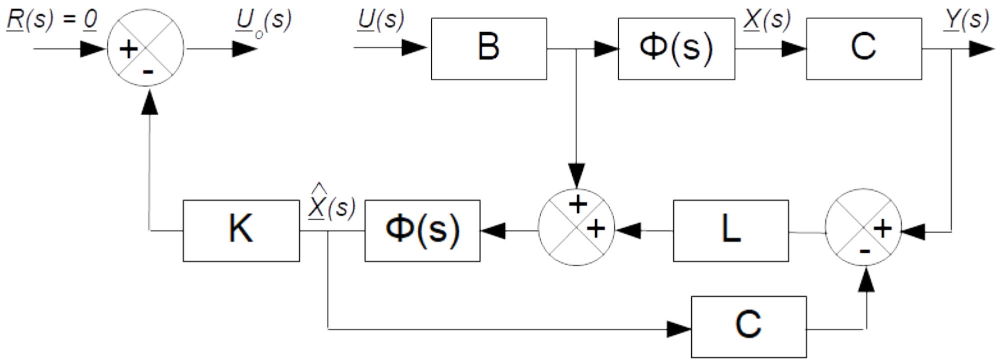


Figure 6.8: Broken observer-based feedback loop

Taking the Laplace transform results in:

$$\begin{cases} s\underline{\widehat{X}}(s) = \mathbf{A}\underline{\widehat{X}}(s) + \mathbf{B}\underline{U}(s) + \mathbf{L}\left(\underline{Y}(s) - \mathbf{C}\underline{\widehat{X}}(s)\right) \\ \underline{U}_o(s) = -\mathbf{K}\underline{\widehat{X}}(s) \end{cases} \tag{6.170}$$

or, equivalently:

$$\begin{cases} \underline{\widehat{X}}(s) = (s\mathbb{I} - \mathbf{A})^{-1}\left(\mathbf{B}\underline{U}(s) + \mathbf{L}\left(\underline{Y}(s) - \mathbf{C}\underline{\widehat{X}}(s)\right)\right) \\ \underline{U}_o(s) = -\mathbf{K}\underline{\widehat{X}}(s) \end{cases} \tag{6.171}$$

Usually equation (6.171) is not the same than (6.166). Relation (6.171) can be represented by the block diagram in Figure 6.8. The loop transfer function evaluated when the loop is broken at the input point of the closed-loop system becomes:

$$\begin{array}{l} \begin{cases} \underline{\widehat{X}}(s) = \left(\mathbf{\Phi}(s)^{-1} + \mathbf{LC}\right)^{-1}(\mathbf{B}\underline{U}(s) + \mathbf{L}\underline{Y}(s)) \\ \underline{Y}(s) = \mathbf{C\Phi}(s)\mathbf{B}\underline{U}(s) \end{cases} \\ \begin{aligned} \Rightarrow \underline{\widehat{X}}(s) &= \left(\mathbf{\Phi}(s)^{-1} + \mathbf{LC}\right)^{-1}(\mathbf{B}\underline{U}(s) + \mathbf{LC\Phi}(s)\mathbf{B}\underline{U}(s)) \\ &= \left(\mathbf{\Phi}(s)^{-1} + \mathbf{LC}\right)^{-1}(\mathbf{B} + \mathbf{LC\Phi}(s)\mathbf{B})\,\underline{U}(s) \\ &= \left(\mathbf{\Phi}(s)^{-1} + \mathbf{LC}\right)^{-1}\mathbf{B}\underline{U}(s) \\ &\quad + \left(\mathbf{\Phi}(s)^{-1} + \mathbf{LC}\right)^{-1}\mathbf{LC\Phi}(s)\mathbf{B}\underline{U}(s) \end{aligned} \end{array} \tag{6.172}$$

Finally, the actual loop transfer function $\mathbf{L}_a(s)$ reads as follows:

$$\begin{array}{l} \underline{U}_o(s) = \mathbf{L}_a(s)\underline{U}(s) \\ \quad \text{where } \mathbf{L}_a(s) = -\mathbf{K}\left(\mathbf{\Phi}(s)^{-1} + \mathbf{LC}\right)^{-1}(\mathbf{B} + \mathbf{LC\Phi}(s))\,\mathbf{B} \end{array} \tag{6.173}$$

6.9.2 Loop Transfer Recovery (LTR) condition

Loop Transfer Recovery (LTR) will be achieved when the loop transfer function with state-feedback and with state-based observer are equal, that is when $\mathbf{L}_a(s) = \mathbf{L}_t(s)$, where loop transfer functions $\mathbf{L}_t(s)$ and $\mathbf{L}_a(s)$ are given by (6.167) and (6.173) respectively.

The matrix inversion lemma[5] is the equation:

$$(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1} = \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1} \tag{6.174}$$

Simple manipulations show that:

$$\left(\mathbf{\Phi}(s)^{-1} + \mathbf{L}\mathbf{C}\right)^{-1} = \mathbf{\Phi}(s)\left(\mathbb{I} - \mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\mathbf{\Phi}(s)\right) \tag{6.175}$$

So we have for the first term to the right of equation (6.172):

$$\left(\mathbf{\Phi}(s)^{-1} + \mathbf{L}\mathbf{C}\right)^{-1}\mathbf{B} = \mathbf{\Phi}(s)\left(\mathbb{I} - \mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}\right) \tag{6.176}$$

And for the second term to the right of equation (6.172):

$$\begin{aligned}\left(\mathbf{\Phi}(s)^{-1} + \mathbf{L}\mathbf{C}\right)^{-1}\mathbf{L}\mathbf{C}\mathbf{\Phi}(s)\mathbf{B} &= \mathbf{\Phi}(s)\left(\mathbb{I} - \mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\mathbf{\Phi}(s)\right)\mathbf{L}\mathbf{C}\mathbf{\Phi}(s)\mathbf{B} \\ &= \mathbf{\Phi}(s)\left(\mathbf{L} - \mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)\mathbf{C}\mathbf{\Phi}(s)\mathbf{B} \\ &= \mathbf{\Phi}(s)\mathbf{L}\left(\mathbb{I} - \left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}\end{aligned} \tag{6.177}$$

In addition, applying again the matrix inversion lemma to the following equality, we have:

$$\left(\mathbb{I} + \mathbf{A}\right)^{-1} = \mathbb{I} - \left(\mathbb{I} + \mathbf{A}\right)^{-1}\mathbf{A} \Leftrightarrow \left(\mathbb{I} + \mathbf{A}\right)^{-1}\mathbf{A} = \mathbb{I} - \left(\mathbb{I} + \mathbf{A}\right)^{-1} \tag{6.178}$$

Thus :

$$\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\mathbf{\Phi}(s)\mathbf{L} = \mathbb{I} - \left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1} \tag{6.179}$$

Applying this result to equation (6.177) leads to:

$$\left(\mathbf{\Phi}(s)^{-1} + \mathbf{L}\mathbf{C}\right)^{-1}\mathbf{L}\mathbf{C}\mathbf{\Phi}(s)\mathbf{B} = \mathbf{\Phi}(s)\mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\mathbf{\Phi}(s)\mathbf{B} \tag{6.180}$$

And here comes the light ! Indeed, if we impose:

$$\mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1} = \mathbf{B}\left(\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}\right)^{-1} \tag{6.181}$$

Then equations (6.176) and (6.180) become:

$$\left\{\begin{aligned}\left(\mathbf{\Phi}(s)^{-1} + \mathbf{L}\mathbf{C}\right)^{-1}\mathbf{B} &= \mathbf{\Phi}(s)\left(\mathbb{I} - \mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\Phi(s)\mathbf{B}\right) \\ &= \mathbf{\Phi}(s)\left(\mathbb{I} - \mathbf{B}\right) \\ \left(\mathbf{\Phi}(s)^{-1} + \mathbf{L}\mathbf{C}\right)^{-1}\mathbf{L}\mathbf{C}\mathbf{\Phi}(s)\mathbf{B} &= \mathbf{\Phi}(s)\mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1}\mathbf{C}\Phi(s)\mathbf{B} \\ &= \mathbf{\Phi}(s)\mathbf{B}\end{aligned}\right. \tag{6.182}$$

and thus, when summing those two terms, (6.172) reads:

$$\underline{\widehat{X}}(s) = \mathbf{\Phi}(s)\mathbf{B}\underline{U}(s) \tag{6.183}$$

[5] D. J. Tylavsky, G. R. L. Sohie, Generalization of the matrix inversion lemma, Proceedings of the IEEE, Year: 1986, Volume: 74, Issue: 7, Pages: 1050 - 1052

We finally get:

$$\underline{U}_o(s) = -\mathbf{K}\underline{\widehat{X}}(s) = -\mathbf{K}\mathbf{\Phi}(s)\mathbf{B}\underline{U}(s) \tag{6.184}$$

That is, we get for $\underline{U}_o(s)$ the same expression than the expression obtained through the full state-feedback given in (6.166).

As a conclusion, the Loop Transfer Recovery (LTR) is achieved when the loop transfer function with state-feedback and with state-based observer are equal, that is when $\mathbf{L}_a(s) = \mathbf{L}_t(s)$. This property is achieved as soon as $\underline{U}_o(s)$ has the same expression as the full state-feedback $\underline{U}_f(s)$, that is when the following relation holds:

$$\boxed{\mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1} = \mathbf{B}\left(\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}\right)^{-1}} \tag{6.185}$$

6.9.3 Setting the Loop Transfer Recovery design parameter

Condition (6.185) is not an easy condition to satisfy. The traditional approaches to this problem is to design matrix $\mathbf{L}$ of the observer such that the condition is satisfied asymptotically and ρ a design parameter.

One way to asymptotically satisfy (6.185) is to set $\mathbf{L}$ such that:

$$\lim_{\rho\to\infty} \frac{\mathbf{L}}{\rho} = \mathbf{B}\mathbf{W}_0 \tag{6.186}$$

where $\mathbf{W}_0$ is a non-singular matrix.

Indeed in this case we have:

$$\begin{aligned} \mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1} &= \tfrac{\mathbf{L}}{\rho}\rho\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1} \\ &= \tfrac{\mathbf{L}}{\rho}\left(\tfrac{1}{\rho}\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\tfrac{\mathbf{L}}{\rho}\right)^{-1} \end{aligned} \tag{6.187}$$

Thus, as $\rho \to \infty$:

$$\begin{aligned} \lim_{\rho\to\infty} \mathbf{L}\left(\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\mathbf{L}\right)^{-1} &= \lim_{\rho\to\infty} \tfrac{\mathbf{L}}{\rho}\left(\tfrac{1}{\rho}\mathbb{I} + \mathbf{C}\mathbf{\Phi}(s)\tfrac{\mathbf{L}}{\rho}\right)^{-1} \\ &= \lim_{\rho\to\infty} \tfrac{\mathbf{L}}{\rho}\left(\mathbf{C}\mathbf{\Phi}(s)\tfrac{\mathbf{L}}{\rho}\right)^{-1} \\ &= \mathbf{B}\mathbf{W}_0\left(\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}\mathbf{W}_0\right)^{-1} \\ &= \mathbf{B}\left(\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}\right)^{-1} \end{aligned} \tag{6.188}$$

Now let's concentrate how (6.186) can be achieved. First we have seen in (6.143) that if the transfer function $\mathbf{C}\mathbf{\Phi}(s)\mathbf{B}$ is right invertible with no unstable zeros then for some unitary matrix $\mathbf{W}$ ($\mathbf{W}^T\mathbf{W} = \mathbb{I}$) and some symmetric positive definite matrix $\mathbf{M}$ ($\mathbf{M} = \mathbf{M}^T > 0$), the asymptotic value of feedback gain $\mathbf{K}$ reads as follows, where ϵ has been replaced by $\frac{1}{\rho}$:

$$\begin{cases} \mathbf{R} = \frac{\mathbf{M}}{\rho^2} \\ \mathbf{Q} = \mathbf{C}^T\mathbf{C} \\ \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \end{cases} \Rightarrow \lim_{\rho\to\infty} \mathbf{K} = \rho\mathbf{M}^{-0.5}\mathbf{W}\mathbf{C} = \mathbf{R}^{-0.5}\mathbf{W}\mathbf{C} \tag{6.189}$$

Applying the duality principle we have the same result for the asymptotic value of the observer gain $\mathbf{L}$:

$$\begin{cases} \mathbf{P}_v = \frac{\mathbf{M}}{\rho^2} \\ \mathbf{P}_w = \mathbf{B}\mathbf{B}^T \\ \mathbf{L} = \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1} \end{cases} \Rightarrow \lim_{\rho\to\infty} \mathbf{L} = \rho\mathbf{B}\mathbf{W}\mathbf{M}^{-0.5} = \mathbf{B}\mathbf{W}\mathbf{P}_v^{-0.5} \tag{6.190}$$

Then, we concentrate on input recovery (6.152). We design a new observer with the following design matrices:

$$\begin{cases} \mathbf{P}_w = \mathbf{P}_{w0} + \rho^2\mathbf{B}\mathbf{B}^T \\ \mathbf{P}_v = \mathbf{P}_{v0} \end{cases} \tag{6.191}$$

Then if we replace $\mathbf{P}_v$ by $\mathbf{P}_{v0}$ and $\mathbf{P}_w = \mathbf{B}\mathbf{B}^T$ by $\mathbf{P}_w = \mathbf{P}_{w0} + \rho^2\mathbf{B}\mathbf{B}^T$ in (6.190), the asymptotic value of the observer gain $\mathbf{L}$ reads as follows:

$$\begin{cases} \mathbf{P}_v = \mathbf{P}_{v0} \\ \mathbf{P}_w = \mathbf{P}_{w0} + \rho^2\mathbf{B}\mathbf{B}^T \\ \mathbf{L} = \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1} \end{cases} \Rightarrow \lim_{\rho\to\infty} \mathbf{L} = \rho\mathbf{B}\mathbf{W}\mathbf{P}_{v0}^{-0.5} \Rightarrow \lim_{\rho\to\infty} \frac{\mathbf{L}}{\rho} = \mathbf{B}\mathbf{W}\mathbf{P}_{v0}^{-0.5} \tag{6.192}$$

By setting $\mathbf{W}_0 := \mathbf{W}\mathbf{P}_{v0}^{-0.5}$ we finally get (6.186):

$$\lim_{\rho\to\infty} \frac{\mathbf{L}}{\rho} = \mathbf{B}\mathbf{W}_0 \tag{6.193}$$

6.10 Robust control design

Robust control problems, and especially H_2 robust control problems, are solved in a dedicated framework presented in Figure 6.9 where:

- $\mathbf{G}(s)$ is the transfer function of the *generalized* plant;
- $\mathbf{K}(s)$ is the transfer function of the controller ;
- $\underline{u}$ is the control vector of the *generalized* plant $\mathbf{G}(s)$ which is computed by the controller $\mathbf{K}(s)$;
- $\underline{w}$ is the input vector formed by *exogenous inputs* such as disturbances or noise;
- $\underline{y}$ is the vector of output available for the controller $\mathbf{K}(s)$;
- $\underline{z}$ is the performance output vector, also-called the *controlled* output, that is the vector that allows to characterize the performance of the closed-loop system. This is a *virtual* output used only for design that we wish to maintain as small as possible.

It is worth noticing that in the standard feedback control loop in Figure 6.9 *all reference signals are set to zero.*

The H_2 control problem consists in finding the optimal controller $\mathbf{K}(s)$ which minimizes $\|\mathbf{T}_{zw}(s)\|_2$, that is the H_2 norm of the transfer between the exogenous inputs vector $\underline{w}$ and the vector of interest variables $\underline{z}$.

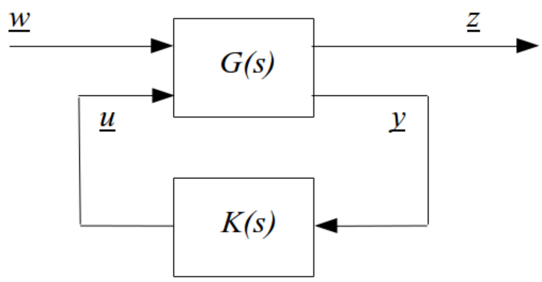


Figure 6.9: Standard feedback control loop

The general form of the realization of a plant is the following:

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \underline{z}(t) \\ \underline{y}(t) \end{bmatrix} = \left[\begin{array}{c|cc} \mathbf{A} & \mathbf{B}_1 & \mathbf{B}_2 \\ \hline \mathbf{C}_1 & \mathbf{0} & \mathbf{D}_{12} \\ \mathbf{C}_2 & \mathbf{D}_{21} & \mathbf{0} \end{array}\right] \begin{bmatrix} \underline{x}(t) \\ \underline{w}(t) \\ \underline{u}(t) \end{bmatrix} \tag{6.194}$$

Linear-quadratic-Gaussian (LQG) control is a special case of H_2 optimal control applied to stochastic system.

Let's consider the following system realization:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}_2\underline{u}(t) + \underline{d}(t) \\ \underline{y}(t) = \mathbf{C}_2\underline{x}(t) + \underline{n}(t) \end{cases} \tag{6.195}$$

Where $\underline{d}(t)$ and $\underline{n}(t)$ are white noise with the intensity of their autocorrelation function equals to $\mathbf{W}_d$ and $\mathbf{W}_n$ respectively. Denoting by $E()$ the mathematical expectation we have:

$$E\left(\begin{bmatrix} \underline{d}(t) \\ \underline{n}(t) \end{bmatrix} \begin{bmatrix} \underline{d}^T(\tau) & \underline{n}^T(\tau) \end{bmatrix}\right) = \begin{bmatrix} \mathbf{W}_d & 0 \\ 0 & \mathbf{W}_n \end{bmatrix} \delta(t-\tau) \tag{6.196}$$

The LQG problem consists in finding a controller $\underline{u}(s) = \mathbf{K}(s)\underline{y}(s)$ such that the following performance index is minimized:

$$J_{LQG} = E\left(\lim_{T\to\infty} \int_0^T \left(\underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t)\right) dt\right) \tag{6.197}$$

Where matrices $\mathbf{Q}$ ans $\mathbf{R}$ are symmetric and (semi)-positive definite matrices:

$$\begin{cases} \mathbf{Q} = \mathbf{Q}^T \geq 0 \\ \mathbf{R} = \mathbf{R}^T > 0 \end{cases} \tag{6.198}$$

This problem can be cast as the H_2 optimal control framework in the following manner. Define signal $\underline{z}(t)$ whose norm is to be minimized as follows:

$$\underline{z}(t) = \begin{bmatrix} \mathbf{Q}^{0.5} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}^{0.5} \end{bmatrix} \begin{bmatrix} \underline{x}(t) \\ \underline{u}(t) \end{bmatrix} \tag{6.199}$$

And represent the stochastic inputs $\underline{d}(t)$ and $\underline{n}(t)$ as a function of the vector $\underline{w}(t)$ of exogenous disturbances :

$$\begin{bmatrix} \underline{d}(t) \\ \underline{n}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{W}_d^{0.5} & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_n^{0.5} \end{bmatrix} \underline{w}(t) \tag{6.200}$$

Where $\underline{w}(t)$ is a white noise process of unit intensity. Then the LQG cost function reads as follows:

$$J_{LQG} = E\left(\lim_{T \to \infty} \int_0^T \underline{z}^T(t)\underline{z}(t)dt \right) = \|\mathbf{T}_{zw}(s)\|_2^2 \tag{6.201}$$

And the generalized plant reads as follows:

$$\begin{bmatrix} \underline{\dot{x}}(t) \\ \underline{z}(t) \\ \underline{y}(t) \end{bmatrix} = \left[\begin{array}{c|cc} \mathbf{A} & \mathbf{B}_1 & \mathbf{B}_2 \\ \hline \mathbf{C}_1 & \mathbf{0} & \mathbf{D}_{12} \\ \mathbf{C}_2 & \mathbf{D}_{21} & \mathbf{0} \end{array} \right] \begin{bmatrix} \underline{x}(t) \\ \underline{w}(t) \\ \underline{u}(t) \end{bmatrix} \tag{6.202}$$

Where:

$$\begin{cases} \mathbf{B}_1 = \begin{bmatrix} \mathbf{W}_d^{0.5} & \mathbf{0} \end{bmatrix} \\ \mathbf{C}_1 = \begin{bmatrix} \mathbf{Q}^{0.5} \\ \mathbf{0} \end{bmatrix} \\ \mathbf{D}_{12} = \begin{bmatrix} \mathbf{0} \\ \mathbf{R}^{0.5} \end{bmatrix} \\ \mathbf{D}_{21} = \begin{bmatrix} \mathbf{0} & \mathbf{W}_n^{0.5} \end{bmatrix} \end{cases} \tag{6.203}$$

It follows that:

$$\begin{cases} \mathbf{B}_1\underline{w}(t) = \begin{bmatrix} \mathbf{W}_d^{0.5} & \mathbf{0} \end{bmatrix} \underline{w}(t) = \underline{d}(t) \\ \mathbf{D}_{21}\underline{w}(t) = \begin{bmatrix} \mathbf{0} & \mathbf{W}_n^{0.5} \end{bmatrix} \underline{w}(t) = \underline{n}(t) \end{cases} \tag{6.204}$$

And:

$$\begin{array}{ll} \underline{z}^T(t)\underline{z}(t) & = \left(\mathbf{C}_1\underline{x}(t) + \mathbf{D}_{12}\underline{w}(t)\right)^T \left(\mathbf{C}_1\underline{x}(t) + \mathbf{D}_{12}\underline{w}(t)\right) \\ & = \underline{x}^T(t)\mathbf{Q}\underline{x}(t) + \underline{u}^T(t)\mathbf{R}\underline{u}(t) \end{array} \tag{6.205}$$

Thus costs (6.197) and (6.201) are equivalent.

6.11 Sensor data fusion

6.11.1 Complementary filter

Sensor data fusion considers the problem to integrate redundant measurement information from separate sensor systems.

The basic idea of complementary filter consists in taking the measurements of two sensors, filtering out low-frequency and high-frequency noises of each sensor, and combining the filtered outputs to get a better estimate of the signal

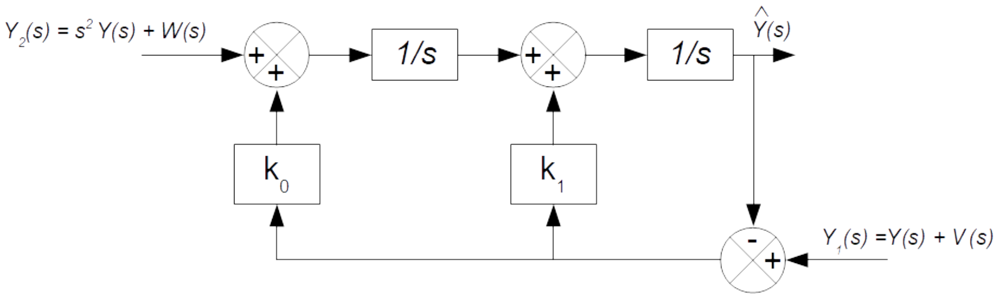


Figure 6.10: Complementary filter implementing fusion between baro-altimeter and vertical accelerometer measurements

of interest. An example of two sensors that complement each other are baro-altimeter and vertical accelerometer.

Let $y_1(t)$ and $y_2(t)$ noisy measurements of some signal $y(t)$, coming for example from a baro-altimeter and a vertical accelerometer, respectively. Denoting by $v(t)$ some low frequency zero mean noise process, by $w(t)$ some high frequency zero mean noise process and by s the Laplace variable, we will assume that:

$$\left\{ \begin{array}{l} y_1(t) = y(t) + v(t) \\ y_2(t) = \ddot{y}(t) + w(t) \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} Y_1(s) = Y(s) + V(s) \\ Y_2(s) = s^2\,Y(s) + W(s) \end{array} \right. \tag{6.206}$$

The complementary filter that implements the fusion between the two measurements is shown in Figure 6.10[6].

From Figure 6.10, the expression of $\widehat{Y}(s)$ reads:

$$\begin{array}{rl} \widehat{Y}(s) & = \frac{1}{s}\left(k_1\left(Y_1(s) - \widehat{Y}(s)\right)\right. \\ & \quad \left. + \frac{1}{s}\left(Y_2(s) + k_0\left(Y_1(s) - \widehat{Y}(s)\right)\right)\right) \\ \Leftrightarrow \left(1 + \frac{k_1}{s} + \frac{k_0}{s^2}\right)\widehat{Y}(s) & = \left(\frac{k_1}{s} + \frac{k_0}{s^2}\right)Y_1(s) + \frac{1}{s^2}Y_2(s) \\ \Leftrightarrow \widehat{Y}(s) & = \frac{k_1\,s + k_0}{s^2 + k_1\,s + k_0}Y_1(s) + \frac{1}{s^2 + k_1\,s + k_0}Y_2(s) \end{array} \tag{6.207}$$

Using the fact that $Y_1(s) = Y(s) + V(s)$ and $Y_2(s) = s^2\,Y(s) + W(s)$, we finally get:

$$\begin{array}{rl} \widehat{Y}(s) & = Y(s) + \frac{k_1\,s + k_0}{s^2 + k_1\,s + k_0}V(s) + \frac{1}{s^2 + k_1\,s + k_0}W(s) \\ & := Y(s) + F(s)\,V(s) + \frac{1 - F(s)}{s^2}\,W(s) \end{array} \tag{6.208}$$

where:

$$F(s) := \frac{k_1\,s + k_0}{s^2 + k_1\,s + k_0} \Leftrightarrow 1 - F(s) := \frac{s^2}{s^2 + k_1\,s + k_0} \tag{6.209}$$

Transfer function $F(s)$ is a low-pass filter with unity static gain whereas $1 - F(s)$ is a high-pass filter.

[6] W. T. Higgins, A Comparison of Complementary and Kalman Filtering, IEEE Transactions on Aerospace and Electronic Systems, vol. AES-11, no. 3, pp. 321-325, May 1975, doi: 10.1109/TAES.1975.308081.

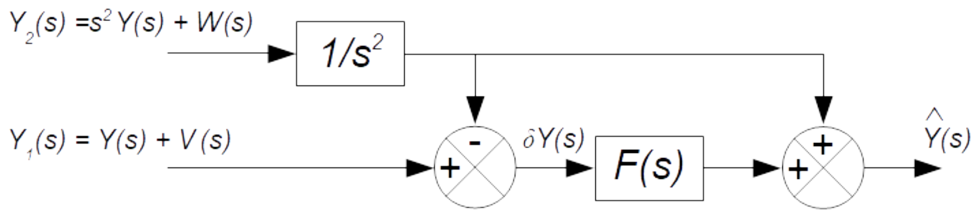


Figure 6.11: Equivalent filter implementing fusion between baro-altimeter and vertical accelerometer measurements

Thanks to the expression of $F(s)$, relation (6.207) can also be written as follows:

$$\begin{array}{ll}\widehat{Y}(s) & = F(s)\, Y_1(s) + \left(\frac{1-F(s)}{s^2}\right) Y_2(s) \\ & = \frac{1}{s^2}\, Y_2(s) + F(s)\, \left(Y_1(s) - \frac{1}{s^2}\, Y_2(s)\right)\end{array} \tag{6.210}$$

Relation (6.210) leads to an equivalent version of the complementary filter shown in Figure 6.11 where the low-pass filter $F(s)$ operates only on the noises.

6.11.2 Kalman filter

More generally, there are two measurements for sensor data fusion problems, $\underline{y}_1(t)$ and $\underline{y}_2(t)$, and one of it serves as an input to the state equation, which is seen as the process model. Denoting $\underline{v}(t)$ the random process which represents the measurement noise and $\underline{\widetilde{x}}(t)$ the *noisy* state vector, we have:

$$\boxed{\left\{\begin{array}{ll}\dot{\underline{\widetilde{x}}}(t) = \mathbf{A}\, \underline{\widetilde{x}}(t) + \mathbf{B}\, \underline{y}_2(t) & \text{(process)} \\ \underline{y}_1(t) = \mathbf{C}\, \underline{x}(t) + \underline{v}(t) & \text{(measurement)}\end{array}\right.} \tag{6.211}$$

Take care that in the *measurement* equation, this is the *actual* state vector $\underline{x}(t)$ which is used, not the *noisy* state vector $\underline{\widetilde{x}}(t)$ of the *process* equation.

Furthermore, denoting $\underline{w}(t)$ the random processes which represents the process noise, and $\underline{y}_2(t) := \underline{u}(t) + \underline{w}(t)$, we finally get:

$$\underline{y}_2(t) := \underline{u}(t) + \underline{w}(t) \Rightarrow \left\{\begin{array}{l}\dot{\underline{\widetilde{x}}}(t) = \mathbf{A}\, \underline{\widetilde{x}}(t) + \mathbf{B}\, \left(\underline{u}(t) + \underline{w}(t)\right) \\ \underline{y}_1(t) = \mathbf{C}\, \underline{x}(t) + \underline{v}(t)\end{array}\right. \tag{6.212}$$

Assuming no noise, $\underline{v}(t) = \underline{w}(t) = \underline{0}$, then $\underline{\widetilde{x}}(t)$ changes into its *noiseless* value $\underline{x}(t)$ and actual measurement $\underline{y}_1(t)$ changes into its *noiseless* value $\underline{y}(t)$. Then we get the following *noiseless* state equation:

$$\underline{v}(t) = \underline{w}(t) = \underline{0} \Rightarrow \left\{\begin{array}{l}\dot{\underline{x}}(t) = \mathbf{A}\, \underline{x}(t) + \mathbf{B}\, \underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\, \underline{x}(t)\end{array}\right. \tag{6.213}$$

The error equations reads as follows where $\delta\underline{x}(t)$ is the error state vector and $\delta\underline{y}(t)$ the error output vector. Note that we define $\delta\underline{y}(t)$ as $\delta\underline{y}(t) := \underline{y}_1(t) - \mathbf{C}\, \underline{\widetilde{x}}(t)$ to be compliant with Figure 6.11:

$$\left\{\begin{array}{l}\delta\underline{x}(t) := \underline{\widetilde{x}}(t) - \underline{x}(t) \\ \delta\underline{y}(t) := \underline{y}_1(t) - \mathbf{C}\, \underline{\widetilde{x}}(t)\end{array}\right. \Rightarrow \left\{\begin{array}{l}\delta\dot{\underline{x}}(t) = \mathbf{A}\, \delta\underline{x}(t) + \mathbf{B}\, \underline{w}(t) \\ \delta\underline{y}(t) = -\mathbf{C}\, \delta\underline{x}(t) + \underline{v}(t)\end{array}\right. \tag{6.214}$$

According to (6.10), the dynamics of the observer, which is actually a Kalman-Bucy filter, reads:

$$\begin{aligned}\delta\dot{\widehat{\underline{x}}}(t) &= \mathbf{A}\,\delta\widehat{\underline{x}}(t) + \mathbf{L}(t)\left(\delta\underline{y}(t) - (-\mathbf{C})\ \delta\widehat{\underline{x}}(t)\right) \\ &= \mathbf{A}\,\delta\widehat{\underline{x}}(t) + \mathbf{L}(t)\left(\delta\underline{y}(t) + \mathbf{C}\,\delta\widehat{\underline{x}}(t)\right)\end{aligned} \tag{6.215}$$

The time dependent observer gain $\mathbf{L}(t)$, also-called Kalman gain, is similar to (6.71):

$$\mathbf{L}(t) = \mathbf{Y}(t)\,(-\mathbf{C})^T\,\mathbf{P}_v^{-1} = -\mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1} \tag{6.216}$$

where matrix $\mathbf{Y}(t)$ is the solution of the following *differential* Riccati equation (see (6.72) where $\mathbf{P}_w$ has been replaced by $\mathbf{B}\mathbf{P}_w\mathbf{B}^T$, that is the covariance of noise $\mathbf{B}\,\underline{w}(t)$):

$$\dot{\mathbf{Y}}(t) = \mathbf{A}\mathbf{Y}(t) + \mathbf{Y}(t)\mathbf{A}^T - \mathbf{Y}(t)\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y}(t) + \mathbf{B}\mathbf{P}_w\mathbf{B}^T \tag{6.217}$$

The steady-state Kalman filter is achieved when $\dot{\mathbf{Y}}(t) = \mathbf{0}$. Then the preceding *differential* Riccati equation turns to be the *algebraic* Riccati equation and matrix $\mathbf{Y}$ is the *constant* positive solution of the following *algebraic* Riccati equation:

$$\mathbf{0} = \mathbf{A}\mathbf{Y} + \mathbf{Y}\mathbf{A}^T - \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y} + \mathbf{B}\mathbf{P}_w\mathbf{B}^T \tag{6.218}$$

Furthermore the actual estimate of the signal reads:

$$\delta\widehat{\underline{x}}(t) = \widetilde{\underline{x}}(t) - \widehat{\underline{x}}(t) \Leftrightarrow \widehat{\underline{x}}(t) = \widetilde{\underline{x}}(t) - \delta\widehat{\underline{x}}(t) \tag{6.219}$$

By taking the time derivative of the preceding equation, and using (6.211) and (6.214), we get the state equation for the estimate of the actual state vector $\widehat{\underline{x}}(t)$:

$$\begin{aligned}\dot{\widehat{\underline{x}}}(t) &= \dot{\widetilde{\underline{x}}}(t) - \delta\dot{\widehat{\underline{x}}}(t) \\ &= \mathbf{A}\,\widetilde{\underline{x}}(t) + \mathbf{B}\,\underline{y}_2(t) - \left(\mathbf{A}\,\delta\widehat{\underline{x}}(t) + \mathbf{L}(t)\left(\delta\underline{y}(t) + \mathbf{C}\,\delta\widehat{\underline{x}}(t)\right)\right) \\ &= \mathbf{A}\ \left(\widetilde{\underline{x}}(t) - \delta\widehat{\underline{x}}(t)\right) + \mathbf{B}\,\underline{y}_2(t) \\ &\quad +\mathbf{L}(t)\left(\underline{y}_1(t) - \cancel{\mathbf{C}\,\widetilde{\underline{x}}(t)} + \mathbf{C}\ \left(\cancel{\widetilde{\underline{x}}(t)} - \widehat{\underline{x}}(t)\right)\right)\end{aligned} \tag{6.220}$$

We finally get:

$$\boxed{\dot{\widehat{\underline{x}}}(t) = \mathbf{A}\,\widehat{\underline{x}}(t) + \mathbf{B}\,\underline{y}_2(t) + \mathbf{L}(t)\left(\underline{y}_1(t) - \mathbf{C}\,\widehat{\underline{x}}(t)\right)} \tag{6.221}$$

6.11.3 Relation between complementary filter and Kalman filter

By taking the Laplace transform of (6.221), and assuming that $\mathbf{L}(t)$ is constant, we get:

$$\begin{aligned}\mathbf{L}(t) := \mathbf{L} \Rightarrow s\,\widehat{\underline{X}}(s) &= \mathbf{A}\,\widehat{\underline{X}}(s) + \mathbf{B}\,\underline{Y}_2(s) + \mathbf{L}\left(\underline{Y}_1(s) - \mathbf{C}\,\widehat{\underline{X}}(s)\right) \\ \Rightarrow \widehat{\underline{X}}(s) &= (s\mathbb{I} - (\mathbf{A} - \mathbf{L}\mathbf{C}))^{-1}\left(\mathbf{L}\,\underline{Y}_1(s) + \mathbf{B}\,\underline{Y}_2(s)\right)\end{aligned} \tag{6.222}$$

Then let $\widehat{y}(t) = \mathbf{C}\,\widehat{\underline{x}}(t)$. Multiplying (6.222) by $\mathbf{C}$ yields:

$$\mathbf{C}\,\widehat{\underline{X}}(s) = \widehat{\underline{Y}}(s) = \mathbf{C}\,\left(s\mathbb{I} - (\mathbf{A} - \mathbf{L}\mathbf{C})\right)^{-1}\left(\mathbf{L}\,\underline{Y}_1(s) + \mathbf{B}\,\underline{Y}_2(s)\right) \tag{6.223}$$

Comparing the preceding relation with (6.210), we conclude that complementary filter and Kalman filter are equivalent. Furthermore the transfer function $F(s)$ of the low-pass filter reads as follows:

$$\boxed{F(s) = \mathbf{C}\,\left(s\mathbb{I} - (\mathbf{A} - \mathbf{L}\mathbf{C})\right)^{-1}\,\mathbf{L}} \tag{6.224}$$

Example 6.3. *In the specific case of sensor fusion between baro-altimeter and vertical accelerometer presented in* (6.206)*, the state vector can be chosen as follows, assuming no noise:*

$$\begin{cases} x_1(t) := y(t) \\ x_2(t) := \dot{y}(t) \end{cases} \tag{6.225}$$

Then (6.211) *reads:*

$$\begin{cases} \begin{bmatrix} \dot{\widetilde{x}}_1(t) \\ \dot{\widetilde{x}}_2(t) \end{bmatrix} & = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \widetilde{x}_1(t) \\ \widetilde{x}_2(t) \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} (\ddot{y}(t) + w(t)) \\ & := \mathbf{A}\,\underline{\widetilde{x}}(t) + \mathbf{B}\,\underline{y}_2(t) \\ \underline{y}_1(t) & = \begin{bmatrix} 1 & 0 \end{bmatrix} \underline{x}(t) + \underline{v}(t) \\ & := \mathbf{C}\,\underline{x}(t) + \underline{v}(t) \end{cases} \tag{6.226}$$

Let $\mathbf{L}$ *be the steady state observer gain, also-called steady state Kalman gain, which is obtained as follows:*

$$\mathbf{L} = -\mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1} \tag{6.227}$$

where matrix $\mathbf{Y}$ *is the constant positive solution of the following* algebraic *Riccati equation:*

$$\mathbf{0} = \mathbf{A}\mathbf{Y} + \mathbf{Y}\mathbf{A}^T - \mathbf{Y}\mathbf{C}^T\mathbf{P}_v^{-1}\mathbf{C}\mathbf{Y} + \mathbf{B}\mathbf{P}_w\mathbf{B}^T \tag{6.228}$$

Then, according to (6.224)*, the transfer function* $F(s)$ *of the low-pass filter reads as follows:*

$$\mathbf{L} := \begin{bmatrix} k_1 \\ k_0 \end{bmatrix} \Rightarrow F(s) \begin{aligned} & = \mathbf{C}\,\left(s\mathbb{I} - (\mathbf{A} - \mathbf{L}\mathbf{C})\right)^{-1}\,\mathbf{L} \\ & = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s + k_1 & -1 \\ k_0 & s \end{bmatrix}^{-1} \begin{bmatrix} k_1 \\ k_0 \end{bmatrix} \\ & = \frac{k_1\,s + k_0}{s^2 + k_1\,s + k_0} \end{aligned} \tag{6.229}$$

We then retrieve the expression of $F(s)$ *obtained in* (6.209)*.*

■

More generally, and following Higgins[6], typical application of the Kalman filter in navigation systems extends Figure 6.11 as shown in Figure 6.12, although, as seen before, the actual implementation may be different. Note that the Kalman filter just operates on noises and is not affected by actual signals that are to be estimated.

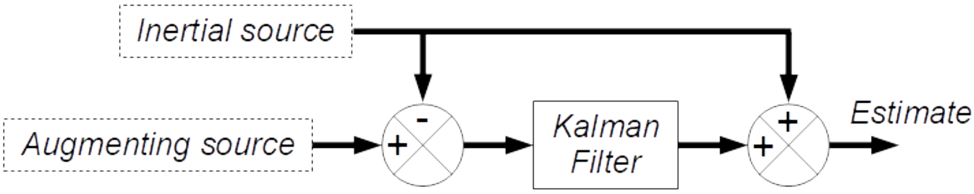


Figure 6.12: Typical application of the Kalman filter in inertial navigation

6.12 Euler angles estimation

6.12.1 One dimensional attitude estimation

System description

We consider the simple pendulum in Figure 6.13 fitted with an accelerometer and a gyroscope in the ball at the end of the arm.

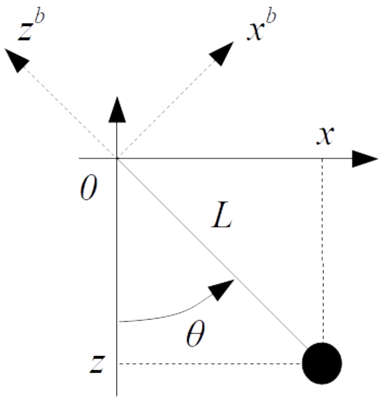


Figure 6.13: Simple pendulum

Accelerometer measurement

First, we compute the position, velocity and acceleration in the inertial frame:

– Inertial position:

$$\begin{cases} x = L\,\sin(\theta) \\ z = -L\,\cos(\theta) \end{cases} \tag{6.230}$$

– Inertial velocity:

$$\begin{cases} \dot{x} = L\,\dot{\theta}\,\cos(\theta) \\ \dot{z} = L\,\dot{\theta}\,\sin(\theta) \end{cases} \tag{6.231}$$

– Inertial acceleration:

$$\begin{cases} \ddot{x} = L\,\ddot{\theta}\,\cos(\theta) - L\,\dot{\theta}^2\,\sin(\theta) \\ \ddot{z} = L\,\ddot{\theta}\,\sin(\theta) + L\,\dot{\theta}^2\,\cos(\theta) \end{cases} \tag{6.232}$$

Let a_x and a_z the x-component and z-component provided by the accelerometer. Because the accelerometer is linked to the body frame, and denoting by $\mathbf{R}_i^b$ the rotation matrix from the inertial frame to the body frame, its provides the following data, known as *specific acceleration*:

$$\begin{bmatrix} a_x \\ a_z \end{bmatrix} = \mathbf{R}_i^b \left(\begin{bmatrix} \ddot{y} \\ \ddot{z} \end{bmatrix} - \begin{bmatrix} 0 \\ -g \end{bmatrix} \right) \tag{6.233}$$

where:

$$\mathbf{R}_i^b = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \tag{6.234}$$

Thus:

$$\begin{bmatrix} a_x \\ a_z \end{bmatrix} = \begin{bmatrix} L\,\ddot{\theta} + g\,\sin(\theta) \\ L\,\dot{\theta}^2 + g\,\cos(\theta) \end{bmatrix} \tag{6.235}$$

Then neglecting $\dot{\theta}^2$ and $\ddot{\theta}$ we get:

$$\frac{a_x}{a_z} = \frac{L\,\ddot{\theta} + g\,\sin(\theta)}{L\,\dot{\theta}^2 + g\,\cos(\theta)} \approx \frac{g\,\sin(\theta)}{g\,\cos(\theta)} = \tan(\theta) \tag{6.236}$$

Thus a first approximation of angle θ, provided by the accelerometer, is θ_a where:

$$\theta_a \approx \arctan\left(\frac{a_x}{a_z}\right) \tag{6.237}$$

Attitude estimation problem

For this one dimensional example, the gyroscope provides $q := \dot{\theta}$. Then attitude estimation problem consists in computing an estimate of θ from noisy measurements of a_x, a_z and q.

6.12.2 One dimensional complementary filter

Sensor data fusion considers the problem to integrate redundant measurement information from separate sensor systems.

The basic idea of complementary filter consists in taking the measurements of two sensors, filtering out low-frequency and high-frequency noises of each sensor, and combining the filtered outputs to get a better estimate of the signal of interest. An example of two sensors that complement each other are gyro and accelerometer.

Let $y_1(t)$ and $y_2(t)$ noisy measurements of some signal $y(t)$, coming for example from an accelerometer and a gyroscope, respectively. Denoting by $v(t)$ some low frequency zero mean noise process, by $w(t)$ some high frequency zero mean noise process and by s the Laplace variable, we will assume that:

$$\left\{ \begin{array}{l} y_1(t) = y(t) + v(t) \\ y_2(t) = \dot{y}(t) + w(t) \end{array} \right. \Leftrightarrow \left\{ \begin{array}{l} Y_1(s) = Y(s) + V(s) \\ Y_2(s) = s\,Y(s) + W(s) \end{array} \right. \tag{6.238}$$

For the example of section 6.12.1, we have $y_1(t) := \theta_a(t)$ and $y_2(t) := q(t)$.

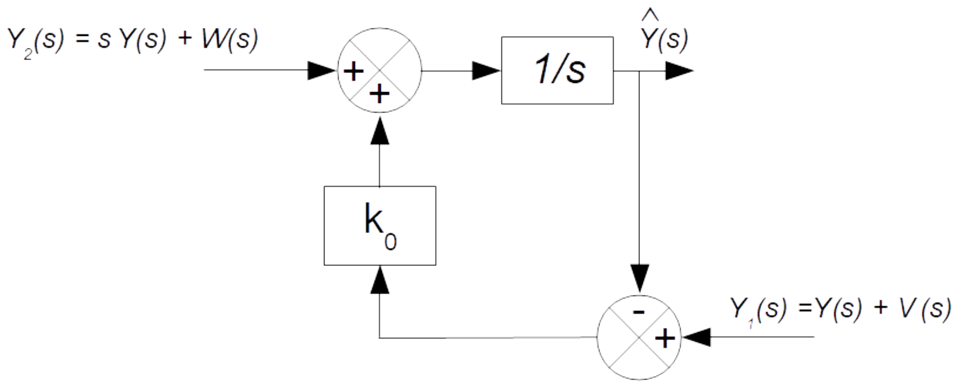


Figure 6.14: Complementary filter implementing fusion between gyro and accelerometer measurements

The complementary filter that implements the fusion between the two measurements is shown in Figure 6.14[7].

From Figure 6.14, the expression of $\widehat{Y}(s)$ reads:

$$\begin{array}{rl} \widehat{Y}(s) & = \frac{1}{s}\left(Y_2(s) + k_0\left(Y_1(s) - \widehat{Y}(s)\right)\right) \\ \Leftrightarrow \left(1 + \frac{k_0}{s}\right)\widehat{Y}(s) & = \frac{k_0}{s}\,Y_1(s) + \frac{1}{s}\,Y_2(s) \\ \Leftrightarrow \widehat{Y}(s) & = \frac{k_0}{s+k_0}Y_1(s) + \frac{1}{s+k_0}Y_2(s) \end{array} \tag{6.239}$$

Using the fact that $Y_1(s) = Y(s) + V(s)$ and $Y_2(s) = s\,Y(s) + W(s)$, we finally get:

$$\begin{array}{rl} \widehat{Y}(s) & = Y(s) + \frac{k_0}{s+k_0}\,V(s) + \frac{1}{s+k_0}\,W(s) \\ & := Y(s) + F(s)\,V(s) + \frac{1-F(s)}{s}\,W(s) \end{array} \tag{6.240}$$

where:

$$F(s) := \frac{k_0}{s+k_0} \Leftrightarrow 1 - F(s) := \frac{s}{s+k_0} \tag{6.241}$$

Transfer function $F(s)$ is a low-pass filter with unity static gain whereas $1 - F(s)$ is a high-pass filter.

In the continuous time domain, (6.239) reads:

$$\begin{array}{rl} \widehat{Y}(s) & = \frac{k_0}{s+k_0}Y_1(s) + \frac{1}{s+k_0}Y_2(s) \\ \Leftrightarrow k_0\,\widehat{Y}(s) + s\,\widehat{Y}(s) & = k_0\,Y_1(s) + Y_2(s) \\ \Rightarrow k_0\,\widehat{y}(t) + \frac{d}{dt}\widehat{y}(t) & = k_0\,y_1(t) + y_2(t) \end{array} \tag{6.242}$$

In order to discretize this continuous time complementary filter, we have to find an approximation of the differentiation operator $\frac{d}{dt}$. Let T_s be the sampling period and denote z^{-1} the sample period delay operator. Several options can

[7] W. T. Higgins, A Comparison of Complementary and Kalman Filtering, IEEE Transactions on Aerospace and Electronic Systems, vol. AES-11, no. 3, pp. 321-325, May 1975, doi: 10.1109/TAES.1975.308081.

be used to approximate the differentiation operator. Using backward difference we get:

$$\frac{d}{dt} \approx \frac{1 - z^{-1}}{T_s} \tag{6.243}$$

Then (6.242) is approximated as follows at discrete-time $t = k\,T_s$:

$$\begin{array}{rl} k_0\,\widehat{y}(k\,T_s) + \dot{\widehat{y}}(k\,T_s) & = k_0\,y_1(k\,T_s) + y_2(k\,T_s) \\ \Rightarrow k_0\,\widehat{y}(k\,T_s) + \frac{\widehat{y}(k\,T_s) - \widehat{y}((k-1)\,T_s)}{T_s} & \approx k_0\,y_1(k\,T_s) + y_2(k\,T_s) \end{array} \tag{6.244}$$

Usually the sampling period is omitted in the expression of the discrete-time, and the preceding relation is written as follows:

$$k_0\,\widehat{y}_k + \frac{\widehat{y}_k - \widehat{y}_{k-1}}{T_s} \approx k_0\,y_{1_k} + y_{2_k} \tag{6.245}$$

We finally get:

$$\widehat{y}_k \approx \frac{1}{1 + k_0\,T_s}\left(\widehat{y}_{k-1} + T_s\,\left(k_0\,y_{1_k} + y_{2_k}\right)\right) \tag{6.246}$$

Or equivalently:

$$\widehat{y}_k \approx \alpha\,\left(\widehat{y}_{k-1} + T_s\,y_{2_k}\right) + (1-\alpha)\,y_{1_k} \text{ where } \alpha := \frac{1}{1 + k_0\,T_s} \tag{6.247}$$

6.12.3 Direct Cosine Matrix (DCM) and kinematic relations

Let $\underline{x}^i$ be a vector expressed in the inertial frame, $\underline{x}^b$ a vector expressed in the body frame and $\mathbf{R}_i^b(\underline{\eta})$ the rotation matrix, also called Direct Cosine Matrix (DCM), from the inertial frame to the body frame:

$$\underline{x}^b = \mathbf{R}_i^b(\underline{\eta})\underline{x}^i \tag{6.248}$$

Rotation matrix $\mathbf{R}_i^b(\underline{\eta})$ is obtained by the multiplication of the rotation matrices around Euler angles, namely yaw angle ψ, pitch angle θ and then roll angle ϕ, respectively. Denoting $c_x = \cos(x)$, $s_x = \sin(x)$ and $\mathbf{R}_y$ the rotation matrix dedicated to angle y we get:

$$\begin{array}{rl} \mathbf{R}_i^b(\underline{\eta}) & = \mathbf{R}_\phi \mathbf{R}_\theta \mathbf{R}_\psi \\ & = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\phi & s_\phi \\ 0 & -s_\phi & c_\phi \end{bmatrix} \begin{bmatrix} c_\theta & 0 & -s_\theta \\ 0 & 1 & 0 \\ s_\theta & 0 & c_\theta \end{bmatrix} \begin{bmatrix} c_\psi & s_\psi & 0 \\ -s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ & = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ (s_\phi s_\theta c_\psi - c_\phi s_\psi) & (s_\phi s_\theta s_\psi + c_\phi c_\psi) & s_\phi c_\theta \\ (c_\phi s_\theta c_\psi + s_\phi s_\psi) & (c_\phi s_\theta s_\psi - s_\phi c_\psi) & c_\phi c_\theta \end{bmatrix} \end{array} \tag{6.249}$$

It is worth noticing that $\mathbf{R}_i^b(\underline{\eta})$ is an orthogonal matrix. Consequently the rotation matrix $\mathbf{R}_b^i(\underline{\eta})$ from the body frame to the inertial frame is obtained as follows:

$$\begin{array}{rl} \mathbf{R}_b^i(\underline{\eta}) & := \left(\mathbf{R}_i^b(\underline{\eta})\right)^{-1} = \left(\mathbf{R}_i^b(\underline{\eta})\right)^T \\ & = \begin{bmatrix} c_\theta c_\psi & (s_\phi s_\theta c_\psi - c_\phi s_\psi) & (c_\phi s_\theta c_\psi + s_\phi s_\psi) \\ c_\theta s_\psi & (s_\phi s_\theta s_\psi + c_\phi c_\psi) & (c_\phi s_\theta s_\psi - s_\phi c_\psi) \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix} \end{array} \tag{6.250}$$

The relation between the angular velocities (p, q, r) in the body frame and the time derivative of the Euler angles (ϕ, θ, ψ) is the following:

$$\underline{\nu} := \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \mathbf{R}_\phi \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \mathbf{R}_\phi \mathbf{R}_\theta \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \tag{6.251}$$

We finally get:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \sin(\phi)\cos\theta \\ 0 & -\sin(\phi) & \cos(\phi)\cos\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{6.252}$$

That is:

$$\underline{\nu} = \mathbf{W}(\underline{\eta})\, \dot{\underline{\eta}} \tag{6.253}$$

where:

$$\underline{\eta} := \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{6.254}$$

and:

$$\mathbf{W}(\underline{\eta}) = \begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \sin(\phi)\cos(\theta) \\ 0 & -\sin(\phi) & \cos(\phi)\cos(\theta) \end{bmatrix} \tag{6.255}$$

It is worth noticing that the preceding relation can be obtained from the following equality which simply states that the time derivative of matrix $\mathbf{R}_b^i(\underline{\eta})$ can be seen as matrix $\mathbf{\Omega}(\underline{\nu})$ of the angular velocities in the body frame expressed in the inertial frame:

$$\frac{d}{dt}\mathbf{R}_b^i(\underline{\eta}) = \mathbf{R}_b^i(\underline{\eta})\, \mathbf{\Omega}(\underline{\nu}) \text{ where } \mathbf{\Omega}(\underline{\nu}) = -\mathbf{\Omega}(\underline{\nu})^T = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}] \tag{6.256}$$

Conversely we have:

$$\dot{\underline{\eta}} = \mathbf{W}(\underline{\eta})^{-1}\, \underline{\nu} \tag{6.257}$$

where:

$$\mathbf{W}(\underline{\eta})^{-1} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix} \tag{6.258}$$

6.12.4 Roll and pitch angles estimation from accelerometer measurements

Let $\underline{g}$ be the gravitational acceleration, $\underline{a}^i$ the acceleration in the inertial frame, $\underline{a}^b$ the acceleration in the body frame and $\mathbf{R}_i^b$ the rotation matrix from the inertial frame to the body frame. The accelerometer provides the following measurement, called the specific acceleration:

$$\underline{a}_m = \mathbf{R}_i^b\ (\underline{a}^i - \underline{g}) \tag{6.259}$$

Denoting $\underline{v}^i$ the velocity in the inertial frame and $\underline{v}^b$ the velocity in the body frame, we have the following relation where $\mathbf{R}_b^i$ the rotation matrix from the body frame to the inertial frame:

$$\underline{v}^i = \mathbf{R}_b^i \, \underline{v}^b \tag{6.260}$$

Thus, after derivation:

$$\underline{a}^i := \frac{d}{dt}\underline{v}^i = \mathbf{R}_b^i \, \underline{\dot{v}}^b + \dot{\mathbf{R}}_b^i \, \underline{v}^b \tag{6.261}$$

Thus the specific acceleration $\underline{a}_m$ in (6.259) reads:

$$\begin{aligned} \underline{a}_m &= \mathbf{R}_i^b \left(\mathbf{R}_b^i \, \underline{\dot{v}}^b + \dot{\mathbf{R}}_b^i \, \underline{v}^b \right) - \mathbf{R}_i^b \, \underline{g} \\ &= \underline{\dot{v}}^b + \mathbf{R}_i^b \dot{\mathbf{R}}_b^i \, \underline{v}^b - \mathbf{R}_i^b \, \underline{g} \end{aligned} \tag{6.262}$$

Once the computation achieved, we get the following expression for the measurements provided by a 3-axis accelerometer[8]:

$$\underline{a}_m = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} 0 & w & -v \\ -w & 0 & u \\ v & -u & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} - g \begin{bmatrix} -\sin(\theta) \\ \cos(\theta)\,\sin(\phi) \\ \cos(\theta)\,\cos(\phi) \end{bmatrix} \tag{6.263}$$

The last term of equation (6.263) can be used to approximate roll angle ϕ and pitch angle θ as follows:

$$\boxed{\left\{ \begin{array}{l} \phi \approx \arctan\left(\frac{a_y}{a_z}\right) \\ \theta \approx \arctan\left(\frac{a_x}{\sqrt{a_y^2 + a_z^2}}\right) \end{array} \right.} \tag{6.264}$$

Note that if the Inertial Measurement Unit (IMU) is not situated at the center of mass, then the accelerometers coordinates $(l_x, \, l_y, \, l_z)$ along each axis in the body frame with its origin at the center of gravity shall be taken into account and the measurements (6.263) provided by a 3-axis accelerometer becomes[8]:

$$\begin{aligned} \underline{a}_m = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} &= \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} 0 & w & -v \\ -w & 0 & u \\ v & -u & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} - g \begin{bmatrix} -\sin(\theta) \\ \cos(\theta)\,\sin(\phi) \\ \cos(\theta)\,\cos(\phi) \end{bmatrix} \\ &+ \begin{bmatrix} -r^2 - q^2 & p\,q - r^2 & p\,r + \dot{q} \\ p\,q + \dot{r} & -p^2 - r^2 & r\,q - \dot{p} \\ p\,r - \dot{q} & r\,q + \dot{p} & -q^2 - p^2 \end{bmatrix} \begin{bmatrix} l_x \\ l_y \\ l_z \end{bmatrix} \end{aligned} \tag{6.265}$$

[8] Marian J. Blachuta and Rafal T. Grygiel and Roman Czyba and Grzegorz Szafranski, Attitude and heading reference system based on 3D complementary filter, 2014 19th International Conference on Methods and Models in Automation and Robotics (MMAR)

6.12.5 Yaw angle estimation from magnetometer measurements

Let $\begin{bmatrix} m_x^i & 0 & m_z^i \end{bmatrix}^T$ be the geomagnetic field measurements in the inertial frame and $\begin{bmatrix} m_x & m_y & m_z \end{bmatrix}^T$ be the geomagnetic field measurements in the body frame. Those those vectors are related as follows:

$$\begin{aligned}
\begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} &= \mathbf{R}_i^b \begin{bmatrix} m_x^i \\ 0 \\ m_z^i \end{bmatrix} \\
&= \begin{bmatrix} m_x^i \cos(\psi) \cos(\theta) - m_z^i \sin(\theta) \\ m_z^i \cos(\theta) \sin(\phi) - m_x^i (\cos(\phi) \sin(\psi) - \cos(\psi) \sin(\phi) \sin(\theta)) \\ m_x^i (\sin(\phi) \sin(\psi) + \cos(\phi) \cos(\psi) \sin(\theta)) + m_z^i \cos(\phi) \cos(\theta) \end{bmatrix}
\end{aligned} \tag{6.266}$$

Then it is worth noticing that the following relations hold:

$$\begin{cases} \sin(\phi)\; m_z - \cos(\phi)\; m_y = m_x^i \sin(\psi) \\ \cos(\theta)\; m_x + \sin(\phi) \sin(\theta)\; m_y + \cos(\phi) \sin(\theta)\; m_z = m_x^i \cos(\psi) \end{cases} \tag{6.267}$$

Those two equations can be used to approximate yaw angle ψ as follows:

$$\boxed{\psi = \arctan\left(\frac{\sin(\phi)\; m_z - \cos(\phi)\; m_y}{\cos(\theta)\; m_x + \sin(\phi) \sin(\theta)\; m_y + \cos(\phi) \sin(\theta)\; m_z} \right)} \tag{6.268}$$

6.12.6 Angular velocity from gyroscope measurements

Gyroscope provides the roll, pitch and yaw rate, p, q and r, respectively, with respect to its body axis system. The relationship between rate gyro output and angular velocity of the Euler angles is similar to (6.258). Nevertheless, because the actual values of the Euler angles ϕ, θ and ψ is not known, it is their estimated values $\widehat{\phi}$, $\widehat{\theta}$ and $\widehat{\psi}$ which is used in matrix $\mathbf{W}(\underline{\widehat{\eta}})$:

$$\boxed{\frac{d}{dt} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \approx \mathbf{W}(\underline{\widehat{\eta}})^{-1} \underline{\nu} = \begin{bmatrix} 1 & \sin(\widehat{\phi}) \tan(\widehat{\theta}) & \cos(\widehat{\phi}) \tan(\widehat{\theta}) \\ 0 & \cos(\widehat{\phi}) & -\sin(\widehat{\phi}) \\ 0 & \frac{\sin(\widehat{\phi})}{\cos(\widehat{\theta})} & \frac{\cos(\widehat{\phi})}{\cos(\widehat{\theta})} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}} \tag{6.269}$$

6.12.7 Attitude and Heading Reference System (AHRS) based on complementary filter

The purpose of the Attitude and Heading Reference System (AHRS) is to compute the best estimate of the Euler angles ϕ, θ and ψ from the roll and pitch estimates (6.264) provided by the accelerometers, from the yaw estimate (6.268) provided by the magnetometer and from the angular velocities (6.269) provided by the gyroscopes.

Because each channel is decoupled, the best estimate can be achieved by 3 independent complementary filters of the form (6.241) for continuous time estimate, or (6.247) for discrete time estimate. In those equations, y_1 stands for the measurements provided by the accelerometers or the magnetometer, and y_2 stands for the measurements provided by the gyroscopes.

Introduction to Optimal Control

Exercise

Aircraft pitch control

Exercise 1

The short-period longitudinal dynamics for a medium-sized jet with center of gravity unusually far aft might be described by the following state equation, where α is the angle of attack, q the pitch rate and u the elevator deflection [1]:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} -1.417 & 1 \\ 2.86 & -1.183 \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ -3.157 \end{bmatrix} u$$

1. Compute the open loop poles of the *natural* aircraft

   To stabilize the plant and keep the pitch rate small, the following performance index is chosen:

$$J(u(t)) = \frac{1}{2} \int_0^{\infty} q^2(t) + 5u^2(t)\, dt$$

2. What are the values of matrices R and Q ? Check that $Q = Q^T \geq 0$ and $R = R^T > 0$
3. What is the value of matrix N which is defined by $Q = N^T N$
4. Check that (A, B) is controllable and that (A, N) is observable
5. Compute the positive solution of the algebraic Ricatti equation P thanks to the eigenvectors of the Hamiltonian matrix as well the optimal gain K. Check that you get the same result either with scilab (command 'ricc') or matlab (command 'care')
6. Plot the Bode diagram of $K(s\mathbb{I} - A)^{-1}B$ ('open loop') and compute the phase margin
7. Plot the Nyquist diagram of $K\left(s\mathbb{I} - A\right)^{-1} B$ ('open loop')

Exercise 2

We use the same aircraft model as in the previous exercise but we now wish to stabilize the plant and keep the pitch rate small. Thus the following performance index is chosen:

$$J(u(t)) = \frac{1}{2} \int_0^{\infty} q^2(t) + Ru^2(t)\, dt$$

1. Taking $R = 1000$ (control is weighted very heavily), compute the value of P, solution of the algebraic Ricatti equation, as well as the optimal gain K

[1] Lewis F., Vrabie D. and Syrmos L., Optimal Control, John Wiley & Sons (2012) (p 166)

2. Let $G(s) = N\left(s\mathbb{I} - A\right)^{-1} B = \frac{n(s)}{d(s)}$ be the open loop transfer function.

   Let $d(s) = \det\left(s\mathbb{I} - A\right)$ be the open loop characteristics polynomial.

   Let be *n(s)* the numerator of $G(s) = N\left(s\mathbb{I} - A\right)^{-1} B$.

   And let $\beta(s) = \det\left(s\mathbb{I} - A + BK\right)$ be the closed loop characteristic polynomial.

   Check that:

$$\beta(s)\beta(-s) = d(s)d(-s) + \frac{1}{R}n(s)n(-s)$$

3. Compute the poles of the closed loop system, which are the roots of $\det\left(s\mathbb{I} - A + BK\right)$, and compare then with the poles of the open loop system, which are the roots of $\det\left(s\mathbb{I} - A\right)$. What do you conclude on poles placement when $R \to \infty$?

4. Plot the root locus of $G(s)G(-s) = \frac{n(s)n(-s)}{d(s)d(-s)}$. Using the datatips facilities choose a value for the loop gain k, set $R = 1/k$ and check the values of the poles of the closed loop system

5. Taking $R = 0.01$ (the control is cheap), compute the value of P, solution of the algebraic Ricatti equation, as well as the optimal gain K

6. Compute the poles of the closed loop system and compare then with the zeros of the open loop system $G(s)$. What do you conclude on poles placement when $R \to 0$?

7. Compare $|K|$ and $\frac{|N|}{\sqrt{R}}$. What do you conclude on the gain value when $R \to 0$?

8. Assuming that $u = q_c - Kx$ where q_c is the commanded pitch rate, compute the transfer function between q and q_c of the closed loop plant

9. Plot the step response of the closed loop system. What is the value of the steady state error?

10. Design an LQ tracker which enables to track the commanded pitch rate q_c

Matlab code

An example of the Matlab code for both exercises is the following:

```
%Exercise 1
%----------
clear all
A = [ -1.417 , 1 ; 2.86 , -1.183]
B = [0 ; -3.157]

%1. Compute the open loop poles of the natural aircraft
%-----------------------
lambda = eig(A)

%2. What are the values of matrices R and Q ?
%Check that Q = Q T  0 and R = R T > 0
%-----------------------
R=5 %R = R^T  > 0
Q = [0, 0; 0, 1] %Q = Q^T  >= 0
```

```
%3. What is the value of matrix N which is defined by
%Q = N^T N
%-----------------------
N = [0 1]
Q = N'*N

%4. Check that (A, B) is controllable and that (A, N ) is observable
%-----------------------
%We may use the Kalman test to check the controllability of the system
rank(ctrb(A,B)) %rank([B, A*B])

%we may use the Kalman test to check the observability of the system
rank(obsv(A,N)) %rank([N; N*A])

%5.Compute the positive solution of the algebraic Ricatti
%equation P thanks to the eigenvectors of the Hamiltonian
%matrix as well the optimal gain K.
%Check that you get the same result either with scilab (command 'ricc')
%or matlab (command 'care')
%-----------------------
%Hamiltonian matrix
H = [A, -B*inv(R)*B'; -Q, -A']
[V,D] = eig(H)
%The eigenvalues in the matrix D are symmetric with respect
%to the imaginary axis
%The eigenvectors assiciated with the eigenvalues in the
%left half plane are the following:
X12=V(:,1:2)
X1=X12(1:2,:)
X2=X12(3:4,:)
P12=X2*inv(X1)

%P: solution of the algebraic Riccati equation
%L are the closed-loop eigenvalues
%K is the feebback gain matrix
[P,lambdaFTBF,K] = care(A,B,Q,R)
%P is the same than P12
%K is the same than B'*inv(R)*P
K
B'*inv(R)*P

%lambdaFTBF are the eigenvalues of A-B*K
%They are the same than the eigenvalues of the Hamiltonian
%matrix H in the left half plane
D
lambdaFTBF
eig(A-B*K)

%6.Plot the Bode diagram of K(sI  A)^{-1} B ('open loop')
%and compute the phase margin
%-----------------------
%dot x = A x + B u; y = N x => G(s) = N (sI-A)^-1 B
%To obtain the Bode plot of K (sI-A)^-1 B
```

```
%we just change N by K
KPhiB = ss(A,B,K,0);
%[num,den] = ss2tf(A,B,N,0,1)
figure; bode(KPhiB); grid;
[Gm,Pm,Wcg,Wcp] = margin(KPhiB)
Gm_dB = 20*log10(Gm)

%7. Plot the Nyquist diagram of K (sI  A)^{-1} B ('open loop')
%-----------------------
figure; nyquist(KPhiB);

%----------
%Exercise 2
%----------
%1. Taking R = 1000 (control is weighted very heavily),
%compute the value of P, solution of the algebraic Ricatti
%equation, as well as the optimal gain K
%-----------------------
R=100
[P,lambdaFTBF,K] = care(A,B,Q,R)

%2. Check that beta(s) beta(-s) = d(s) d(-s) + 1/R n(s) n(-s)
%-----------------------
[ns,ds] = ss2tf(A,B,N,0,1)
[nms,dms] = ss2tf(-A,B,-N,0,1)

[betans,betads] = ss2tf(A-B*K,B,N,0,1)
[betanms,betadms] = ss2tf(-(A-B*K),B,-N,0,1)

conv(betads,betadms)
conv(ds,dms) + 1/R*conv(ns,nms)

%3. Compute the poles of the closed loop system,
%which are the roots of det (sI  A + BK ) ,
%and compare then with the poles of the open loop system,
%which are the roots of det (sI  A).
%What do you conclude on poles placement when R   ?
%-----------------------
r1 = roots(ds)
lambdaFTBF


%4. Plot the root locus of  $G(s)G(-s)=\frac{n(s)n(-s)}{d(s)d(-s)}$.
%Using the datatips facilities  choose a value for the loop gain $k$,
%set $R = 1 / k$ and check the values of the poles of the closed loop system
%-----------------------
GsGms = tf(conv(ns,nms),conv(ds,dms))
figure; rlocus(GsGms)

k = 5
R = 1/k
[P,lambdaFTBF,K] = care(A,B,Q,R)

%5. Taking $R = 0.01$ (the control is cheap), compute the value of $P$,
```

```
%solution of the algebraic Ricatti equation, as well as the optimal gain $K$
%-----------------------
R = 0.01
[P,lambdaFTBF,K] = care(A,B,Q,R)

%6. Compute the poles of the closed loop system and compare then with
%the zeros of the open loop system $G(s)$.
%What do you conclude on poles placement when $R \to 0$ ?
%-----------------------
Gs = tf(ns,ds)
roots(Gs.num{1})

%7. Compare $|K|$ and $\frac{|N|}{\sqrt{R}}$. What do you conclude on the
%gain value when $R \to 0$ ?
%-----------------------
K
N/sqrt(R)

%8. Assuming that  $u=q_c - K x$ where $q_c$ is the commanded pitch rate,
%compute the transfer function between $q$ and  $q_c$  of the closed loop plant
%-----------------------
[ncl,dcl] = ss2tf(A-B*K,B,N,0,1)

%9. Plot the step response of the closed loop system. What is the value of
%the steady state error?
figure;
step(tf(ncl,dcl));
grid on;
ess = ncl(3) / dcl(3) %steady state error

%10. Design an LQ tracker which enables to track the commanded pitch rate
%$q_c$
L = 10;
Aa = [A, zeros(2,1); -L*N, 0]
Ba = [B;0]

%We check that (Aa,Ba) is controllable and that (A,C) is observable
%We may use the Kalman test to check the controllability of the system
rank(ctrb(Aa,Ba))

%We check that (Aa,Ba) is observable with Kalman test
Na = [N, 1];
rank(obsv(Aa,Na))

Qa = Na'*Na
[Pa,lambdaFTBFa,Ka] = care(Aa,Ba,Qa,R)

% [ncl,dcl] = ss2tf(A-B*K,B,C,0,1) %tf without integrator
[ncla,dcla] = ss2tf(Aa-Ba*Ka,[zeros(2,1);L],[N, 0],0,1) %tf with integrator
figure;
step(tf(ncla,dcla));
grid on;
```

Part 2 - Robust control

ONERA
THE FRENCH AEROSPACE LAB

ENAC

isae
Institut Supérieur de l'Aéronautique et de l'Espace

Introduction to robust $\mathcal{H}_\infty$ control

Clément Roos

Information Processing and Systems Department

ONERA Toulouse

with contributions from Jean-Marc Biannic and Daniel Alazard

Introduction

Over the past thirty years, thanks to the fast development of increasingly powerful computers, **robust control theory** has become very successful. Today, this field is **one of the most important and mature** in the field of Automatic Control.

In short, robust control methods aim at **using simplified linear models to design controllers that still work on real plants**, which are most often too complicated to be accurately described by a set of linear differential equations.

Objectives of this course

- learn the basics in robust $\mathcal{H}_\infty$ control
- learn how to use robust control design tools in Matlab/Simulink

Outline

1 Preliminaries
- Definition of robustness
- Why robustness?
- Robustness of closed-loop systems

2 A classical approach to robust stability

3 Towards a modern approach

4 Extension to robust performance

5 The $\mathcal{H}_\infty$ control problem

Definition of robustness

In the general field of Automatic Control, and more precisely in Control Theory, the notion of robustness **quantifies the sensitivity of a controlled system with respect to internal or external disturbing phenomena**.

Examples of internal disturbances

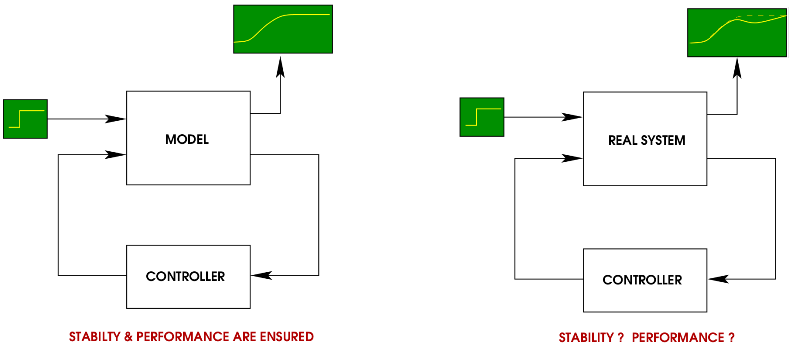
- badly known phenomena $\Rightarrow$ uncertain or time-varying parameters
- inherently complex and/or nonlinear phenomena $\Rightarrow$ modelling errors
- digital implementation $\Rightarrow$ delay
- limited actuator capacity $\Rightarrow$ saturations
- limited sensor bandwidth and accuracy $\Rightarrow$ neglected dynamics, noise, bias

Examples of external disturbances in aerospace applications

- windshear
- atmospheric turbulences
- solar pressure

Why robustness?

There is usually a **significant gap** between the model used to design the controller and the real system on which it will be implemented.



Objective of robust control

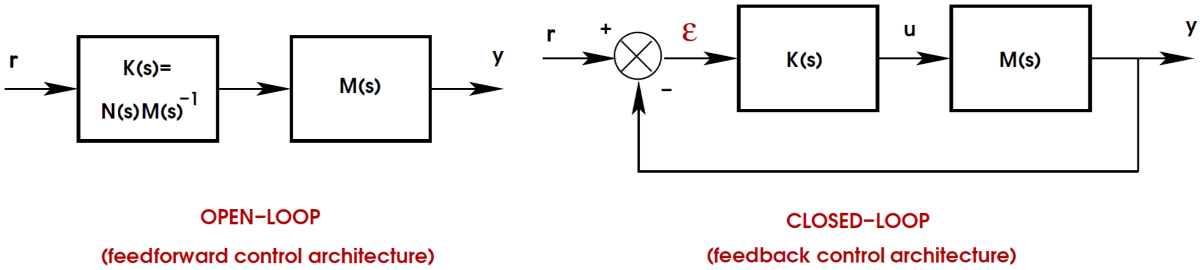
Take robustness issues into account **from the design stage**, so as to speed-up controller validation and facilitate its implementation on the real system.

Robustness properties of closed-loop systems

Open-loop control techniques can be used if:

- a very accurate model of the plant is available
- the plant is invertible
- external pertubations are negligible

The above conditions are rarely fullfilled. This is why it is highly preferable to use a feedback control structure, which offers nice robustness properties. . .



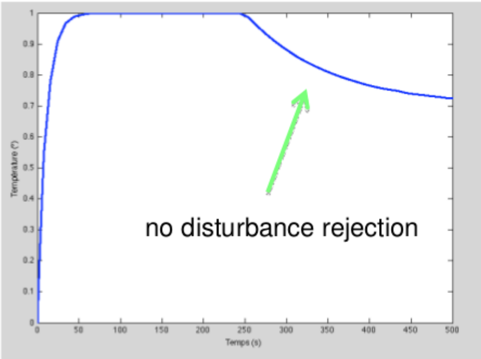
Robustness properties of closed-loop systems

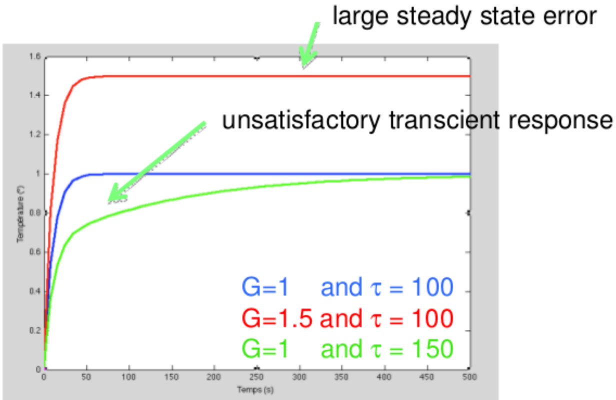
Example: $M(s) = \frac{G}{1+\tau s}$ where $G = 1$ and $\tau = 100\,s$.

Desired behavior: first-order system with a time constant $\tau_d = 10\,s$.

First case: open-loop control

A suitable controller is $K(s) = \frac{1+\tau s}{1+\tau_d s}$, which leads to $\frac{y}{r} = \frac{G}{1+\tau_d s}$.

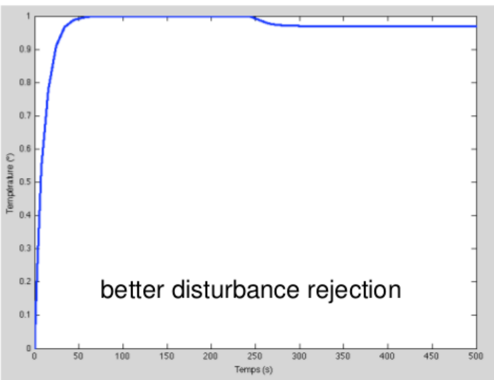


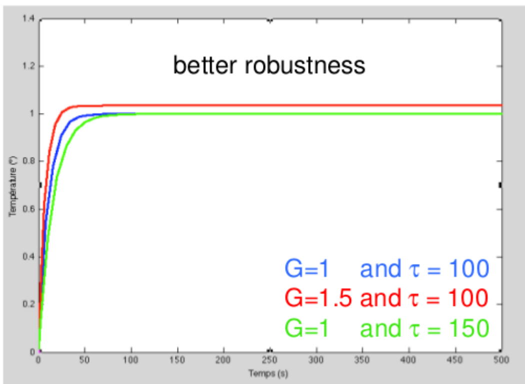


Robustness properties of closed-loop systems

Second case: closed-loop control

A suitable controller is $K = 9$ and $H = \frac{10}{9}$, which leads to $\frac{y}{r} = \frac{10G}{1+9G+\tau s}$.

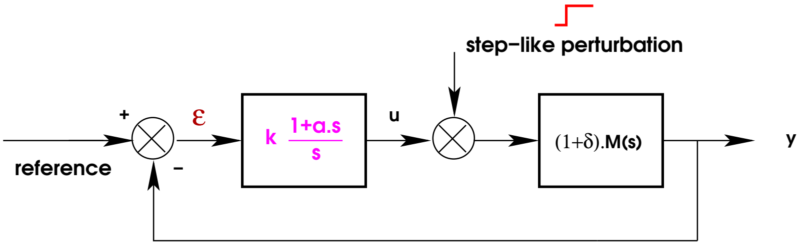




Closed-loop control is inherently more robust than open-loop control.

Illustration: robustness of PI feedback controllers

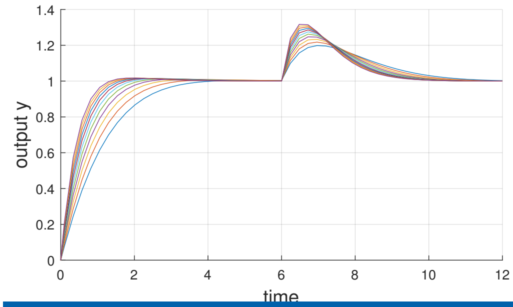
The static behavior of a PI controlled system is neither affected by external step-like perturbations nor by gain uncertainties on the nominal plant.



P.I. Controlled system

Simulations: step reference at $t = 0\,s$ and step perturbation at $t = 6\,s$ for $\delta \in [-0.5\ 0.5]$

How to tune the controller gains to maximize robustness and performance?



Outline

1 Preliminaries

2 A classical approach to robust stability
- Standard robustness margins
- Limitations
- The modulus margin

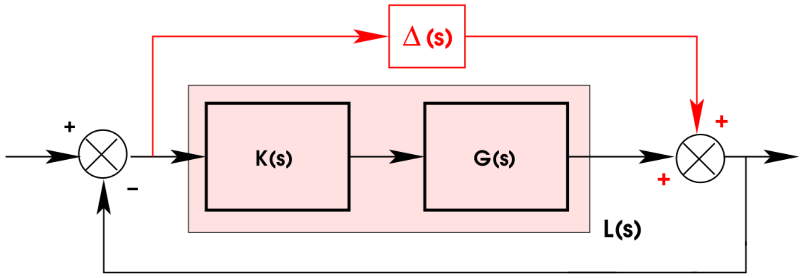
3 Towards a modern approach

4 Extension to robust performance

5 The $\mathcal{H}_\infty$ control problem

Standard robustness margins

Robustness has always been a central issue in control. Because of computers limitations, very simple notions were introduced for SISO systems and are still widely used today: **the gain and phase margins**.

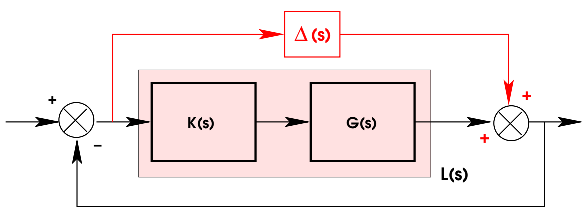


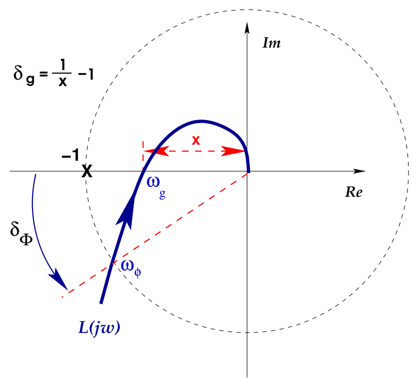
The closed-loop transfer function is $\frac{L(s)}{1+L(s)}$. If the system is stable, then $L(j\omega) \neq -1$ for all $\omega \geq 0$. But at each frequency, the smallest additive uncertainty $\Delta^*(j\omega)$ such that $L(j\omega) + \Delta^*(j\omega) = -1$ can be computed.

Standard robustness margins

Two particular frequencies are usually considered in practice:

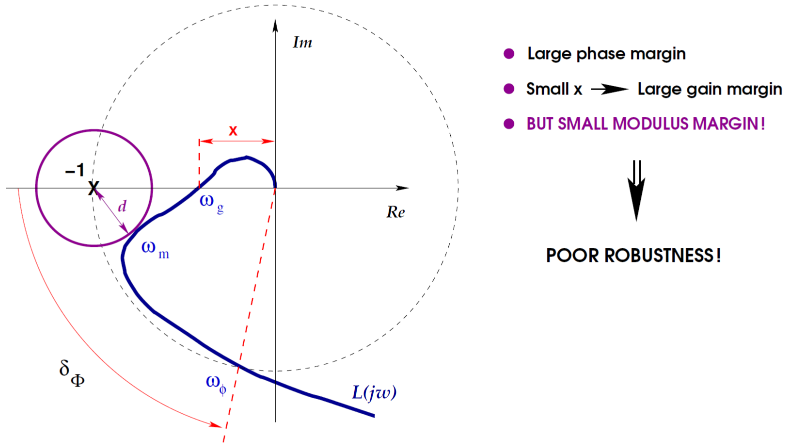
- **gain margin:** when $\omega = \omega_g$, Δ_g^* acts only on the magnitude of L
$$L(j\omega_g) + \Delta_g^* = (1+\delta_g)L(j\omega_g) = -1$$
- **phase margin:** when $\omega = \omega_\phi$, Δ_ϕ^* acts only on the phase of L
$$L(j\omega_\phi) + \Delta_\phi^* = L(j\omega_\phi)e^{-j\delta_\phi} = -1$$





Limitation: illustration in the Nyquist plane

Despite large gain and phase margins δ_g and δ_ϕ, the Nyquist plot may get dangerously close to the critical point $(-1, 0)$ at another frequency ω_m!



$$\begin{cases} L(j\omega_m) + \Delta_m^* = (1 + \delta_{g_m}) L(j\omega_m) e^{-j\delta_{\phi_m}} = -1 \,, \; |\Delta_m^*| = d \\ \delta_{g_m} \ll \delta_g \,, \; \delta_{\phi_m} \ll \delta_\phi \end{cases}$$

The modulus margin

Good gain and phase margins mean that a large variation in either the gain or the phase is necessary to make the system unstable. But a small simultaneous variation of both the gain and the phase can be sufficient...

A new robustness margin should then be introduced. It is defined as the minimal distance d between the Nyquist plot of $L(j\omega)$ and the critical point $(-1, 0)$. This distance is refered to as the **modulus margin**:

$$\begin{aligned} d &= \inf_{\omega \geq 0} |1 + L(j\omega)| \\ &= \inf_{\omega \geq 0} |1 + G(j\omega)K(j\omega)| \end{aligned}$$

Outline

1 Preliminaries

2 A classical approach to robust stability

3 Towards a modern approach
- Extension to MIMO systems
- Singular values
- The $\mathcal{H}_\infty$ norm
- Uncertainties characterization
- The small-gain theorem

4 Extension to robust performance

5 The $\mathcal{H}_\infty$ control problem

Extension to MIMO systems

The "modern" approach to robust control developed at the end of the 1980s is essentially based on a **multivariable extension of the modulus margin**, which is the only stability margin that can be generalized to MIMO systems.

In the SISO case, the distance to singularity can be obtained equivalently by computing the minimum gain of the transfer function $1 + G(s)K(s)$ or by solving at each frequency the equation:

$$d(\omega) = \inf\{|\Delta(\omega)| : 1 + G(j\omega)K(j\omega) + \Delta(\omega) = 0\}$$

In the MIMO case, the singularity of a matrix is evaluated through its determinant, and the condition "simply" becomes:

$$\det(I + G(j\omega)K(j\omega) + \Delta(\omega)) = 0$$

Extension to MIMO systems

The above extension is conceptually simple. However it leads to numerically untractable problems. Basically, the idea of robust control consists of computing a controller $K(s)$ such that:

$$\det(I + G(j\omega)K(j\omega) + \Delta(\omega)) \neq 0 \tag{1}$$

for all uncertainties Δ whose size (to be defined) is bounded by a given (as large as possible) scalar γ.

Two problems appear:

- How can we obtain a numerically tractable condition to enforce (1)?
- How can we simply measure the size of the uncertainty matrix?

The **singular value** notion provides a possibly conservative but elegant answer to both problems.

Singular values

Definitions

A $p \times m$ matrix M has $n = \min(p, m)$ singular values defined as:

$$\sigma_i(M) = \sqrt{\lambda_i(M^*M)}$$

where $\lambda_i(.)$ is the ith eigenvalue and M^* is the conjugate transpose of M.

It gives a **metric information** on the linear transformation defined by M:

- $\overline{\sigma}(M) = \max_{i=1\ldots n} \sigma_i = \max_{u \neq 0} \frac{\|Mu\|}{\|u\|}$
- $\underline{\sigma}(M) = \min_{i=1\ldots n} \sigma_i = \min_{u \neq 0} \frac{\|Mu\|}{\|u\|}$

Singular values can be viewed as **a generalization of the absolute value** of a complex number to complex-valued (possibly non-square) matrices.

Singular values

Singular value decomposition

$$M = V\Sigma U^* = [V_1 \; V_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1^* \\ U_2^* \end{bmatrix}$$

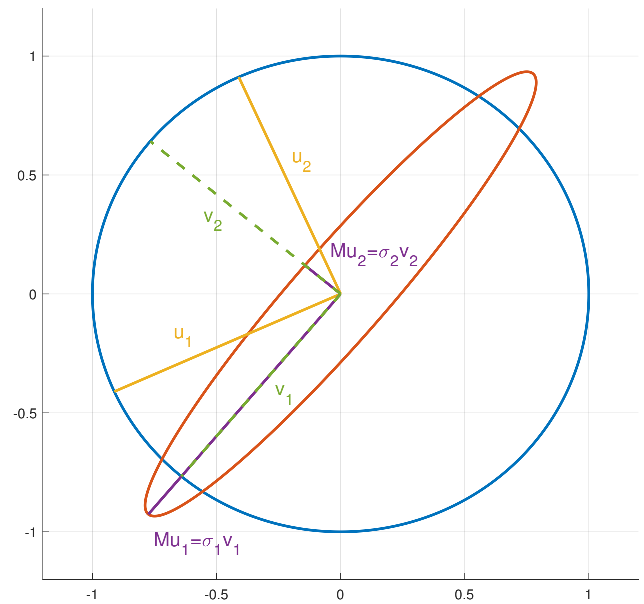
where:

- $r \leq \min(p, m)$ is the rank of M
- $U \in \mathbb{C}^{m \times m}$ is a unitary matrix ($UU^* = I_{m \times m}$)
- $V \in \mathbb{C}^{p \times p}$ is a unitary matrix ($VV^* = I_{p \times p}$)
- $\Sigma_1 =$ diag$(\sigma_i), i = 1, \ldots, r$ with $\sigma_1 > \sigma_2 > \cdots > \sigma_r > 0$

Property: M nonsingular if and only if $\underline{\sigma}(M) > 0$

Singular values

Illustration in the 2-D case



Singular values

Properties

- $\overline{\sigma}(.)$ is a norm and satisfies $\overline{\sigma}(M_1 + M_2) \leq \overline{\sigma}(M_1) + \overline{\sigma}(M_2)$
- $\overline{\sigma}(M_1 M_2) \leq \overline{\sigma}(M_1)\overline{\sigma}(M_2)$
- $\overline{\sigma}(M^{-1}) = \frac{1}{\underline{\sigma}(M)}$ and $\underline{\sigma}(M^{-1}) = \frac{1}{\overline{\sigma}(M)}$
- $\overline{\sigma}(VMU) = \overline{\sigma}(M)$ for all unitary matrices U and V
- M is real and symmetric $\Rightarrow \sigma_i(M) = |\lambda_i(M)|$ for all $i = 1, \ldots, n$
- singularity property:

$$\overline{\sigma}(E) < \underline{\sigma}(M) \Rightarrow \underline{\sigma}(M + E) > 0$$

  $\underline{\sigma}(M)$ is the "size" of the smallest matrix E such that $M + E$ becomes singular, in particular:
  - $\overline{\sigma}(M) < 1 \quad \Rightarrow \quad \det(I + M) \neq 0$
  - $\underline{\sigma}(M) > 1 \quad \Rightarrow \quad \det(I + M) \neq 0$

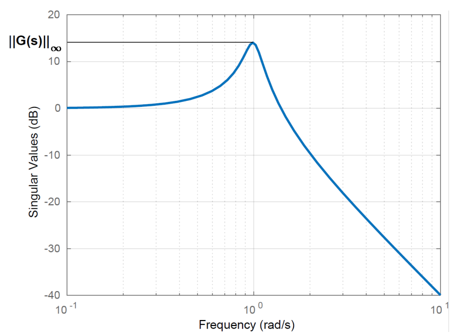
The $\mathcal{H}_\infty$ norm

Frequency-domain definition

$$\|G(s)\|_\infty = \sup_{\omega \geq 0} \overline{\sigma}(G(j\omega)) \text{ if } G(s) \text{ is stable } (\infty \text{ if not})$$

This norm is a measure of the largest magnitude of the system frequency-domain response $G(j\omega)$. In the SISO case, $\|G(s)\|_\infty = \sup_{\omega \geq 0} |G(j\omega)|$ can be obtained easily from the Bode plot.

Example: compute the $\mathcal{H}_\infty$ norm of $G(s) = \dfrac{\omega^2}{s^2 + 2\xi\omega s + \omega^2}$



Practical computation of the $\mathcal{H}_\infty$ norm

- $\|G(s)\|_\infty$ cannot be computed analytically.
- $\|G(s)\|_\infty$ can be approximated on a frequency grid $(\omega_i)_{1\leq i\leq N}$:

$$\|G(s)\|_\infty \approx \max_{\omega_i} \overline{\sigma}(G(j\omega_i))$$

  but the peak value can be missed, *e.g.* in case of a badly damped system!
- $\|G(s)\|_\infty$ can be **computed with the desired accuracy by a dichotomic search** using the following result (Matlab function **hinfnorm**):

Let $G(s) = C(sI - A)^{-1}B + D$ be a stable system. Then $\|G(s)\|_\infty > \overline{\sigma}(D)$, and for all $\gamma > \overline{\sigma}(D)$, the two following properties are equivalent:

(i) $\|G(s)\|_\infty < \gamma$

(ii) the matrix $\begin{bmatrix} A & 0 \\ 0 & -A^T \end{bmatrix} - \begin{bmatrix} 0 & B \\ C^T & 0 \end{bmatrix} \begin{bmatrix} \gamma I & D \\ D^T & \gamma I \end{bmatrix}^{-1} \begin{bmatrix} C & 0 \\ 0 & -B^T \end{bmatrix}$ has no eigenvalue on the imaginary axis

Back to the modulus margin

Property: M is nonsingular $\Leftrightarrow \det(M) \neq 0 \Leftrightarrow \underline{\sigma}(M) > 0$.

The condition $\det(I + G(j\omega)K(j\omega) + \Delta(\omega)) \neq 0$ is thus equivalent to $\underline{\sigma}(I+G(j\omega)K(j\omega)+\Delta(\omega)) > 0$, and the modulus margin can be generalized to MIMO systems using singular values:

$$d = \inf_{\omega\geq 0} \underline{\sigma}\left(I + G(j\omega)K(j\omega)\right) = \frac{1}{\sup\limits_{\omega\geq 0} \overline{\sigma}\left((I + G(j\omega)K(j\omega))^{-1}\right)}$$

$\mathcal{H}_\infty$ formulation of the modulus margin

- The modulus margin d is equal to $\|S(s)\|_\infty^{-1}$, where $S(s) = (I + G(s)K(s))^{-1}$ is called the **output sensitivity function**.
- Minimizing the $\mathcal{H}_\infty$ norm of $S(s)$ is equivalent to maximizing the modulus margin, thus **improving robustness to uncertainties** $\Delta(s)$.

Uncertainties characterization

Two main kinds of uncertainties can affect a system:

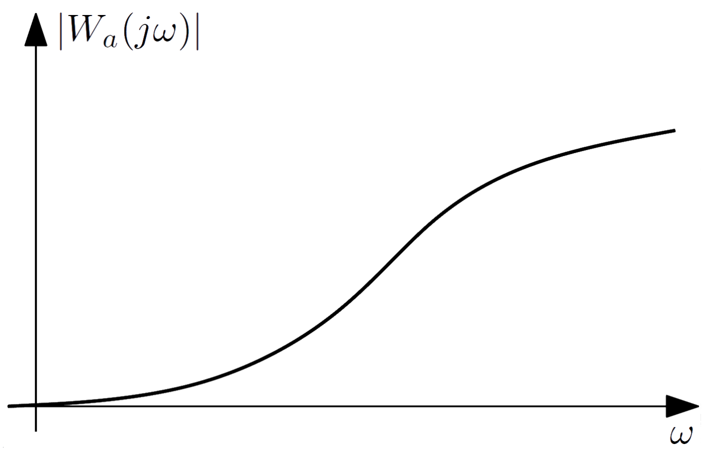
- **frequency-dependent unstructured uncertainties**
  - $\rightarrow$ example: neglected high-frequency modes and actuator/sensor dynamics
  - $\rightarrow$ rough description, but $\mathcal{H}_\infty$ control can handle different configurations:
    - additive uncertainties: $G(s) = G_0(s) + \Delta(s)$
    - input multiplicative uncertainties: $G(s) = G_0(s)(1 + \Delta(s))$
    - output multiplicative uncertainties: $G(s) = (1 + \Delta(s))G_0(s)$
    - feedback uncertainties: $K(s) = K_0(s) + \Delta(s)$
  - $\rightarrow$ additive uncertainties on $G(s)K(s)$ for modulus margin computation
- **constant parametric uncertainties**
  - $\rightarrow$ example: mass, inertia, stability derivatives of an aircraft: $p = (1 + \delta)p_0$
  - $\rightarrow$ very accurate description, but $\mathcal{H}_\infty$ control cannot take them into account
  - $\rightarrow$ use more advanced control techniques (μ-synthesis) or analyze them during post-validation using dedicated tools (**μ-analysis**)

Focus on unstructured uncertainties

It is assumed that an **upper bound** on $\Delta(j\omega)$ exits for each frequency ω:

$$\overline{\sigma}(\Delta(j\omega)) \leq |W_a(j\omega)| \quad \forall \omega \geq 0$$

which defines **the set of admissible uncertainties**.



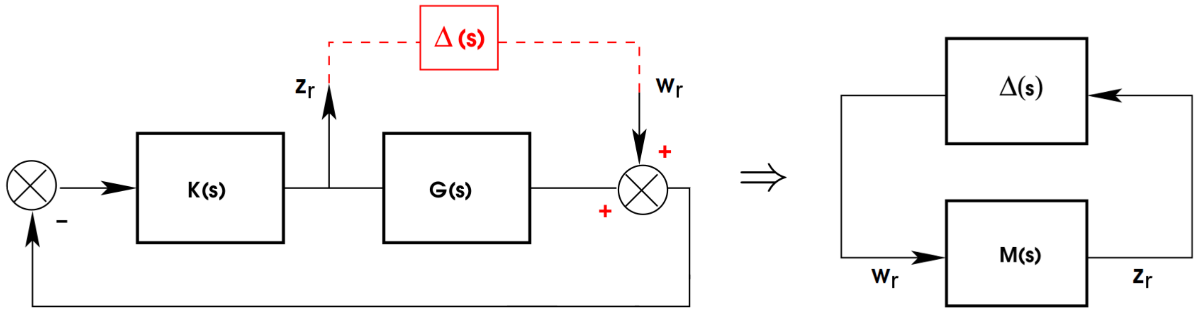
Typical profile for $|W_a(j\omega)|$

Standard interconnection

The uncertain closed-loop system is rewritten as a feedback interconnection between:

- the stable nominal closed-loop system $M(s)$
- the stable unstructured uncertainty $\Delta(s)$

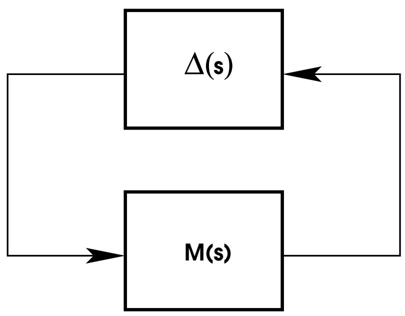
Stability is considered, so all exogenous inputs and outputs are removed.



For an additive uncertainty, $M(s) = -K(s)(I+G(s)K(s))^{-1} = -K(s)S(s)$.

The small gain theorem

Consider the following interconnection:



where $M(s)$ and $\Delta(s)$ are stable linear systems of compatible dimensions.

The interconnection is stable for all $\Delta(s)$ such that $\|\Delta(s)\|_\infty \leq 1$ **if and only if** $\|M(s)\|_\infty < 1$.

The small gain theorem

If the uncertainty $\Delta(s)$ is bounded by $W_a(s)$ instead of 1, the previous result can be slightly adapted:

The interconnection is stable for all $\Delta(s)$ such that $\|\Delta(s)/W_a(s)\|_\infty \leq 1$ **if and only if** $\|W_a(s)M(s)\|_\infty < 1$.

$\|W_a(s)M(s)\|_\infty < 1$ is a **hard constraint** (it is satisfied or not). It is usually more flexible to consider a **soft contraint** instead:

The interconnection is stable for all $\Delta(s)$ such that $\|\Delta(s)/W_a(s)\|_\infty \leq 1/\gamma$ **if and only if** $\|W_a(s)M(s)\|_\infty < \gamma$.

Objective: find the minimum value of γ such that $\|W_a(s)M(s)\|_\infty < \gamma$:

- if $\gamma < 1 \rightarrow$ stability is proved for a larger uncertainty set than required
- if $\gamma > 1 \rightarrow$ stability is proved for a subset of the admissible uncertainties

Robust stability conditions

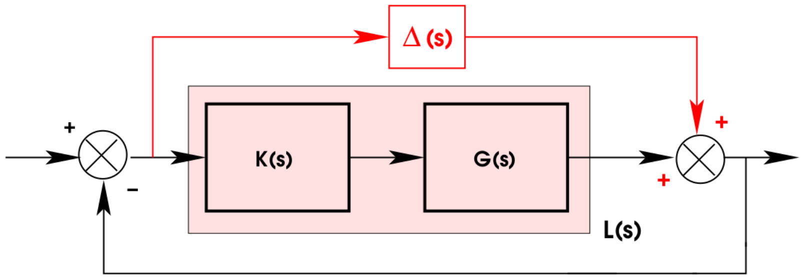
Type of uncertainty	Uncertainty upper bound	Robust stability iff
$G + \Delta$ (additive)	$\overline{\sigma}(\Delta(j\omega)) \leq \lvert W_a(j\omega)\rvert$	$\|W_a\ K(I+GK)^{-1}\|_\infty < 1$
$G(I+\Delta)$ (input mult.)	$\overline{\sigma}(\Delta(j\omega)) \leq \lvert W_{mi}(j\omega)\rvert$	$\|W_{mi}\ KG(I+KG)^{-1}\|_\infty < 1$
$(I+\Delta)G$ (output mult.)	$\overline{\sigma}(\Delta(j\omega)) \leq \lvert W_{mo}(j\omega)\rvert$	$\|W_{mo}\ GK(I+GK)^{-1}\|_\infty < 1$
$K + \Delta$ (feedback)	$\overline{\sigma}(\Delta(j\omega)) \leq \lvert W_k(j\omega)\rvert$	$\|W_k\ G(I+KG)^{-1}\|_\infty < 1$

Remark: The Laplace variable is omitted in the table to simplify the notations.

- $W_a(s), W_{mi}(s), W_{mo}(s), W_k(s)$ are usually **high-pass** SISO transfer functions
- robust stability requires **small gains in the frequency range where the uncertainty is big** (usually at high frequency)

Back again to the modulus margin

The interconnection used to compute the modulus margin corresponds to an additive uncertainty $\Delta(s)$ on $L(s) = G(s)K(s)$:



The transfer $M(s)$ seen by $\Delta(s)$ is $M(s) = -(I + G(s)K(s))^{-1} = -S(s)$.

According to the small gain theorem, the interconnection is stable for all $\Delta(s)$ such that $\|\Delta(s)\|_\infty \leq 1/\gamma$ iff $\|S(s)\|_\infty < \gamma$. As already emphasized, minimizing the $\mathcal{H}_\infty$ norm of $S(s)$ improves the modulus margin.

Outline

1 Preliminaries

2 A classical approach to robust stability

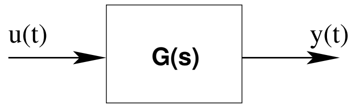
3 Towards a modern approach

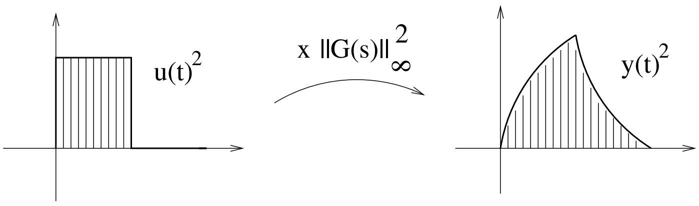
4 Extension to robust performance
- Time-domain interpretation of the $\mathcal{H}_\infty$ norm
- Performance requirements in the $\mathcal{H}_\infty$ framework

5 The $\mathcal{H}_\infty$ control problem

Time-domain interpretation of the $\mathcal{H}_\infty$ norm

The $\mathcal{H}_\infty$ norm has an interesting time-domain interpretation which makes it useful to handle performance problems as well.

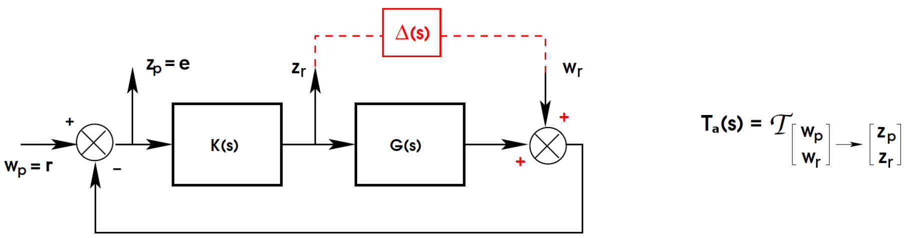




$$\|G(s)\|_\infty^2 = \sup_{\substack{u \in \mathcal{L}_2 \\ u \neq 0}} \frac{\int_0^\infty y(t)^2 dt}{\int_0^\infty u(t)^2 dt}$$

Time-domain interpretation of the $\mathcal{H}_\infty$ norm

Consider the following diagram:



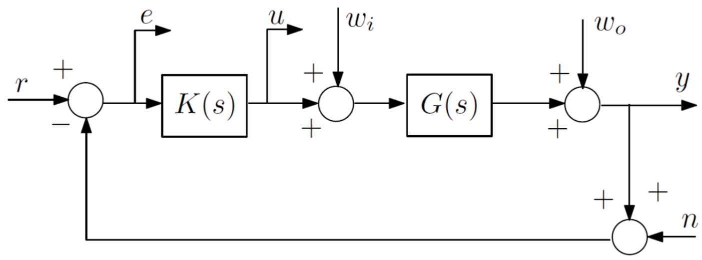
which defines an augmented transfer function $T_a(s)$. If $\|T_a(s)\|_\infty < \gamma$, then from the small-gain theorem and using the time-domain interpretation of the $\mathcal{H}_\infty$ norm, it is readily checked that:

$$\underbrace{\|\Delta(s)\|_\infty < 1/\gamma}_{\text{stability is ensured}} \Rightarrow \int_0^\infty z_p^2(t)\,dt < \gamma^2 \int_0^\infty w_p^2(t)\,dt$$

Minimizing $\|T_a(s)\|_\infty$ amounts to minimizing the "size" of the transfer between w_p and z_p, *i.e.* the error signal e.

Performance requirements in the $\mathcal{H}_\infty$ framework

Several classical performance requirements can be expressed as an $\mathcal{H}_\infty$ constraint, *i.e.* a weighted transfer function whose $\mathcal{H}_\infty$ norm must either be less than 1 (hard constraint) or be minimized (soft constraint).



$$
\begin{aligned}
Y &= GK(I+GK)^{-1}(R-N) + (I+GK)^{-1}W_o + G(I+KG)^{-1}\,W_i \\
E &= (I+GK)^{-1}(R-W_o-N) - G(I+KG)^{-1}\,W_i \\
U &= K(I+GK)^{-1}(R-W_o-N) - KG(I+KG)^{-1}W_i
\end{aligned}
$$

- $S(s) = (I+G(s)K(s))^{-1}$: **output sensitivity function**
- $T(s) = G(s)K(s)(I+G(s)K(s))^{-1}$: **complementary output sensitivity function**

Performance requirements in the $\mathcal{H}_\infty$ framework

- **servo-loop performance:** the transfer between the reference r and the error e must be "small" mainly at low frequency to have a low steady-state error:

$$\overline{\sigma}(S(j\omega)) < |S_{desired}(j\omega)| \quad \forall\, \omega \geq 0$$

$$\Rightarrow \|W_1 S\|_\infty < 1 \text{ where } W_1(s) = \frac{1}{S_{desired}(s)} \text{ has a **low-pass** response}$$

$\rightarrow$ also **improves the modulus margin** (slide 31)

- **high frequency attenuation of the control signal (actuator health):** the transfer between the reference r and the control signal u must be "small" mainly at high frequency:

$$\Rightarrow \|W_2 KS\|_\infty < 1 \text{ where } W_2(s) = \frac{1}{KS_{desired}(s)} \text{ has a **high-pass** response}$$

$\rightarrow$ also **improves robustness to an additive uncertainty** (slide 30)

Performance requirements in the $\mathcal{H}_\infty$ framework

- **noise attenuation on the output:** the transfer between n and y must be "small" mainly at high frequency (assuming that n is a high frequency signal):

$\Rightarrow \|W_3 T\|_\infty < 1$ where $W_3(s) = \dfrac{1}{T_{desired}(s)}$ has a **high-pass** response

$\rightarrow$ also **improves robustness to an output mult. uncertainty** (slide 30)

The weighting functions $W_1(j\omega)$ and $W_3(j\omega)$ allow to **handle the tradeoff** between $S(j\omega)$ and $T(j\omega)$. Indeed, the following equality must be satisfied:

$$S(j\omega) + T(j\omega) = I \quad \text{for all} \quad \omega \geq 0$$

so $S(j\omega)$ and $T(j\omega)$ cannot both be small in the same frequency range!

- $W_1(j\omega)$ forces $S(j\omega)$ to be small at low frequency
- $W_3(j\omega)$ forces $T(j\omega)$ to be small at high frequency

Outline

1 Preliminaries

2 A classical approach to robust stability

3 Towards a modern approach

4 Extension to robust performance

5 The $\mathcal{H}_\infty$ control problem
- The standard form
- Problem formulation
- Practical solution

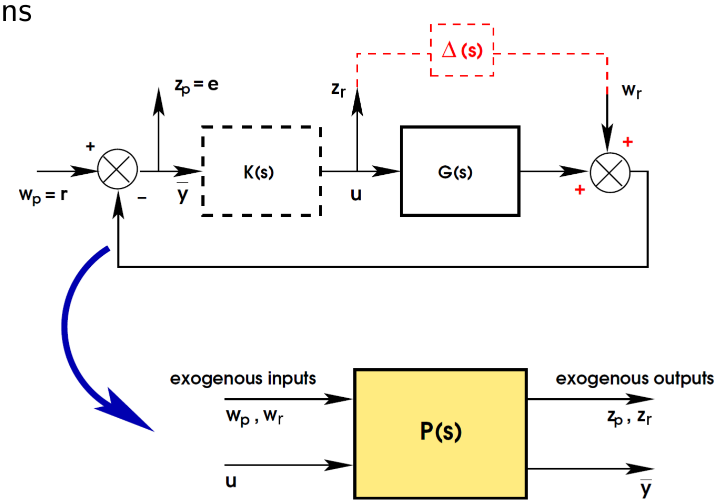
The standard form

The **standard form** $P(s)$ is an augmented design model which includes:

- **the model of the system** to be controlled
- **some exogenous inputs** (disturbances, noises, reference inputs. . .) **and outputs** (control signals, errors, controlled outputs. . .) defining the robustness and performance specifications

Example 1: see slide 34

- servo-loop performance: $\mathcal{T}_{w_p \to z_p}(s)$
- robustness to an additive uncertainty: $\mathcal{T}_{w_r \to z_r}(s)$

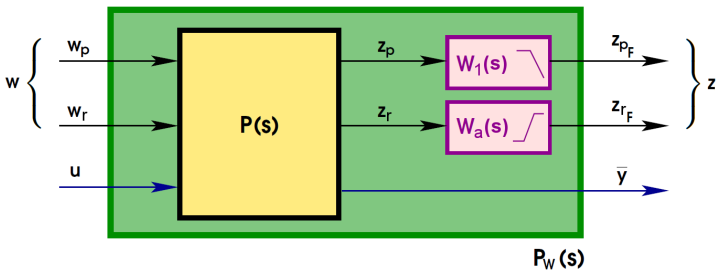


The standard form

Weighting functions are then added (see slide 30 for robustness and slides 36-37 for performance), so as to create a **weighted standard form** $P_W(s)$.

Example 1 (cont'd):

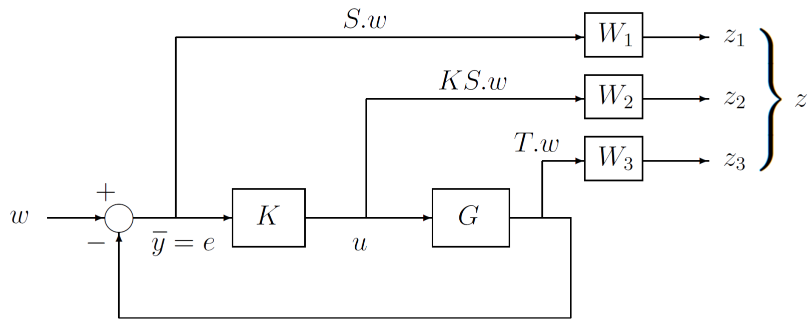
- $\mathcal{T}_{w_p \to z_{p_F}}(s) = W_1(s)S(s)$ with $W_1(s)$ a (usually first-order) low-pass filter
- $\mathcal{T}_{w_r \to z_{r_F}}(s) = W_a(s)K(s)S(s)$ with $W_a(s)$ a (usually first-order) high-pass filter



Remark: In the general case, weighting functions can also be placed on the exogenous inputs.

The standard form

Example 2: the mixed-sensitivity problem

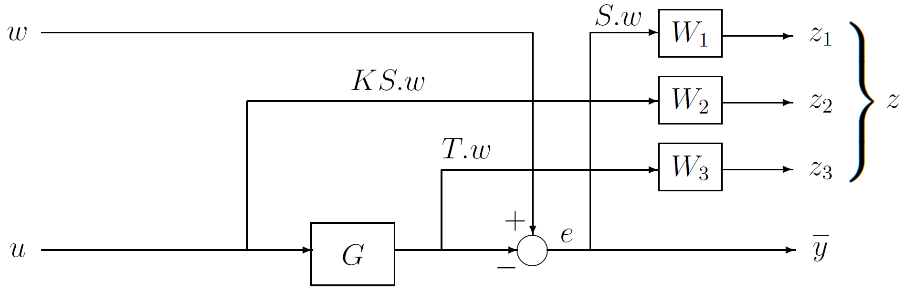


- $\mathcal{T}_{w \to z_1} = W_1 S$ where W_1 is a low-pass filter: servo-loop performance
- $\mathcal{T}_{w \to z_2} = W_2 K S$ where W_2 is a high-pass filter: actuator health
- $\mathcal{T}_{w \to z_3} = W_3 T$ where W_3 is a high-pass filter: measurement noise attenuation

$\Rightarrow$ W_1 & W_3 allow to **handle the tradeoff between S & T** (see slide 37)

The standard form

Example 2 (cont'd): the mixed-sensitivity problem



Weighted standard form $P_W(s)$

More generally, from a Simulink diagram describing $P_W(s)$ and the Matlab function `linmod`, **a representation of $P_W(s)$ can be easily computed**.

Warning

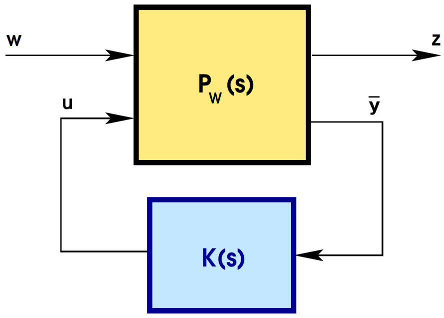
Always plug $\bar{y}$ to the **last** output ports and u to the **last** input ports.

The $\mathcal{H}_\infty$ control problem

Given a weighted standard form $P_W(s)$, the resolution of the $\mathcal{H}_\infty$ control problem consists of finding a stabilizing controller $\hat{K}(s)$ defined as follows:

$$\hat{K}(s) = arg \min_{K(s)} \|\mathcal{F}_l\left(P_W(s), K(s)\right)\|_\infty$$

where $\mathcal{F}_l(P_W(s), K(s))$ denotes the closed-loop transfer function $\mathcal{T}_{w\to z}(s)$.



Remark: $\mathcal{F}_l(.)$ is called the **lower linear fractional transformation** and $\mathcal{F}_l(M, N)$ is obtained by connecting N to the **last inputs & outputs** of M.

The $\mathcal{H}_\infty$ control problem

Optimal formulation: Find a stabilizing controller which **minimizes the value of** γ such that $\|\mathcal{F}_l(P_W(s), K(s))\|_\infty \leq \gamma$.

Suboptimal formulation: For **a given** $\gamma > 0$, find a stabilizing controller such that $\|\mathcal{F}_l(P_W(s), K(s))\|_\infty \leq \gamma$.

Example 2 (cont'd): the mixed-sensitivity problem

Find a stabilizing controller $K(s)$ such that:

$$\left\| \begin{bmatrix} W_1\ S \\ W_2\ KS \\ W_3\ T \end{bmatrix} \right\|_\infty < 1 \tag{2}$$

which implies (sufficient but not necessary condition):

$$\left\{ \begin{array}{rcl} \|W_1 S\|_\infty & < & 1 \\ \|W_2 KS\|_\infty & < & 1 \\ \|W_3 T\|_\infty & < & 1 \end{array} \right. \tag{3}$$

Practical solution: standard Riccati-based approach

- It was shown in the 1980's that the above $\mathcal{H}_\infty$ control problem is **convex** provided **the order of the controller is not constrained**: $K(s)$ has the same (usually high) order as $P_W(s)$. The **optimal controller** $\hat{K}(s)$ which minimizes γ can thus be computed.
- Since 1989, an efficient algorithm exists, which relies on the resolution of **two coupled Riccati equations** via a bissection algorithm.
- This algorithm can only be applied to "regular" standard forms which fulfill some technical assumptions.
- Regularization techniques exist and the whole design process is implemented in the function **hinfsyn** of the Matlab Robust Control Toolbox.

$\Rightarrow$ **limited success in the industry** due to the high order of the controller and its lack of structure, which makes it quite difficult to implement

$\Rightarrow$ controller reduction is possible, but the outcome is not guaranteed

Practical solution: LMI-based approach

- In 1994, the $\mathcal{H}_\infty$ control problem was rewritten as **a linear objective minimization problem over LMI constraints**. Thanks to the flexibility of the Linear Matrix Inequalities (LMI) framework, regularity assumptions are no longer required.
- This algorithm is implemented in the function **hinfsyn** of the Matlab Robust Control Toolbox.
- The LMI-based approach offers more flexibility, but is also numerically more demanding than the standard algorithm. As a result, it is still restricted today to medium-order systems involving at most 20 to 30 states. But such a limit is rapidly reached in practice with real-world applications and high-order weighting functions.

$\Rightarrow$ same practical limitations as with the standard Riccati-based approach

Practical solution: nonsmooth-based approach

- In 2006, the $\mathcal{H}_\infty$ control problem was rewritten as **a nonsmooth optimization problem**. The controller can be **structured as desired**, and **no regularization is required**. But the resulting problem is non convex.
- Since 2010, **structured and fixed-order** $\mathcal{H}_\infty$ controllers can be computed with the function **hinfstruct** – based on specialized nonsmooth optimization algorithms by P. Apkarian and D. Noll. Although local solutions (*i.e.* suboptimal controllers) are obtained, this routine works very well in practice and an almost minimum value of γ is often obtained.
- A **structured performance index** can also be considered. If $\mathcal{F}_l(P_W(s), K(s))$ is replaced with $\mathrm{diag}(\mathcal{F}_l(P_1(s), K(s)), \ldots, \mathcal{F}_l(P_n(s), K(s)))$, then:

$$\hat{K}(s) = arg \min_{K(s)} \max_{1 \leq k \leq n} \|\mathcal{F}_l\left(P_k(s), K(s)\right)\|_\infty$$

  For example, a stabilizing controller $K(s)$ satisfying (3) instead of (2) is computed for the mixed-sensitivity problem, which reduces conservatism.

Practical solution: nonsmooth-based approach

- Additional features are now provided by the function **systune**. $\mathcal{H}_\infty/\mathcal{H}_2/$**time-domain objectives and pole constraints** can be mixed. Moreover, **several models** can be considered simultaneously to enforce parametric robustness. And the latest version also handles **uncertain design models**: worst-cases are extracted and used in an automatized robust design process.
- The function **systune** sometimes pays the price of its versatility and becomes difficult to use even for expert $\mathcal{H}_\infty$ control designers.

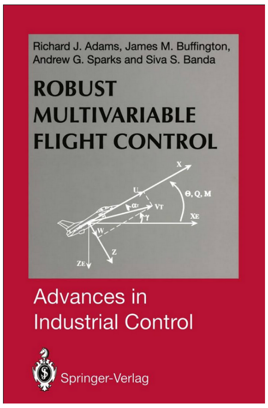
$\Rightarrow$ **growing success in the industry** due to its flexibility: any controller structure and order can be considered (*e.g.* PID gains can be optimized to satisfy $\mathcal{H}_\infty$ constraints)

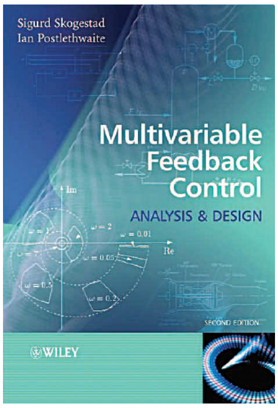
$\Rightarrow$ it can be applied to high-order systems with many specifications

$\Rightarrow$ several additional features are available (structured performance index, multi-model design, $\mathcal{H}_2$/time-domain/pole specifications...)

References

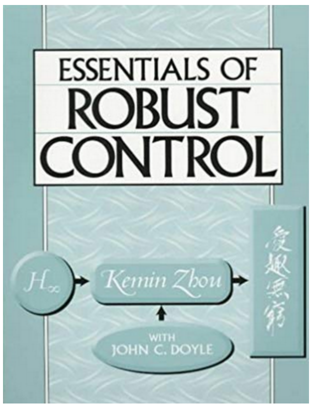
- **Robust Multivariable Flight Control.** *Advances in Industrial Control.*
  R. Adams, J. Buffington, A. Sparks and S. Banda.
  Springer Verlag, London 1994.
- **Multivariable Feedback Control.** *Analysis and Design.*
  S. Skogestad and I. Postlethwaite Wiley, 2005.

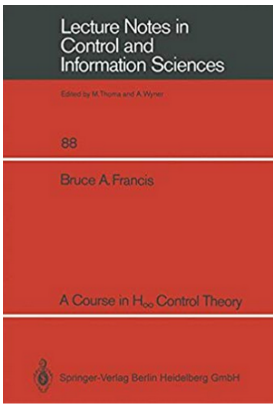




References

- **Essentials of Robust Control.**
  K. Zhou and J. Doyle. Prentice Hall, New Jersey 1999.
- **A Course in H_∞ Control Theory.**
  B. Francis. Springer Verlag, 1987.





```
%###########################
%  MATLAB DEMO - HINF DESIGN
%###########################
%
% simple model of a rigid satelite: theta(s) = 1/(J*s^2) * u(s)
% theta = pointing angle
% u = commanded torque
%
% closed-loop objectives:
% 1. track a second-order reference model (xi=0.7 and omega=2.5rad/s)
% 2. avoid large values of the control input u (actuator saturations)
% 3. ensure robustness in the presence of uncertainties on the inertia
%
clear variables
close all
%
J=0.5; % nominal value of the inertia
%
%################################################
%  First tuning: using static weighting functions
%################################################
%
% weighting functions
num1=1;den1=1;  % W1=1 (performance channel between w and z1)
num2=1;den2=1;  % W2=1 (control channel between w and z2)
%
% open-loop plant P(s)
[a,b,c,d]=linmod('designModel1');
P=ss(a,b,c,d);
%
% full-order design
r=[3 1]; % 2 measurements + 1 reference input ; 1 control input
[K,Gc,gopt]=hinfreg(P,r,100);  % gopt = 0.781
damp(K)
damp(Gc)
% => a fourth-order controller is obtained since the order of P is 4 (two integrators
%    + second-order reference model)
%
% time-domain simulation
simModel1;
% => large static error
% => small control amplitude = 0.05  => increase W1 or decrease W2
%
%#################################################
%  Second tuning: using static weighting functions
%#################################################
%
% weighting functions
num1=1;den1=1;    % W1=1 (performance channel between w and z1)
num2=0.1;den2=1;  % W2=0.1 (control channel between w and z2)
%
% open-loop plant P(s)
[a,b,c,d]=linmod('designModel1');
P=ss(a,b,c,d);
%
% full-order design
r=[3 1];
[K,Gc,gopt]=hinfreg(P,r,100);  % gopt = 0.293
damp(K)
damp(Gc)
```

```
%
% time-domain simulation
simModel1;
% => static error is reduced but is still there
% => control activity = 0.23
%
%#################################################
%  Third tuning: using static weighting functions
%#################################################
%
% weighting functions
num1=1;den1=1;     % W1=1 (performance channel between w and z1)
num2=0.01;den2=1;  % W2=0.01 (control channel between w and z2)
%
% open-loop plant P(s)
[a,b,c,d]=linmod('designModel1');
P=ss(a,b,c,d);
%
% full-order design
r=[3 1];
[K,Gc,gopt]=hinfreg(P,r,100);  % gopt = 0.098
damp(K)
damp(Gc)
%
% time-domain simulation
simModel1;
% => static error is reduced and almost acceptable
% => control activity = 0.43
%
%####################################################
%  Forth tuning: using a low-pass performance filter
%####################################################
%
% weighting functions
W10=200;
tau=100;
num1=200;den1=[tau 1];  % W1=W10/(1+tau*s) (performance channel between w and z1)
num2=0.07;den2=1;       % W2=0.07 (control channel between w and z2)
%
figure(1)
bode(tf(num1,den1),logspace(-3,2,100));  % Bode plot of W1
hold on,grid on
bode(tf(6.25,[1 3.5 6.25]));             % Bode plot of the reference model
% => W1 is tuned so that |W1(w)| satisfies |W1(2)|=1, i.e. wc=2 rad/s
% => this corresponds approximately to the bandwidth of the reference model
%
% open-loop plant P(s)
[a,b,c,d]=linmod('designModel1');
P=ss(a,b,c,d);
%
% 1. full-order design
r=[3 1];
[K,Gc,gopt]=hinfreg(P,r,100);  % gopt = 0.195
damp(K)
damp(Gc)
%
% time-domain simulation
simModel1;
% => very low static error
% => control activity = 0.32
```

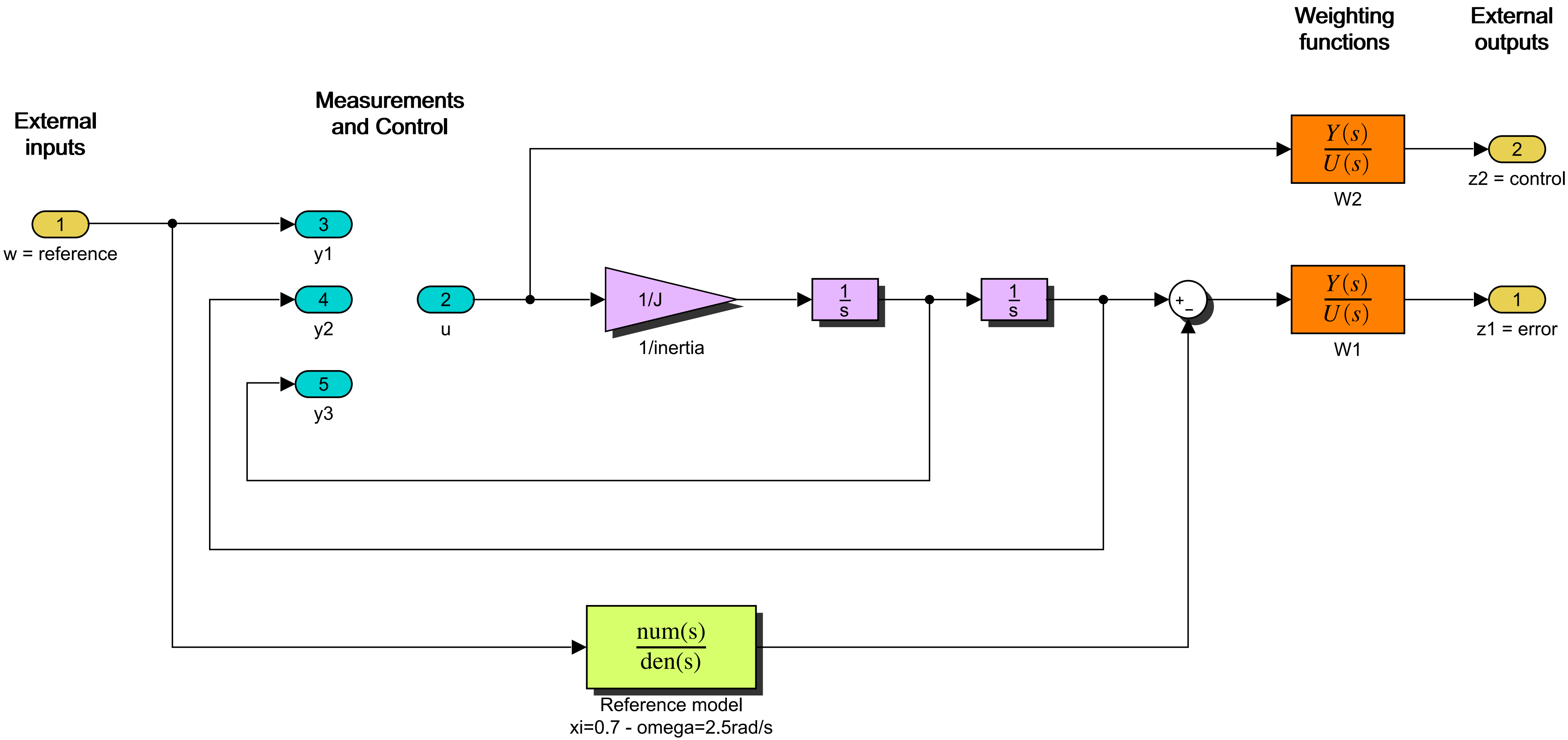
```
% => much better results than with constant weighting functions
%
% 2. fixed-order design with hinfstruct
K0=ltiblock.gain('K0',1,3);  % static controller (3 measurements, 1 control input)
opt=hinfstructOptions('randomstart',3);
[Kf,gopt]=hinfstruct(P,K0,opt);  % gopt = 0.175
K=ss(Kf);
damp(feedback(P,K,2,3:5,1))
%
% time-domain simulation
simModel1;
% => very low static error
% => control activity = 0.43
% => closed-loop poles very close to the reference model (xi~0.69 and w~2.24rad/s)
% => gamma smaller than with the full-order controller (in fact, the latter leads to
%    a slightly better value of gamma (0.1724) if the tolerance in hinfsyn is set to
%    0.001, but a very fast pole appears in K and the time-domain simulations are not
%    satisfactory)
% => results similar to the full-order case with a static controller except the
%    control activity which is slightly larger
%
%############################################
%  Controller design without speed measurement
%############################################
%
% weighting functions
W10=200;
tau=100;
num1=200;den1=[tau 1];  % W1=W10/(1+tau*s) (performance channel between w and z1)
num2=0.07;den2=1;       % W2=0.07 (control channel between w and z2)
%
% open-loop plant P(s)
[a,b,c,d]=linmod('designModel2');
P=ss(a,b,c,d);
%
% 1. full-order design
%
r=[2 1]; % 1 measurement + 1 reference input ; 1 control input
[K,Gc,gopt] = hinfreg(P,r,1);  % gopt = 0.188 => same as before!
damp(K)
damp(Gc)
%
% time-domain simulation
simModel2;
% => very low static error
% => control activity = 0.33
% => same results as before even if the speed measurement is not available
%
% 2. fixed-order design with hinfstruct
%
K0=ltiblock.gain('K0',1,2);  % static controller (2 measurements, 1 control input)
opt=hinfstructOptions('randomstart',3);
[Kf,gopt]=hinfstruct(P,K0,opt);  % fail to enforce closed-loop stability
K=ss(Kf);
% => a double integrator cannot be stabilized by a static controller if only the
%    position measurement is available (see root locus below)
figure
subplot(1,2,1),rlocus(P(4,2));  % transfer between measured position and control input
subplot(1,2,2),rlocus(-P(4,2));
%
```

```
K0=ltiblock.ss('K0',1,1,2);  % 1st order controller (2 measurements, 1 control input)
opt=hinfstructOptions('randomstart',3);
[Kf,gopt]=hinfstruct(P,K0,opt);  % gopt = 0.173 => same as before!
K=ss(Kf);
damp(K)
damp(feedback(P,K,2,3:4,1))
%
% time-domain simulation
simModel2;
% => very low static error
% => control activity = 0.43
% => same results as before even if the speed measurement is not available, but a 1st
%    order controller is needed to estimate the velocity which is not measured anymore
%
%############################################
%  Controller design with uncertainties on J
%############################################
%
% 1. Evaluation of the different criteria for the previous controller
%
[a,b,c,d]=linmod('analysisModel');
CL=ssbal(ss(a,b,c,d));
W1=tf(num1,den1);
W2=tf(num2,den2);
normhinf(blkdiag(W1,W2)*CL(1:2,1))
% => gamma = 0.173 (same result as above)
normhinf(CL(3,2))
% => gamma ~ 1.44 (can slightly vary due to random initialization of hinfstruct)
% => stability proved for J=J0*(1+/-deltaJ) with |deltaJ| < 1/1.44 = 0.69 (small gain
%    theorem), ie 69% multiplicative uncertainty
%
% 2. New design taking into account the uncertainty on J
%
% the Hinf norm of the transfer between [w1;w2] and [z1;z2;z3] is minimized
%
% weighting functions
W10=200;
tau=100;
num1=200;den1=[tau 1];  % W1=W10/(1+tau*s) (performance channel between w1 and z1)
num2=0.07;den2=1;       % W2=0.07 (control channel between w1 and z2)
num3=0.16;den3=1;       % W3=0.16 (uncertainty channel between w2 and z3)
%
% open-loop plant P(s)
[a,b,c,d]=linmod('designModel3');
P=ss(a,b,c,d);
%
% fixed-order design with hinfstruct
K0=ltiblock.ss('K0',2,1,2);  % 2nd order controller (2 measurements, 1 control input)
opt=hinfstructOptions('randomstart',10);
[Kf,gopt]=hinfstruct(P,K0,opt);  % gopt = 0.559
K=ss(Kf);
damp(K)
% => a second-order controller is required (gopt>5 for a first-order controller, and
%    no improvement if the order is larger than 2)
% => the poles of the controller are not very satisfactory (one is slow around -1e-2
%    and one is fast around -1e2), so numerical problems can be expected
%
% evaluation of the different criteria
[a,b,c,d]=linmod('analysisModel');
CL=ssbal(ss(a,b,c,d));
```

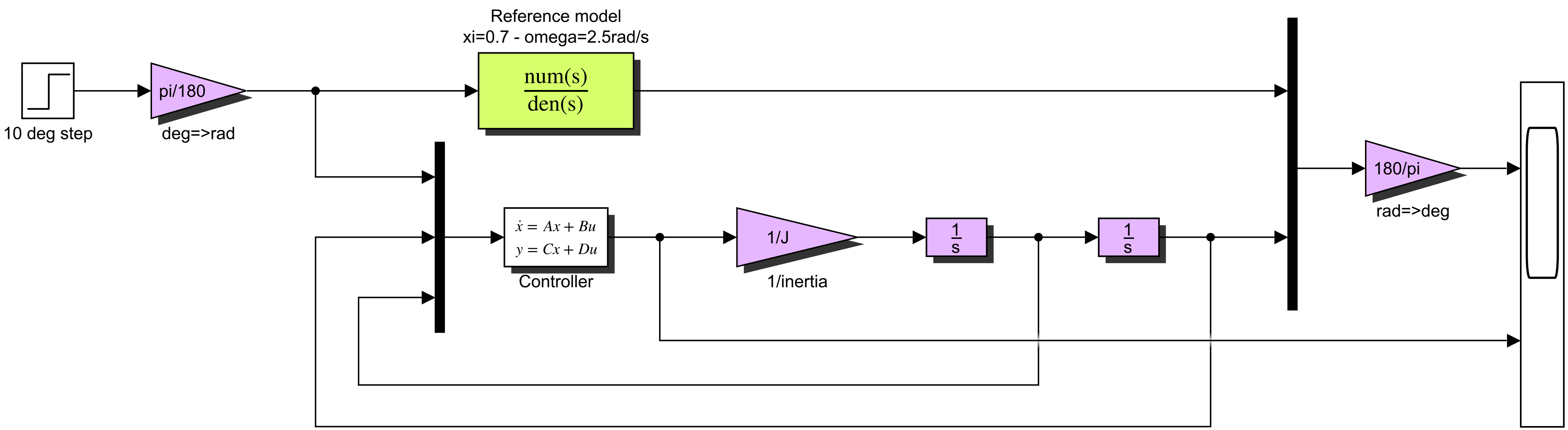
```
normhinf(blkdiag(W1,W2)*CL(1:2,1))
% => gamma ~ 0.51 => degradation of performance and control activity
normhinf(CL(3,2))
% => gamma ~ 1.08 (between 1.01 and 1.20) => robustness improvement
% => stability proved for J=J0*(1+/-deltaJ) with |deltaJ| < 1/1.08 = 0.92 (small gain
%    theorem), ie 92% multiplicative uncertainty
% => robustness has been improved, but a significant degradation of performance and/or
%    control activity is observed
%
% time-domain simulation
simModel2;
% => the reference model is not tracked very well and numerical problems can be
%    observed (see control input plot)
%
% 3. New design taking into account two different objectives
%
% instead of minimizing the Hinf norm of a single transfer between [w1;w2] and
% [z1;z2;z3], the following problem is considered:
% min gamma such that ||w1->[z1;z2]||_inf < gamma    (performance + control activity)
%                     ||w2->z3||_inf < gamma         (uncertainty rejection)
% solving this problem is possible with hinfstruct, but not with the old full-order
% routine hinfsyn (which can only minimize the norm of a single transfer)
%
% fixed-order design with hinfstruct
K0=ltiblock.ss('K0',1,1,2);  % 1st order controller (2 measurements, 1 control input)
CL0=lft(P,K0);
CL1=CL0(1:2,1);          % mixed-sensitivity
CL2=CL0(3,2);            % uncertainty rejection
CL0f=blkdiag(CL1,CL2);  % concatenation of both Hinf constraints
options=hinfstructOptions('RandomStart',10);
[CL,gopt]=hinfstruct(CL0f,options);  % gopt = 0.175
K=ss(CL.Blocks.K0);
damp(K)
% => a first-order controller is sufficient
% => the controller pole is around -45, so numerical problems should be avoided
%
% evaluation of the different criteria
[a,b,c,d]=linmod('analysisModel');
CL=ssbal(ss(a,b,c,d));
normhinf(blkdiag(W1,W2)*CL(1:2,1))
% => gamma ~ 0.175 => no degradation of performance and control activity
normhinf(CL(3,2))
% => gamma ~ 1.07 => robustness improvement
% => stability proved for J=J0*(1+/-deltaJ) with |deltaJ| < 1/1.07 = 0.93 (small gain
%    theorem), ie 93% multiplicative uncertainty
% => robustness has been improved, and no significant degradation of performance
%    and/or control activity is observed
%
% time-domain simulation
simModel2;
sys=linmod('simModel2');
damp(sys.a)
% => no numerical problem can be observed
% => the reference model is correctly tracked
% => the closed-loop poles are satisfactory (second-order system with omega~2.2rad/s
%    and xi~0.69 very close to the reference model + controller pole at around -40)
```

```
function [K,Gc,gopt] = hinfreg(sys,r,gmax,tol)
%
% [K,Gc,gopt] = hinfreg(sys,r,gmax,tol);
%
% This function is based on hinfsyn.  A regularization step is
% included to make the synthesis plant non singular.
%
% INPUTS:
% -------
%    sys  : synthesis plant in ss format
%    r    : 1x2 vector = [nb of measurements, nb of control inputs]
%    gmax : initial guess for maximum Hinf norm
%    tol  : tolerance level for regularization (optional argument)
%               default = 1d-5
%               tol=[] => no regularization is applied
%
% OUTPUTS:
% --------
%    K    : optimal Hinf controller in ss format
%    Gc   : closed-loop plant
%    gopt : minimized Hinf norm
%
```

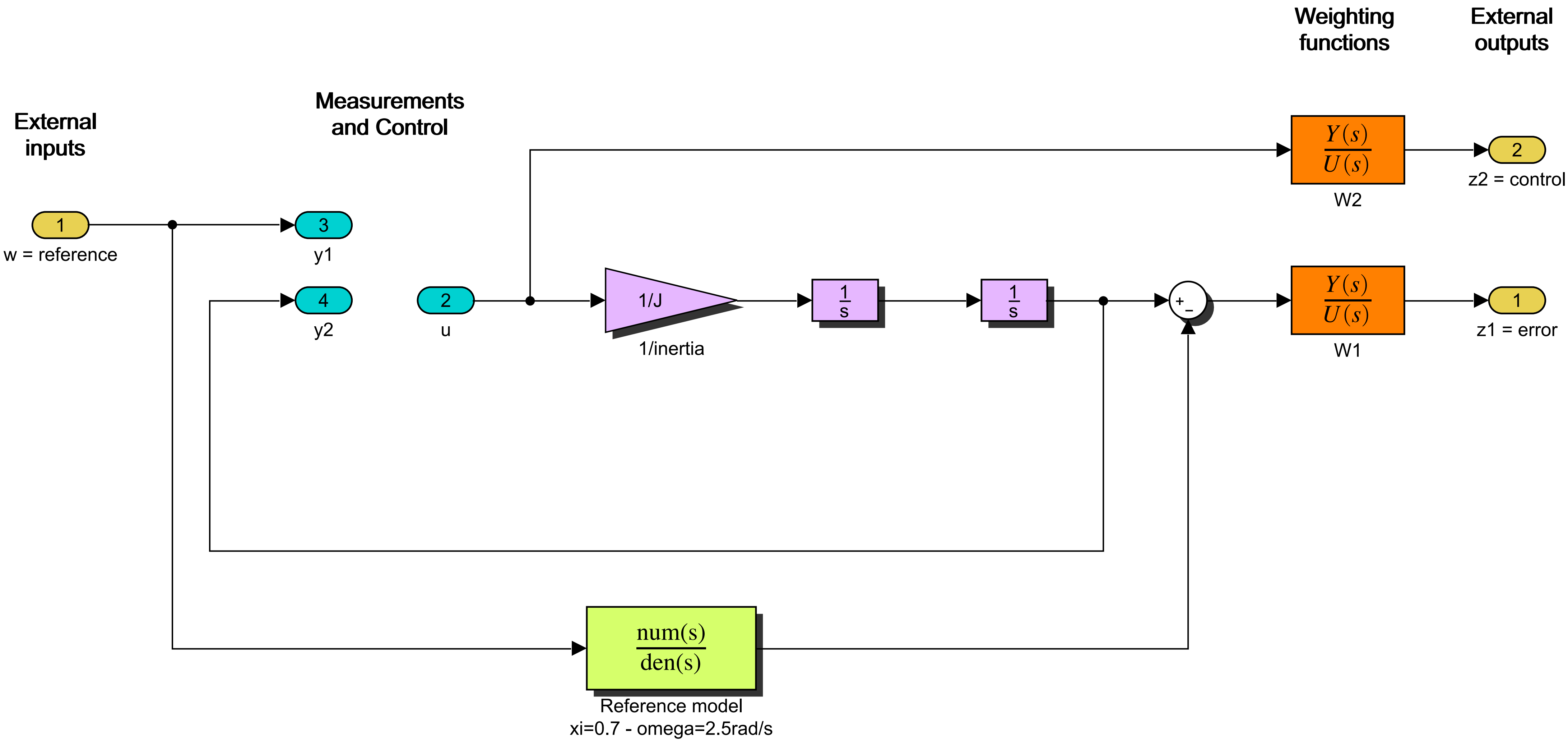
designModel1.slx => HINF DESIGN MODEL #1



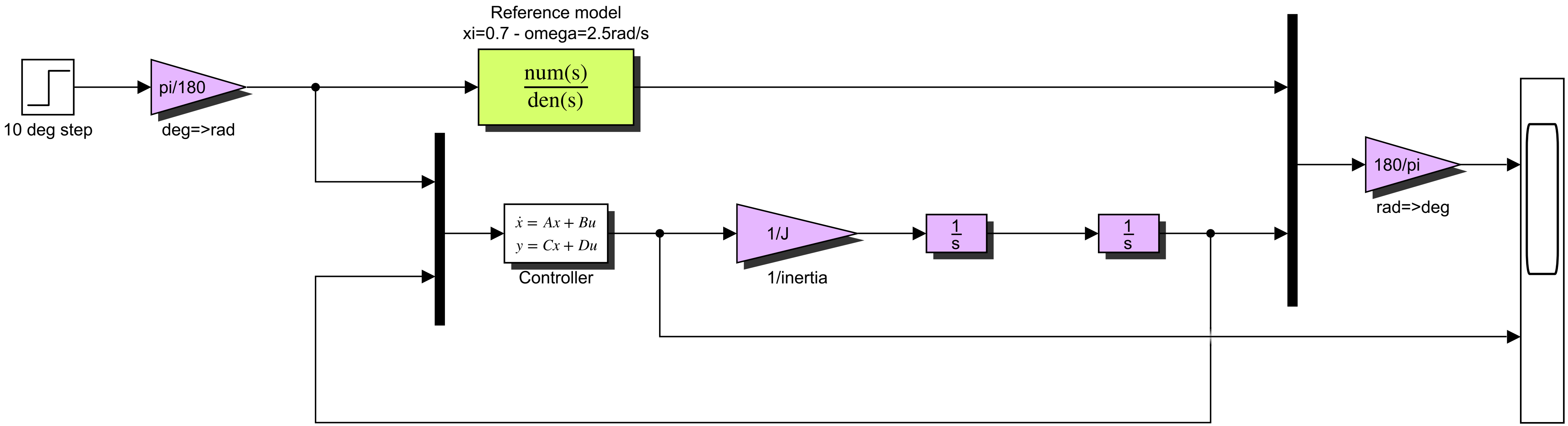
simModel1.slx => CLOSED-LOOP MODEL #1



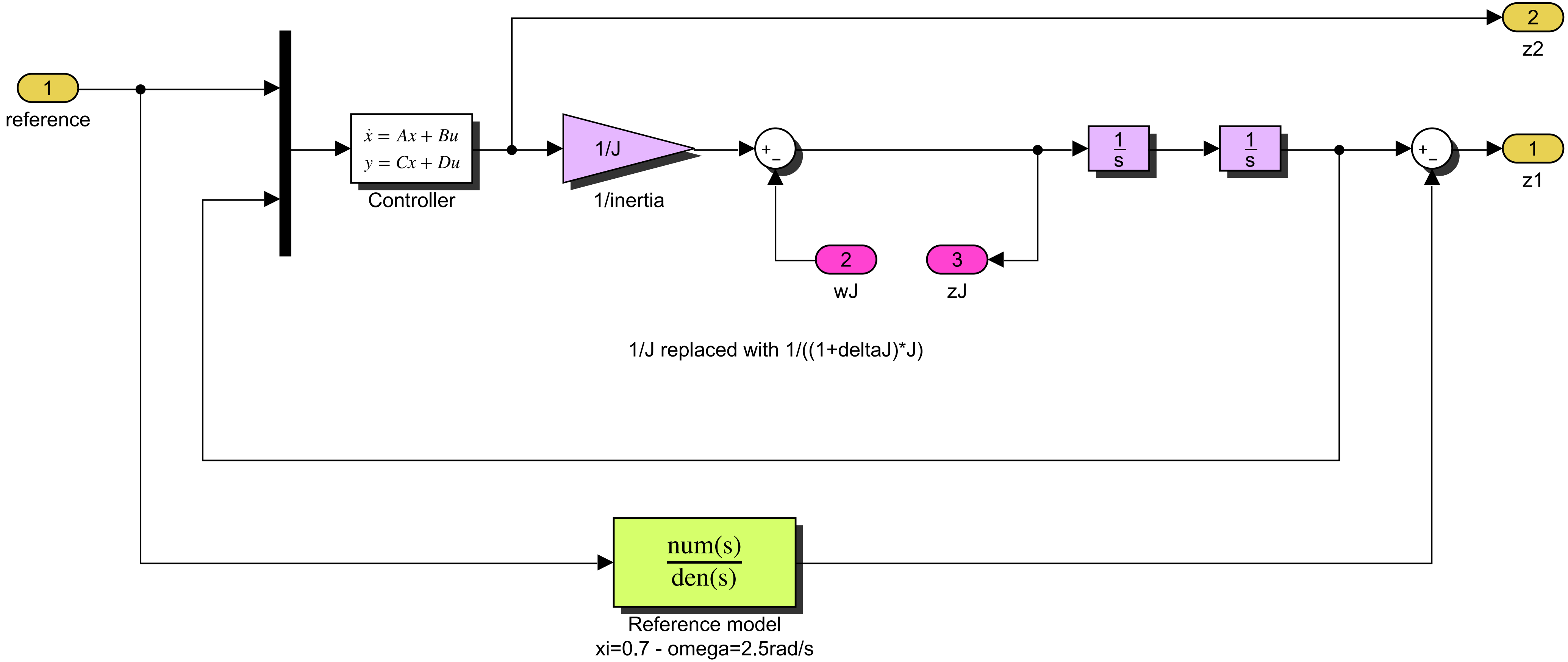
designModel2.slx => HINF DESIGN MODEL #2



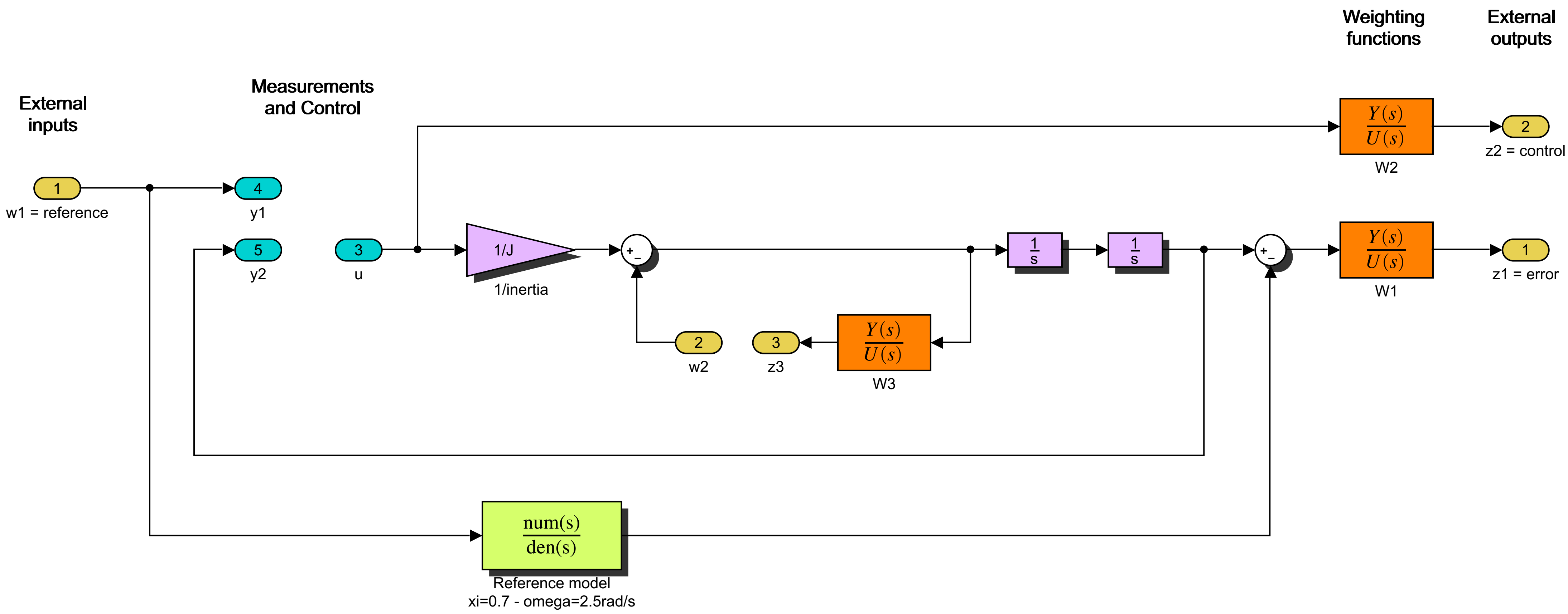
simModel2.slx => CLOSED-LOOP MODEL #2



analysisModel.slx => ROBUSTNESS ANALYSIS vs uncertainties on J



designModel3.slx => HINF DESIGN MODEL #3



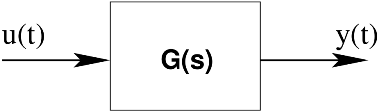
A SIMPLE MIXED $\mathcal{H}_2/\mathcal{H}_\infty$ CONTROL PROBLEM

Clément Roos

Information Processing and Systems Department

ONERA Toulouse

The $\mathcal{H}_2$ norm



Frequency-domain definition

$$\|G(s)\|_2 = \left(\frac{1}{2\pi} \int_{-\infty}^{+\infty} \text{Trace}\,\left(G^*(j\omega)G(j\omega)\right) d\omega \right)^{1/2}$$

$\|G(s)\|_2$ is only finite for **stable** and **strictly proper** (*i.e.* $D = 0$) systems. Otherwise, $\|G(s)\|_2 = \infty$.

Interpretation:

- If $G(s)$ is a SISO system, $\|G(s)\|_2^2$ is the energy of y when u is an impulse.
- In the general case, $\|G(s)\|_2^2$ is the variance of y when u is a centered normalized white noise, *i.e.* a random signal such that $U(j\omega)U^*(j\omega) = I$.

Practical computation of the $\mathcal{H}_2$ norm

Let $G(s)$ be a stable and strictly proper transfer: $G(s) = C(sI - A)^{-1}B$. The Parseval's theorem is applied, noting that $\mathcal{F}(Ce^{At}B) = G(j\omega)$, where $\mathcal{F}(.)$ denotes the Fourier transform.

Time-domain definition

$$\|G(s)\|_2 = \left(\text{Trace} \int_0^{+\infty} B^T e^{A^T t} C^T C e^{At} B\right)^{1/2}$$

$Q_o = \int_0^{\infty} e^{A^T t} C^T C e^{At} \, dt$ is called the **Observability Gramian**. It is a definite positive (symmetric) matrix solution of the Lyapunov equation:

$$A^T Q_o + Q_o A + C^T C = 0$$

$$\Rightarrow \; \|G(s)\|_2^2 = \text{Trace}(B^T Q_o B) \qquad (\text{Matlab function } \mathbf{norm})$$

Noises in theory and practice

Using **centered normalized white noises** allows to derive many important theoretical results. But such signals **do not exist**. Indeed, they are characterized by a **constant power spectral density** (PSD = power distribution as a function of frequency):

$$\phi(\omega) = 1 \quad \forall \omega \geq 0$$

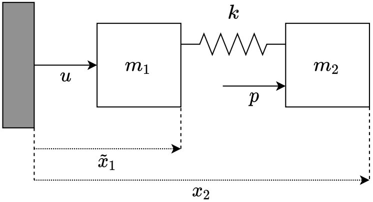
which implies that their power $P = \dfrac{1}{2\pi} \displaystyle\int_{-\infty}^{+\infty} \phi(\omega) d\omega$ is infinite!

In practice, noises have a **finite power**, for example $\phi(\omega) = \frac{1}{1+\omega^2}$.

- Replacing $j\omega$ by s gives the complex spectrum: $\phi(s) = \frac{1}{1-s^2}$.
- The later can then be factorized: $\phi(s) = \underbrace{F(s)}_{stable} \times \underbrace{F(-s)}_{unstable} = \frac{1}{1+s} \times \frac{1}{1-s}$.
- By filtering a centered normalized white noise by $F(s)$, a random signal with the same PSD as the considered noise is finally obtained.

An academic spring-mass example

Let us consider a spring-mass system, where u is the control signal (force) and p is a disturbance (force). The measurement y of the position of the mass m_2 is perturbed by a random signal b ($y = x_2 + b$) with a power spectral density (PSD) $\phi(\omega) = \frac{1}{1+\omega^2}$.



Let $x_1 = \tilde{x}_1 + \Delta x_{eq}$

- compression if $x_1 - x_2 > 0$
- extension if $x_1 - x_2 < 0$

Specifications:

- actuator fatigue alleviation: minimize the variance of $\dot{u}$ (the time-derivative of the control signal) in response to the measurement noise b
- disturbance rejection: $|\mathcal{T}_{p \to x_2}(j\omega)| < A \;\; \forall \omega \geq 0$

Construction of the weighted standard form $P_W(s)$

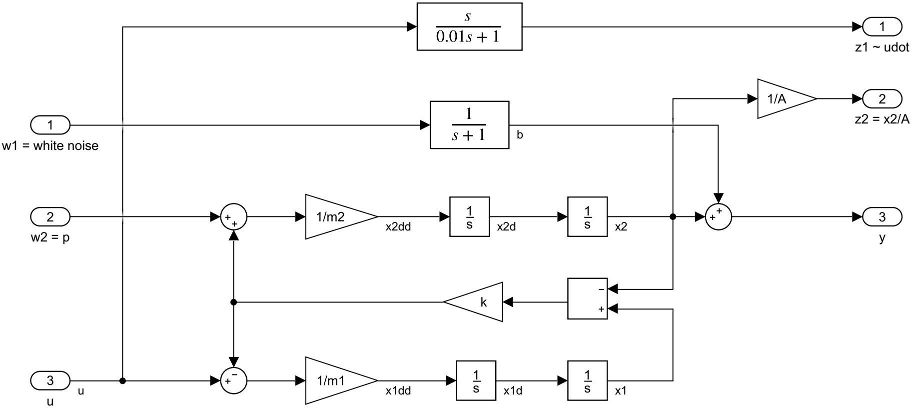
System equations:

$$\begin{aligned}\ddot{x}_1 &= \frac{1}{m_1}\left(u - k\left(x_1 - x_2\right)\right)\\ \ddot{x}_2 &= \frac{1}{m_2}\left(p + k\left(x_1 - x_2\right)\right)\\ y &= x_2 + b\end{aligned}$$

Inclusion of the specifications:

- addition of the transfer function $F(s) = \frac{1}{s+1}$, so that $b = F(s)w_1$ is a random signal with PSD $\phi(\omega) = \frac{1}{1+\omega^2}$ obtained from the centered normalized white noise w_1
- addition of a pseudo-derivator $H(s) = \frac{s}{0.01s+1}$, so that $z_1 = H(s)u$ is a good approximation of $\dot{u}$ inside the system bandwidth
- addition of a static weighting function $W = A^{-1}$, so that $|\mathcal{T}_{p\rightarrow x_2}(j\omega)| < A$ $\forall\omega \geq 0$ is equivalent to $\|\mathcal{T}_{w_2\rightarrow z_2}(s)\|_\infty < 1$, where $w_2 = p$ and $z_2 = Wx_2$

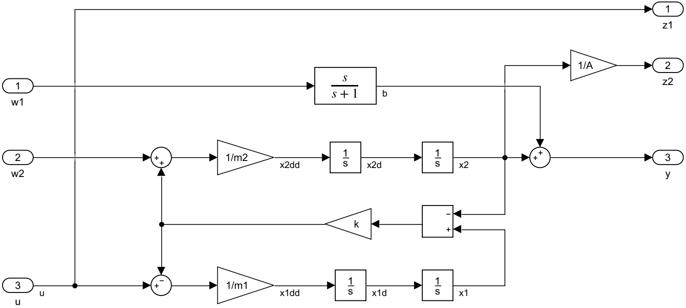
Construction of the weighted standard form $P_W(s)$



Improved weighted standard form

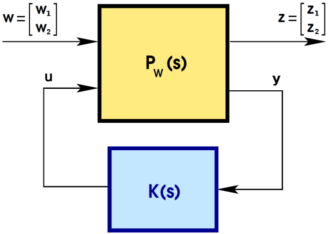
$$\begin{aligned}\mathcal{T}_{w_1\to z_1}(s) &= \frac{s}{0.01s+1}\mathcal{T}_{y\to u}(s)\frac{1}{s+1}\\ &= \frac{1}{0.01s+1}\mathcal{T}_{y\to u}(s)\frac{s}{s+1}\end{aligned}$$

The filter $\frac{1}{0.01s+1}$ was only introduced for regularization and is no longer needed $\Rightarrow \mathcal{T}_{w_1\to z_1}(s) = \mathcal{T}_{y\to u}(s)\frac{s}{s+1}$ is preferred.



Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem

Let $\mathcal{F}_l(P_k(s), K(s))$ denote the closed-loop transfer function $\mathcal{T}_{w_k \to z_k}(s)$.



Control problem

Find a stabilizing controller $\widehat{K}(s)$ such that:

$$\widehat{K}(s) = \arg \min_{K(s) \in \mathcal{K}} \|F_l(P_1(s), K(s))\|_2 \qquad \textbf{(soft constraint)}$$

where $\mathcal{K} = \{K(s) : \|F_l(P_2(s), K(s))\|_\infty \leq 1\}$ **(hard constraint)**

INTRODUCTION TO ROBUSTNESS ANALYSIS

via a practical approach to μ-analysis

Clément Roos

ONERA/DCSD - Toulouse, France

clement.roos@onera.fr

ROBUSTNESS ANALYSIS: context and objective

Observation: Control laws design is usually based on a (linear) mathematical model Σ_0, which significantly simplifies/alters reality (neglected dynamics, badly-known physical phenomena...).

Assumption: The behavior of a physical system can be accurately described by a (possibly infinite) set of simple models $(\Sigma_i)_i$.

Objective

Check whether the control laws designed using the single linear time-invariant model Σ_0 ensure good performances on the whole set of models $(\Sigma_i)_i$. If this is true, it can be guaranteed that good performances will be obtained on the real system.

ROBUSTNESS ANALYSIS: context and objective

A linear time-invariant (LTI) model is not a perfect representation of the real behavior of a physical system because of:

- high-frequency uncertainties (neglected dynamics)
- uncertainties on the parameters which characterize the system (mass, inertia, aerodynamic coefficients...)
- time-varying parameters:
  - fast variations $\rightarrow$ mass of a launcher during atmospheric flight
  - slow variations $\rightarrow$ mass, velocity, altitude of a transport aircraft
- nonlinear phenomena:
  - aerodynamic phenomena at high angles
  - actuators saturations
  - transmission delays
  - intrinsically nonlinear phenomena

Robustness with respect to these phenomena must be ensured!

EXAMPLE: clearance of flight control laws

Before an aircraft can be tested in flight, it has to be proven to the authorities that the flight control system is reliable.

Classical industrial approach = Monte-Carlo simulations:

1. choose many random samples, each of them being composed of random operating points (*e.g.* mass configurations, CoG position...) and random inputs (*e.g.* pilot inputs, wind...),
2. perform a closed-loop simulation for each sample,
3. perform statistical analysis on the resulting output samples to get the probabilities that some stability, performance, loads and comfort criteria are satisfied.

Advantage:

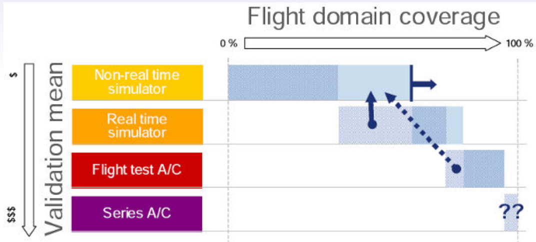
- easy to implement

Drawbacks:

- exponential-time approach $\Rightarrow$ high computational complexity
- statistical approach $\Rightarrow$ worst cases can be missed

EXAMPLE: clearance of flight control laws

New trend: develop some inexpensive tools so as to determine quickly the most critical parametric configurations without simulations.



Some techniques such as μ, IQC-based of Lyapunov-based analysis can be efficient alternatives to Monte-Carlo simulations.

Advantages:

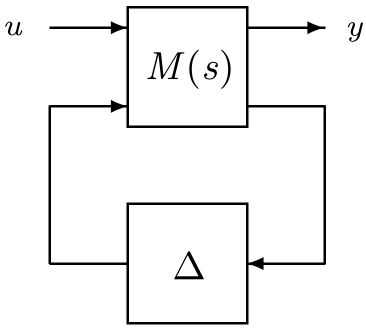
- polynomial-time approach $\Rightarrow$ reduction of the computational burden
- deterministic approach, the criteria can be checked for all admissible operating points $\Rightarrow$ impossible to miss worst cases

Drawbacks:

- requires a preliminary modeling step
- fewer criteria can be checked than with a simulation-based approach

ROBUSTNESS ANALYSIS: standard form

Preliminary task: the physical system must be described by an LFR (Linear Fractional Representation, also called standard form).



Several elements can be isolated in Δ:

- time-invariant uncertainties (parametric and dynamic)
- time-varying parameters
- non-linearities (saturations, deadzones, sector non-linearities. . .)

ROBUSTNESS ANALYSIS: focus on μ-analysis

This presentation focuses on μ-analysis, which allows to consider:

- time-invariant uncertainties (parametric and dynamic)
- ~~time-varying parameters~~
- ~~non-linearities (saturations, deadzones, sector non-linearities...)~~

\- Only time-invariant uncertainties can be considered

\+ Allows to address several important practical issues

\+ Much faster than IQC-based and Lyapunov-based analysis

\+ Usually less conservative than IQC-based analysis

\+ Computation of guaranteed stability margins & performance levels (unlike simulation-based approaches), and of worst-case configurations

ROBUSTNESS ANALYSIS: focus on μ-analysis

μ-analysis and Monte-Carlo simulations can be complementary:

1. Build a high-fidelity LFR from the initial model.
2. Check the stability and performance criteria of interest over the admissible set of parametric configurations using μ-analysis techniques.
3. Use traditional methods such as simulations to examine only the worst-case configurations that have been identified at step 2.

μ-analysis has two main advantages compared to the standard robustness analysis tools currently used by the industry:

- Criteria can be checked for all admissible parametric combinations $\Rightarrow$ impossible to miss a worst case configuration.
- The computational time increases exponentially with the number of parameters for simulation-based approaches and polynomially for μ-based algorithms $\Rightarrow$ reduction of the computational burden.

ISSUES ADDRESSED BY μ-ANALYSIS

Framework:

- LTI nominal closed-loop model
- LTI uncertainties (parametric uncertainties, neglected dynamics), and possibly delays and a few non-linearities

Main robustness issues:

- determine whether the poles of the uncertain closed-loop model are inside a given region of the complex plane (left half plane, unit ball, truncated sector...) $\rightarrow$ stability
- determine whether a closed-loop sensitivity function satisfies a frequency-domain template despite model uncertainties $\rightarrow$ H_∞ performance
- determine whether the gain/phase/delay margins are sufficient despite model uncertainties

NEGLECTED DYNAMICS: physical meaning

A classical way to model a physical system or process is to use identification techniques.

$\rightarrow$ accurate results on a limited frequency interval (low frequencies)

$\rightarrow$ high-frequency phenomena usually ignored

If the bandwidth of the controller is too large, the system can be excited in a frequency interval where its physical behavior is badly known, or even unknown.

$\rightarrow$ can lead to instability

The robustness of the closed-loop system to neglected dynamics must be analyzed.

NEGLECTED DYNAMICS: mathematical definition

Neglected dynamics = stable unstructured transfer matrix $\Delta(s)$, totally unknown except a constraint on its H_∞ norm:

$$\|\Delta(s)\|_\infty \leq 1 \quad \Leftrightarrow \quad \forall\omega \;\; \overline{\sigma}(\Delta(j\omega)) \leq 1$$

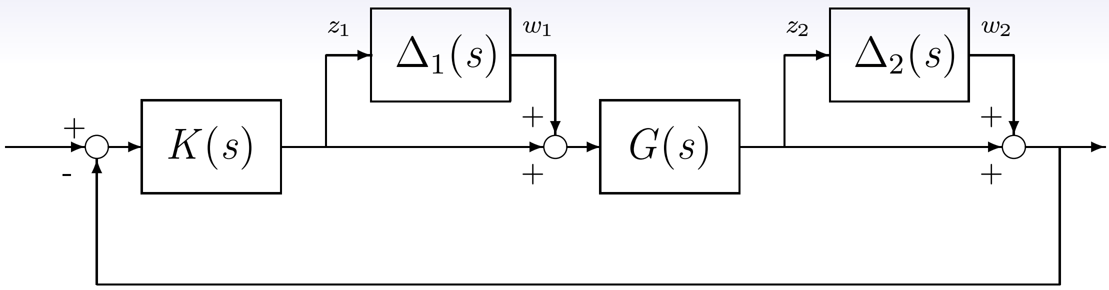
Addition of a stable frequency-domain weighting function $W(s)$:

$$G(s) = G_0(s) + W(s)\Delta(s) \qquad \text{additive dynamics}$$
$$G(s) = (1 + W(s)\Delta(s))\,G_0(s) \qquad \text{multiplicative dynamics}$$

$\Delta(s)$ does not need to be square (addition of rows/columns of 0).

At a fixed frequency ω, $\Delta(j\omega)$ is a full complex block, i.e. an unknown and unstructured complex matrix s.t. $\overline{\sigma}(\Delta(j\omega)) \leq 1$.

NEGLECTED DYNAMICS: example



- multiplicative dynamics added at the input and output of the open-loop plant
- nominal (asymptotically stable) closed-loop model recovered if $\Delta_1(s) = \Delta_2(s) = 0$
- robustness test: determine whether the closed-loop model is stable $\forall \Delta_i(s)$ such that $\|\Delta_i(s)\|_\infty \leq 1$
- robustness margin: compute the largest value of k for which the closed-loop model remains stable $\forall \Delta_i(s)$ such that $\|\Delta_i(s)\|_\infty \leq k$

PARAMETRIC UNCERTAINTIES: definition

Unlike neglected dynamics, parametric uncertainties appear inside the system bandwidth (frequency interval for which the model is representative of the true system) $\Rightarrow$ they are introduced into the open-loop model:

$$\begin{cases} \dot{x} &= A(\delta)x + B(\delta)u \\ y &= C(\delta)x + D(\delta)u \end{cases}$$

- $\delta = (\delta_1, \ldots, \delta_n) \in \mathbf{R}^n$ is the vector of parametric uncertainties composed of n real scalars δ_i.
- all uncertainties are normalized, i.e. $\delta_i \in [-1,\ 1]$.
- robustness test: determine whether the closed-loop model is stable $\forall \delta_i$ such that $|\delta_i| \leq 1$
- robustness margin: compute the largest value of k for which the closed-loop model remains stable $\forall \delta_i$ such that $|\delta_i| \leq k$

PARAMETRIC UNCERTAINTIES: example

Aircraft lateral model (linearized equations):

$$\begin{aligned} \dot{\beta} &= Y_\beta \beta + (Y_p + sin\alpha_0)p + (Y_r - cos\alpha_0)r + \tfrac{g}{V}\phi + Y_{\delta p}\delta p + Y_{\delta r}\delta r \\ \dot{p} &= L_\beta \beta + L_p p + L_r r + L_{\delta p}\delta p + L_{\delta r}\delta r \\ \dot{r} &= N_\beta \beta + N_p p + N_r r + N_{\delta r}\delta r \\ \dot{\phi} &= p + tan\theta_0\ r \end{aligned}$$

Parametric uncertainties are introduced in the 14 stability derivatives:

$$Y_\beta = (1+\delta_1)\,Y_\beta^0 \qquad Y_p = (1+\delta_2)\,Y_p^0 \qquad \ldots$$

Robustness test: does the model remains stable if the stability derivatives vary by $\pm 10\%$ around their nominal values, i.e. if $\delta_i \in [-0.1,\ 0.1]$?

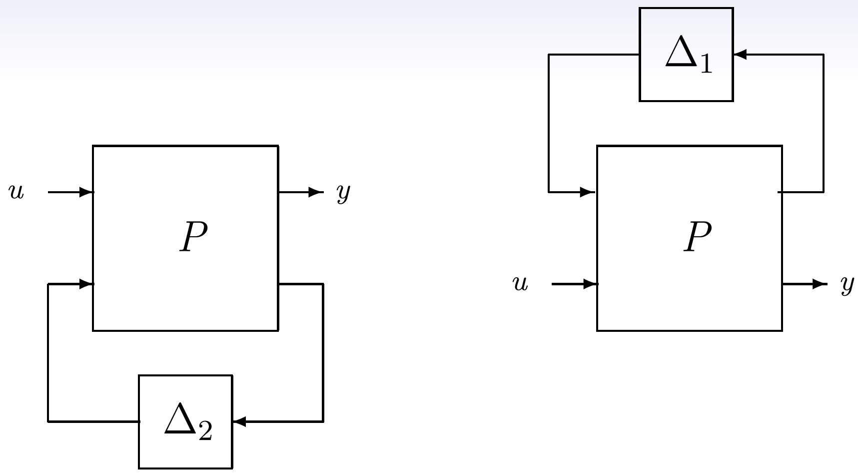
The uncertainties are normalized ($\delta_i \in [-1,\ 1] \Leftrightarrow$ variation of $\pm 10\%$):

$$Y_\beta = (1+0.1\delta_1)\,Y_\beta^0 \qquad Y_p = (1+0.1\delta_2)\,Y_p^0 \qquad \ldots$$

OUTLINE

1. Computation of a standard form
2. Introduction to μ-analysis
3. Computation of the structured singular value μ
4. Application to a passenger aircraft

LINEAR FRACTIONAL TRANSFORMATIONS (LFT)



Let $P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$. The lower and upper LFT are defined as:

$$\begin{aligned} F_l(P, \Delta_2) &= P_{11} + P_{12}\Delta_2(I - P_{22}\Delta_2)^{-1}P_{21} \\ F_u(P, \Delta_1) &= P_{22} + P_{21}\Delta_1(I - P_{11}\Delta_1)^{-1}P_{12} \end{aligned}$$

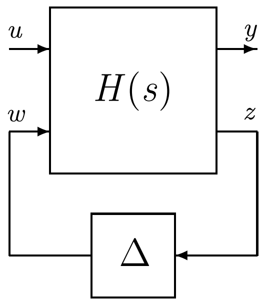
LFT MODELING: parametric uncertainties

Introduction of parametric uncertainties $\delta = (\delta_1, \ldots, \delta_n) \in \mathbf{R}^n$ in the state-space representation of the considered physical system:

$$\begin{aligned} \dot{x} &= A(\delta)x + B(\delta)u \\ y &= C(\delta)x + D(\delta)u \end{aligned}$$

LFT modeling:

$$y = G(s,\delta)u = F_l(H(s),\Delta)u \quad \text{where} \quad \Delta = \mathrm{diag}(\delta_1 I_{q_1}, \ldots, \delta_n I_{q_n})$$



$\Leftarrow$ nominal model (LTI)

$\Leftarrow$ structured uncertainties

LFT MODELING: particular case

The model depends on the uncertainties in an affine fashion:

$$\begin{aligned} \dot{x} &= (A_0 + \sum_i \delta_i A_i)x + (B_0 + \sum_i \delta_i B_i)u \\ y &= (C_0 + \sum_i \delta_i C_i)x + (D_0 + \sum_i \delta_i D_i)u \end{aligned}$$

In a more compact form:

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \left(P_0 + \sum_i \delta_i P_i\right) \begin{bmatrix} x \\ u \end{bmatrix} \quad \text{where} \quad P_i = \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix} \in \mathbf{R}^{l_1 \times l_2}$$

LFT modeling with $\Delta = \mathrm{diag}(\delta_1 I_{q_1}, \ldots, \delta_n I_{q_n})$:

$$P_0 + \sum_i \delta_i P_i = \mathcal{F}_l\left(\begin{bmatrix} \mathcal{H}_{11} & \mathcal{H}_{12} \\ \mathcal{H}_{21} & \mathcal{H}_{22} \end{bmatrix}, \Delta\right) = \mathcal{H}_{11} + \mathcal{H}_{12}\Delta(I - \mathcal{H}_{22}\Delta)^{-1}\mathcal{H}_{21}$$

LFT MODELING: particular case

Let $\mathcal{H}_{22} = 0$ and $\mathcal{H}_{11} = P_0$. Then $\sum_i \delta_i P_i = \mathcal{H}_{12}\Delta\mathcal{H}_{21}$.

Factorize $P_i = U_i V_i$ where $\begin{cases} U_i \in \mathbf{R}^{l_1 \times q_i} \\ V_i \in \mathbf{R}^{q_i \times l_2} \\ q_i = \text{rank}(P_i) \end{cases}$.

Then:

$$\begin{aligned} \sum_i \delta_i P_i &= \sum_i \delta_i U_i V_i \\ &= [U_1 \ \ldots \ U_n] \begin{bmatrix} \delta_1 I_{q_1} & & \\ & \ddots & \\ & & \delta_n I_{q_n} \end{bmatrix} \begin{bmatrix} V_1 \\ \vdots \\ V_n \end{bmatrix} \\ &= \mathcal{H}_{12}\Delta\mathcal{H}_{21} \end{aligned}$$

LFT MODELING: general case

- The elements of $A(\delta), B(\delta), C(\delta), D(\delta)$ must be polynomial or rational functions of the parametric uncertainties δ_i.
  - tabulated data $\Rightarrow$ polynomial/rational fitting (least squares. . .)
  - irrational functions $\Rightarrow$ approximating polynomial/rational functions (Taylor series. . .)

  Systematic methods exist to convert the model into LFT form once it is written in polynomial or rational form.

- Computing a minimal representation (i.e. with the smallest possible Δ matrix) is difficult.
  - always possible if $n = 1$
  - no systematic method if $n > 1$

Minimality is critical in terms of computational time and conservatism when robustness analysis tools are applied.

LFT MODELING: example

Let $y = [\delta_1\delta_2 + \delta_2]u$.

1. Without any preprocessing, $y = F_l(H_1, \Delta_1)u$ where:

$$H_1 = \left[\begin{array}{c|ccc} 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array}\right] \quad \text{and} \quad \Delta_1 = \text{diag}(\delta_1, \delta_2, \delta_2)$$

2. By factorizing $y = [(\delta_1 + 1)\delta_2]u$, $y = F_l(H_2, \Delta_2)u$ where:

$$H_2 = \left[\begin{array}{c|cc} 0 & 1 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 0 & 0 \end{array}\right] \quad \text{and} \quad \Delta_2 = \text{diag}(\delta_1, \delta_2)$$

Although this is not an exact rule, the trend is as follows: the fewer the occurrences of each δ_i in $A(\delta), B(\delta), C(\delta), D(\delta)$, the smaller the size of Δ.

LFT MODELING: example

Let us consider the following second-order model:

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\omega^2 & -2\xi\omega \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \omega^2 \end{bmatrix} u \\ y &= x_1 \end{aligned}$$

The frequency is assumed to be uncertain, i.e. $\omega = (1 + \delta)\omega_0$.

The most compact representation is:

$$y = \mathcal{F}_l(H(s), \Delta)u$$

where:

$$\Delta = \text{diag}(\delta, \delta)$$

LFT MODELING: a direct method

Factorization of ω:

$$\dot{x}_2 = \omega(\omega(u - x_1) - 2\xi x_2) \Leftrightarrow \begin{cases} \dot{x}_2 &= w_1 = \omega z_1 \\ z_1 &= \omega(u - x_1) - 2\xi x_2 = w_2 - 2\xi x_2 \\ w_2 &= \omega z_2 \\ z_2 &= u - x_1 \end{cases}$$

$\Rightarrow y = \mathcal{F}_l(\tilde{H}(s), \tilde{\Delta})u$ where $\tilde{\Delta} = \omega I_2$ and:

$$\tilde{H}(s) \leftrightarrow \begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w_1 \\ w_2 \end{bmatrix} \\ \begin{bmatrix} y \\ z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -2\xi \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ w_1 \\ w_2 \end{bmatrix} \end{cases}$$

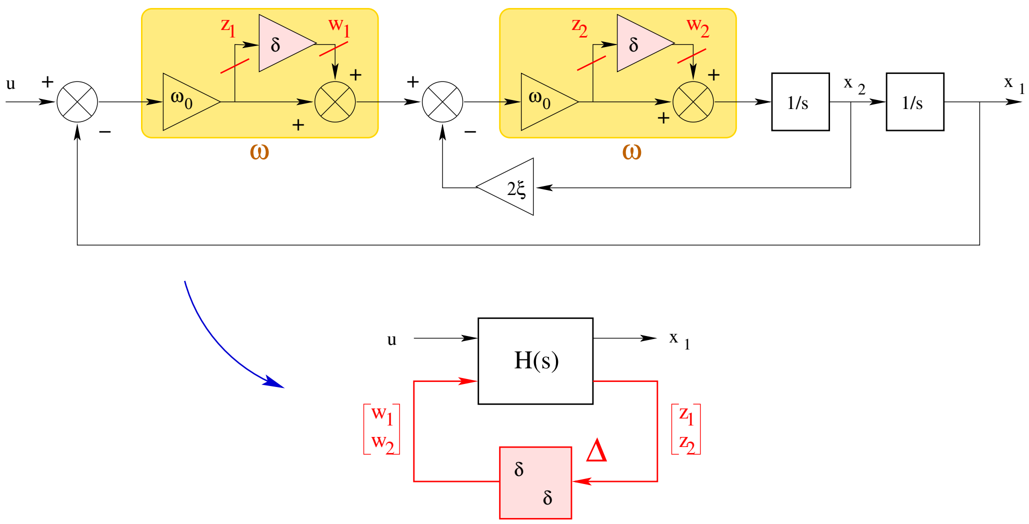
Normalization of ω:

$\tilde{\Delta} = \mathcal{F}_l(T, \Delta)$ where $T = \begin{bmatrix} \omega_0 I_2 & \sqrt{\omega_0} I_2 \\ \sqrt{\omega_0} I_2 & 0_2 \end{bmatrix}$ and $\Delta = \delta I_2$.

Finally, $y = \mathcal{F}_l(\tilde{H}(s), \mathcal{F}_l(T, \Delta))u = \mathcal{F}_l(H(s), \Delta)u$, where $H(s) = \mathcal{R}(\tilde{H}(s), T)$ and $\mathcal{R}(.)$ is the Redheffer star product.

LFT MODELING: a graphical method

Creation of a block diagram from: $\begin{cases} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \omega(\omega(u - x_1) - 2\xi x_2) \\ \omega &= (1 + \delta)\omega_0 \end{cases}$



LFT MODELING: Matlab tools

Matlab tools are implemented in the Generalized State-Space (GSS) Library of the Systems Modeling Analysis and Control (SMAC) Toolbox.

Main features:

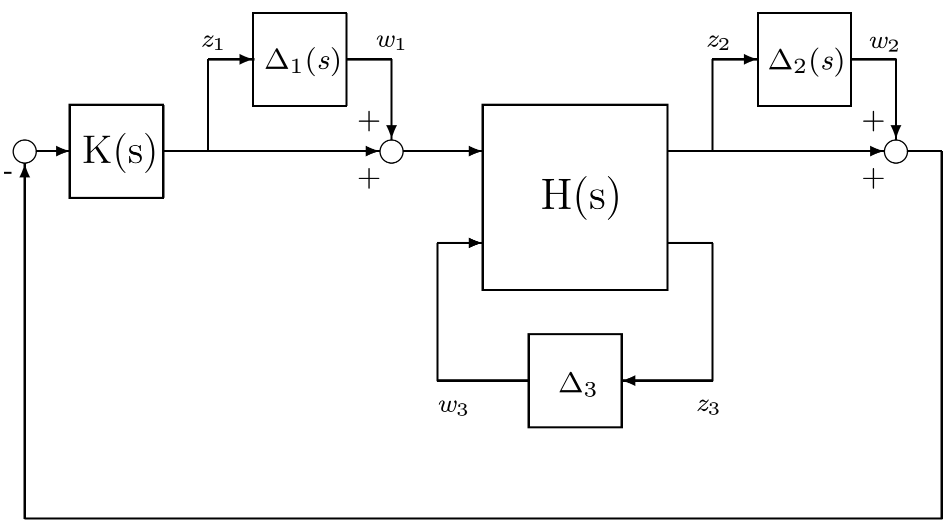
- automatic generation of an LFT:
  - from a numerical description
  - from a symbolic description
  - from an interconnection of LFT created with Simulink
- reduction of an LFT to tend towards minimality

Available online: *http://w3.onera.fr/smac/gss*

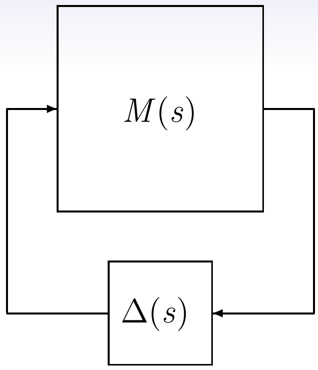
Remark: some tools also exist in the Robust Control Toolbox (uss object) but results are usually better with the GSS Library.

LFT MODELING: neglected dynamics

Neglected dynamics can be added: $\Delta_1(s)$ and $\Delta_2(s)$ are stable unstructured transfer functions.



COMPUTATION OF THE STANDARD FORM



$M(s)$ corresponds to the nominal closed-loop model. It includes both the nominal open-loop model $H(s)$ and the controller $K(s)$.

$\Delta(s)$ contains neglected dynamics & real parametric uncertainties:

$$\Delta(s) = \mathrm{diag}(\Delta_1(s), \ldots, \Delta_m(s), \delta_1 I_{q_1}, \ldots, \delta_n I_{q_n})$$

Both $M(s)$ and $\Delta(s)$ are assumed to be stable.

OUTLINE

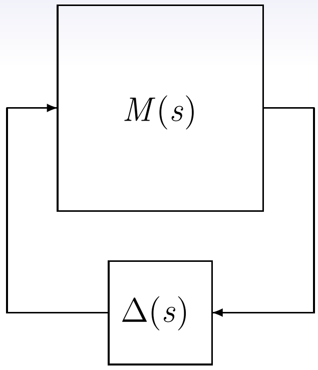
1 Computation of a standard form

2 Introduction to μ-analysis

3 Computation of the structured singular value μ

4 Application to a passenger aircraft

INTRODUCTION TO μ-ANALYSIS



Robustness analysis

Compute the values of $\Delta(s)$ for which the interconnection is stable.

μ-analysis is the most efficient technique to address this issue.

Limitation: only time-invariant uncertainties (nonlinearities and time-varying parameters $\Rightarrow$ IQC-based or Lyapunov-based analysis).

STRUCTURED SINGULAR VALUE μ: particular case

$\rightarrow$ strictly proper system, i.e. $M(s) = C(sI - A)^{-1}B$
$\rightarrow$ only parametric uncertainties, i.e. $\Delta = \mathrm{diag}(\delta_1 I_{q_1}, \ldots, \delta_n I_{q_n})$

The interconnection $M(s) - \Delta$ is asymptotically stable iff all eigenvalues of $\mathbb{A}(\delta) = A + B\Delta C$ are inside the open left half plane.

Assumption: $M(s)$ is asymptotically stable.

Property: The roots of $\mathbb{A}(\delta)$ are continuous functions of δ. It is thus equivalent to compute the values of Δ for which:

1. the interconnection is stable,
2. the interconnection is at the limit of stability.

Limit of stability $\Leftrightarrow$ an eigenvalue λ_i of $\mathbb{A}(\delta)$ lies on the imaginary axis, i.e. $\lambda_i = j\omega \Leftrightarrow \psi(s,\delta) = \det(sI - \mathbb{A}(\delta)) = 0$ for $s = j\omega$.

STRUCTURED SINGULAR VALUE μ: particular case

$$\begin{aligned}\psi(s,\delta) &= \det(sI - A - B\Delta C)\\ &= \det(sI-A)\det(I-(sI-A)^{-1}B\Delta C)\\ &= \det(sI-A)\det(I-C(sI-A)^{-1}B\Delta)\\ &= \det(sI-A)\det(I-M(s)\Delta)\end{aligned}$$

$\Rightarrow$ Limit of stability at frequency ω: $\det(I - M(j\omega)\Delta) = 0$

Definitions:

- structure of the uncertainties: $\mathbf{\Delta} = \{\mathrm{diag}(\delta_1 I_{q_1}, \ldots, \delta_n I_{q_n}\}$
- associated unit ball: $B(\mathbf{\Delta}) = \{\Delta \in \mathbf{\Delta} : \overline{\sigma}(\Delta) \leq 1\}$ where $\overline{\sigma}(\Delta) = \max_i(|\delta_i|)$.

STRUCTURED SINGULAR VALUE μ: main definitions

Structured singular value μ

The size of the smallest perturbation $\Delta \in \mathbf{\Delta}$ which brings a pole of the interconnection on the imaginary axis at frequency ω is:

$$k_m(\omega) = \min_{\Delta \in \mathbf{\Delta}} \{\overline{\sigma}(\Delta) : \det(I - M(j\omega)\Delta) = 0\}$$

The structured singular value is then defined as:

$$\mu_{\mathbf{\Delta}}(M(j\omega)) = \frac{1}{k_m(\omega)}$$

The structured singular value depends on both the nominal system $M(s)$ and the structure of the uncertainties $\mathbf{\Delta}$.

STRUCTURED SINGULAR VALUE μ: main definitions

Robustness margin k_r

The robustness margin is the size of the smallest perturbation $\Delta \in \mathbf{\Delta}$ which brings a pole of the interconnection on the imaginary axis:

$$k_r = \min_{\omega} k_m(\omega) = \min_{\omega} \frac{1}{\mu_{\mathbf{\Delta}}(M(j\omega))} = \frac{1}{\max_{\omega} \mu_{\mathbf{\Delta}}(M(j\omega))}$$

The robustness margin is thus:

- the largest k s.t. the interconnection is stable $\forall \Delta \in kB(\mathbf{\Delta})$.
- the smallest k for which $\exists \Delta^* \in kB(\mathbf{\Delta})$ s.t. the interconnection is **unstable**. Δ^* is the **smallest destabilizing perturbation**.

STRUCTURED SINGULAR VALUE μ: general case

$\rightarrow$ proper system, i.e. $M(s) = C(sI - A)^{-1}B + D \in \mathcal{RH}_\infty^{r \times r}$

$\rightarrow$ both neglected dynamics and parametric uncertainties, i.e.
$\Delta(s) = diag(\Delta_1(s), \dots, \Delta_m(s), \delta_1 I_{q_1}, \dots, \delta_n I_{q_n}) \in \mathcal{RH}_\infty^{r \times r}$

$\rightarrow$ $\mathbf{\Delta} = \{diag(\Delta_1, ..., \Delta_m, \delta_1 I_{q_1}, ..., \delta_n I_{q_n}) \in \mathbf{C}^{r \times r}, \Delta_i \in \mathbf{C}^{p_i \times p_i}, \delta_i \in \mathbf{R}\}$
notation: $\Delta(s) \in \mathbf{\Delta}$ means that $\Delta(j\omega) \in \mathbf{\Delta}$ for all $\omega \in \mathbf{R}$.

The definitions of $\mu_{\mathbf{\Delta}}(M(j\omega))$ and k_r are the same as before!

The robustness margin k_r is the largest k for which the interconnection $M(s) - \Delta(s)$ is stable for all $\Delta_i(s)$ and δ_i such that:

- $\|\Delta_i(s)\|_\infty \leq k$
- $|\delta_i| \leq k$

SMALL-GAIN THEOREM

Unstructured version

Let $M(s)$ be a stable LTI system. The interconnexion $M(s) - \Delta(s)$ is stable for all stable LTI uncertainty $\Delta(s) \in kB(\mathbf{C}^{r \times r})$ (i.e. such that $\|\Delta(s)\|_\infty \leq k$) iff:

$$\|M(s)\|_\infty \leq 1/k$$

where $\|M(s)\|_\infty = \sup_\omega \overline{\sigma}(M(j\omega))$.

Structured version

Let $M(s)$ be a stable LTI system. The interconnexion $M(s) - \Delta(s)$ is stable for all stable LTI uncertainty $\Delta(s) \in kB(\mathbf{\Delta})$ iff:

$$\|M(s)\|_\mu \leq 1/k$$

where $\|M(s)\|_\mu = \max_\omega \mu_{\mathbf{\Delta}}(M(j\omega))$. Note that $\|.\|_\mu$ is not a norm!

STRUCTURED/UNSTRUCTURED UNCERTAINTIES

Example: $M = \begin{bmatrix} a & a \\ b & b \end{bmatrix}$, where $a, b \in \mathbf{R}$.

- if $\mathbf{\Delta} = \mathbf{C}^{2\times 2}$ (unstructured uncertainty): $\overline{\sigma}(M) = \overline{\lambda}(M^* M) = \sqrt{2(a^2+b^2)}$
- if $\mathbf{\Delta} = \{\delta I_2, \delta \in \mathbf{R}\}$ (structured uncertainty): $\det(I - M\Delta) = 1 - (a+b)\delta \Rightarrow \mu_{\mathbf{\Delta}}(M) = |a+b|$

Note that $\overline{\sigma}(M)^2 - \mu_{\mathbf{\Delta}}(M)^2 = a^2 + b^2 - 2ab = (a-b)^2 \geq 0$, which means that $\overline{\sigma}(M) \geq \mu_{\mathbf{\Delta}}(M)$ for all $a, b \in \mathbf{R}$.

In particular, for $a = 1$ and $b = -1$:

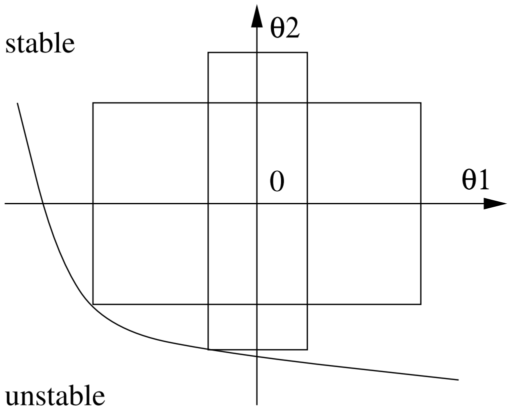
- $k_r = 0.5$ if $\mathbf{\Delta} = \mathbf{C}^{2\times 2}$
- $k_r = +\infty$ if $\mathbf{\Delta} = \{\delta I_2, \delta \in \mathbf{R}\}$

The gap between $\overline{\sigma}(M)$ and $\mu_{\mathbf{\Delta}}(M)$ can be infinite!

INFLUENCE OF THE WEIGHTING FACTORS

An hypercube in the space of the normalized uncertainties is equivalent to an hyperrectangle in the space of the uncertain parameters.

$$\begin{aligned}\theta_1 &= (1 + x_1\delta_1)\theta_1^0 \\ \theta_2 &= (1 + x_2\delta_2)\theta_2^0\end{aligned}$$



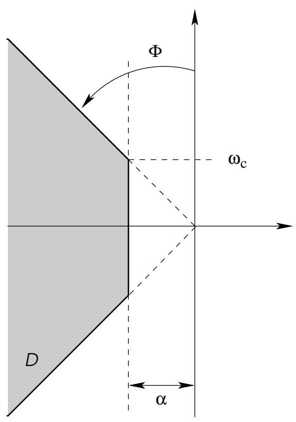
EXTENSIONS: robust modal performance

In practice, it is desirable to quantify the performance degradation, which is induced by uncertainties and appears before instability.

Objective: check whether a minimum stability degree α and a minimum damping ratio sin(ϕ) are guaranteed $\forall \Delta(s) \in B(\mathbf{\Delta})$.

Solution: Compute the structured singular value along the boundary $\partial\mathcal{D}$ of the grey region $\mathcal{D}$ instead of the imaginary axis.

Assumption: all poles of $M(s)$ belong to $\mathcal{D}$.



If $\max_{s\in\partial\mathcal{D}} \mu_{\mathbf{\Delta}}(M(s)) < 1$, all poles of $M(s) - \Delta(s)$ remain inside the grey region $\forall \Delta(s) \in B(\mathbf{\Delta})$.

EXTENSIONS: skew-μ analysis

Assume that $\mathbf{\Delta} = \mathrm{diag}(\mathbf{\Delta}_1, \mathbf{\Delta}_2)$ is split into two distinct block structures and let $\mathbf{\Delta}_s = \mathrm{diag}(B(\mathbf{\Delta}_1), \mathbf{\Delta}_2) \Rightarrow \Delta_1 \in B(\mathbf{\Delta}_1)$ and $\Delta_2 \in \mathbf{\Delta}_2$ are fixed range and unbounded uncertainties respectively.

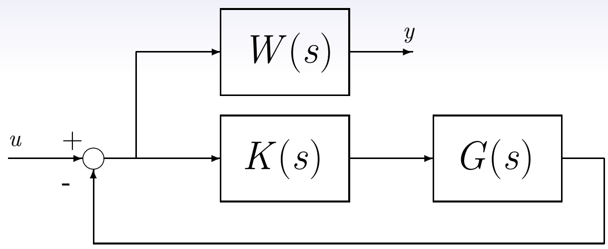
The skewed structured singular value $\nu_{\mathbf{\Delta}_s}(M(j\omega))$ is the inverse of the norm of the smallest $\Delta_2 \in \mathbf{\Delta}_2$ for which there exists $\Delta_1 \in B(\mathbf{\Delta}_1)$ such that $\det(I - M(j\omega)\mathrm{diag}(\Delta_1, \Delta_2)) = 0$.

The interconnection $M(s)$–$\Delta(s)$ is stable $\forall \Delta(s) \in \mathrm{diag}(B(\mathbf{\Delta}_1), k_s B(\mathbf{\Delta}_2))$ where the skewed robustness margin k_s is defined as:

$$k_s = \left[\max_{\omega} \nu_{\mathbf{\Delta}_s}(M(j\omega)) \right]^{-1}$$

Example: The uncertainty on the mass of a given system is $\pm 10\%$. What is the largest uncertainty on the inertia for which stability is guaranteed?

EXTENSIONS: nominal H_∞ performance

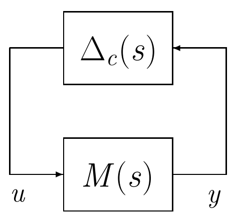


$y = W(s)S(s)u = M(s)u$ where $S(s) = \dfrac{1}{1 + G(s)K(s)}$ is the sensitivity function and $W(s)$ is a weighting function.

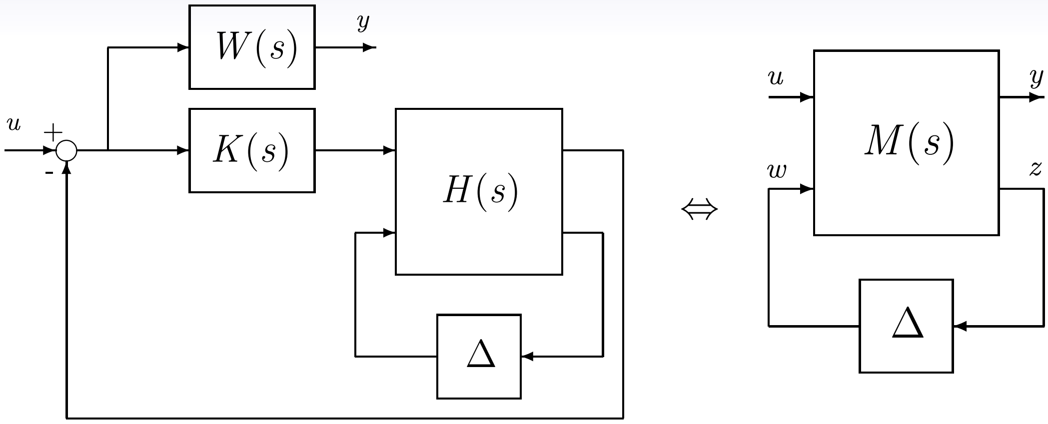
The nominal performance level is $\gamma_{nom} = \|M(s)\|_\infty = \sup_\omega \overline{\sigma}(M(j\omega))$.

This is equivalent to computing k_r for the interconnection $M(s) - \Delta_c(s)$ where $\Delta_c(s) \in \mathbf{\Delta} = \mathbf{C}^{k \times k}$, since:

$$\mu_{\mathbf{C}^{k\times k}}(M(j\omega)) = \overline{\sigma}(M(j\omega))$$



EXTENSIONS: worst-case H_∞ performance



Objective: Compute the highest value of the H_∞ norm of the transfer from u to y when Δ takes all possible values in $B(\mathbf{\Delta})$.

EXTENSIONS: worst-case H_∞ performance

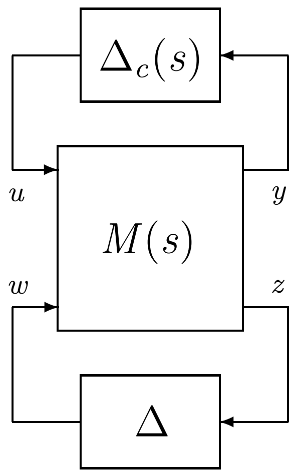
The worst-case performance level is $\gamma = \max\limits_{\Delta \in B(\mathbf{\Delta})} \|\mathcal{F}_l(M(s), \Delta)\|_\infty$.

Computing γ is a skew-μ problem.

It is equivalent to computing k_s for the interconnection $M(s) - \tilde{\Delta}(s)$ where $\tilde{\Delta}(s) = \mathrm{diag}(\Delta, \Delta_c(s))$.

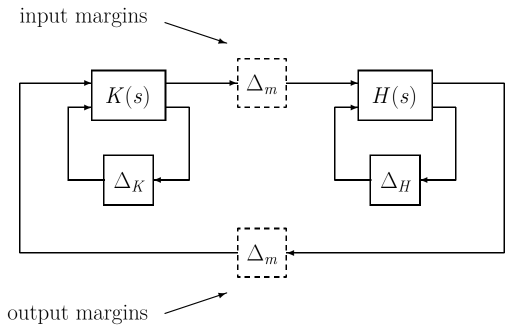
$\rightarrow \Delta \in \mathbf{\Delta} = \mathbf{\Delta}_1$ fixed range

$\rightarrow \Delta_c(s) \in \mathbf{C}^{k \times k} = \mathbf{\Delta}_2$ unbounded



EXTENSIONS: worst-case input-output margins

Objective: Compute the worst-case gain, modulus, phase and delay margins, i.e. the highest value of the gain, the phase shift or the time delay that can be inserted without destabilizing the closed-loop system when $\Delta = \text{diag}(\Delta_H, \Delta_K)$ takes all possible values in $B(\mathbf{\Delta})$.



$\Rightarrow$ input or output margins

$\Rightarrow$ SISO or MIMO margins

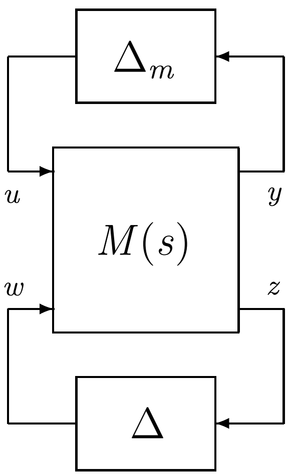
- SISO: $\Delta_m = \text{diag}(I, \delta_{m,i}, I)$
- MIMO: $\Delta_m = \text{diag}(\delta_{m,1}, \ldots, \delta_{m,k})$

- gain margin: $\delta_{m,i} = 1 + \delta_i$, $\delta_i \in \mathbf{R}$
- modulus margin: $\delta_{m,i} = 1 + \delta_i$, $\delta_i \in \mathbf{C}$
- phase margin: $\delta_{m,i} = e^{j\phi_i}$, $\phi_i \in \mathbf{R}$
- time-delay margin: $\delta_{m,i} = e^{-\tau_i s}$, $\tau_i \in \mathbf{R}_+$

EXTENSIONS: worst-case input-output margins

Non-rational elements are replaced by rational functions to get an LFR:

- phase margin: $e^{j\phi_i}$ replaced by $\frac{1-j\delta_i}{1+j\delta_i}$, $\delta_i \in \mathbf{R}$ (bilinear transformation)
- time-delay margin: $e^{-\tau_i s}$ replaced by $f(\delta_i)$, $\delta_i \in \mathbf{R}$ (more complicated)



Computing the worst-case input-output margins is a skew-μ problem.

It is equivalent to computing k_s for the interconnection $M(s) - \tilde{\Delta}$ where $\tilde{\Delta} = \text{diag}(\Delta, \Delta_m)$.

$\rightarrow \Delta \in \mathbf{\Delta} = \mathbf{\Delta}_1$ fixed range

$\rightarrow \Delta_m \in \{\text{diag}(\delta_1, \ldots, \delta_k)\} = \mathbf{\Delta}_2$ unbounded

μ-ANALYSIS: brief summary

- Objective: evaluate the stability and the performance properties of an LTI system in the presence of structured LTI uncertainties (parametric uncertainties and neglected dynamics).
- Key idea: computing the smallest destabilizing perturbation is equivalent to computing the smallest perturbation which brings a pole of the closed-loop interconnection $M(s) - \Delta(s)$ on the imaginary axis ($\Leftrightarrow$ limit of stability $\Leftrightarrow \det(I - M(j\omega)\Delta) = 0$).
- Frequency-domain approach: the structured singular value $\mu_{\boldsymbol{\Delta}}(M(j\omega))$ is computed as a function of ω, and the robustness margin k_r is obtained as the inverse of the peak value.
- Several extensions: modal performance, skew-μ analysis, worst-case H_∞ performance, worst-case gain/phase/modulus/delay margins. . .

OUTLINE

1 Computation of a standard form

2 Introduction to μ-analysis

3 Computation of the structured singular value μ

4 Application to a passenger aircraft

COMPUTATION OF μ AT A SINGLE FREQUENCY

The frequency ω is fixed. Let $M = M(j\omega)$ and $\Delta = \Delta(j\omega)$.

- If $\mathbf{\Delta} = \mathbf{C}^{p\times p}$ (one full complex block):
  $\rightarrow$ unstructured uncertainty $\Rightarrow$ standard small-gain theorem
  $$\mu_{\mathbf{\Delta}}(M) = \overline{\sigma}(M) \text{ and } k_r = 1/\|M(s)\|_\infty$$
- If $\mathbf{\Delta} = \{\delta I_q, \delta \in \mathbf{R}\}$ (one repeated real scalar):
  $\rightarrow \det(I - \delta M) = 0 \Leftrightarrow (I - \delta M)x = 0 \Leftrightarrow Mx = \frac{1}{\delta}x$
  $\rightarrow$ smallest destabilizing value of $\delta \Leftrightarrow$ largest real eigenvalue of M
  $$\mu_{\mathbf{\Delta}}(M) = \rho_R(M)$$
  $\rho_R(M) = \max\{|\lambda|: \lambda \in \mathbf{R}, \lambda$ eigenvalue of $M\} =$ real spectral radius
- In the general case, computing $\mu_{\mathbf{\Delta}}(M)$ is NP hard, so both lower and upper bounds are computed instead.
  $$\{\delta I_r, \delta \in \mathbf{R}\} \subset \mathbf{\Delta} \subset \mathbf{C}^{r\times r} \Rightarrow \rho_R(M) \le \mu_{\mathbf{\Delta}}(M) \le \overline{\sigma}(M)$$

COMPUTATION OF μ AT A SINGLE FREQUENCY

Example: $M = \begin{bmatrix} a & a \\ b & b \end{bmatrix}$, $a, b \in \mathbf{R}$, $\mathbf{\Delta} = \left\{\begin{bmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{bmatrix}, \delta_i \in \mathbf{R}\right\}$.

$$\rho_R(M) = |a+b| \le \mu_{\mathbf{\Delta}}(M) \le \sqrt{2(a^2+b^2)} = \overline{\sigma}(M)$$

In particular, if $a = 1$ and $b = -1$:

$$0 \le \mu_{\mathbf{\Delta}}(M) \le 2$$

The gap between the bounds can be very large!

For $\Delta \in \mathbf{\Delta}$, $\det(I - M\Delta) = 1 - a\delta_1 - b\delta_2 \Rightarrow \mu_{\mathbf{\Delta}}(M) = |a| + |b|$.

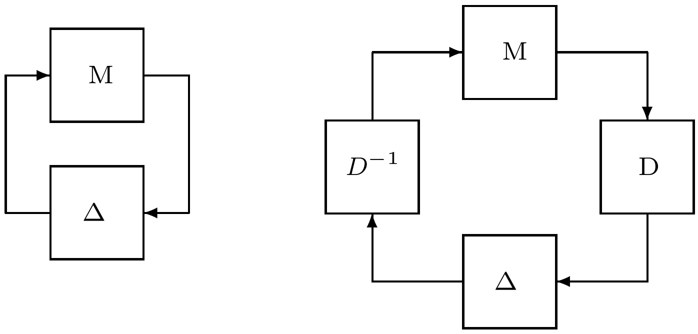
If $a = 1$ and $b = -1$, $\mu_{\mathbf{\Delta}}(M) = \overline{\sigma}(M) = 2$.

μ UPPER BOUND: complex uncertainties

Let $D = D^* > 0$ which commutes with any $\Delta \in \mathbf{\Delta}$, i.e. $D\Delta = \Delta D$.

- $\mu_{\mathbf{\Delta}}(M) = \mu_{\mathbf{\Delta}}(DMD^{-1})$
  $\rightarrow \det(I - D^{-1}MD\Delta) = \det(I - MD\Delta D^{-1}) = \det(I - M\Delta)$
- $\mu_{\mathbf{\Delta}}(DMD^{-1}) \leq \overline{\sigma}(DMD^{-1})$
  $\rightarrow \mu$ upper bound

$$\mu_{\mathbf{\Delta}}(M) \leq \overline{\sigma}(DMD^{-1})$$



μ UPPER BOUND: complex uncertainties

Complex uncertainties $\Rightarrow \mathbf{\Delta} = \{\mathrm{diag}(\Delta_1, \ldots, \Delta_m) \in \mathbf{C}^{r \times r}, \Delta_i \in \mathbf{C}^{p_i \times p_i}\}$

Let $\mathcal{D}$ be the set of all positive definite hermitian matrices which commute with any $\Delta \in \mathbf{\Delta}$. Then:

$$\mathcal{D} = \left\{\mathrm{diag}(d_1 I_{p_1}, \ldots, d_m I_{p_m}) \in \mathbf{C}^{r \times r}, d_i > 0\right\}$$

A μ upper bound is obtained by solving the optimization problem:

$$\mu_{\mathbf{\Delta}}(M) \leq \inf_{D \in \mathcal{D}} \overline{\sigma}(DMD^{-1}) \leq \overline{\sigma}(M)$$

This optimization problem is convex:

$\overline{\sigma}(DMD^{-1}) < \alpha \Leftrightarrow (D^{-*}M^*D^*)(DMD^{-1}) < \alpha^2 I_r$ (definition of $\overline{\sigma}$)
$\Leftrightarrow M^*D^*DM < \alpha^2 D^*D$ (congruence)
$\mathbf{D} = D^*D \in \mathcal{D} \Rightarrow$ if $\exists \mathbf{D} \in \mathcal{D}$ such that $M^*\mathbf{D}M < \alpha^2 \mathbf{D}$, then $\mu_{\mathbf{\Delta}}(M) < \alpha$.
Minimizing α is a generalized eigenvalue problem (convex) $\Rightarrow$ global optimum can be obtained in polynomial time with an LMI solver.

μ UPPER BOUND: mixed real/complex uncertainties

Mixed real/complex uncertainties:

$\mathbf{\Delta} = \{\text{diag}(\Delta_1, \ldots, \Delta_m, \delta_1 I_{q_1}, \ldots, \delta_n I_{q_n}) \in \mathbf{C}^{r\times r}, \Delta_i \in \mathbf{C}^{p_i \times p_i}, \delta_i \in \mathbf{R}\}$

The previous μ upper bound still holds with:

$\mathcal{D} = \{\text{diag}(d_1 I_{p_1}, \ldots, d_m I_{p_m}, D_1, \ldots, D_n) \in \mathbf{C}^{r\times r}, d_i > 0, D_i = D_i^* > 0\}$

$D \in \mathcal{D}$ commutes with both real and complex scalars:
$\Rightarrow$ the real nature of the parametric uncertainties is not considered,
$\Rightarrow$ the complex μ upper bound can be very pessimistic.

Let $\mathcal{G}$ be the set of all hermitian matrices such that $G\Delta = \Delta^* G$ for all $\Delta \in \mathbf{\Delta}$. Then:

$$\mathcal{G} = \left\{\text{diag}(0_{p_1}, \ldots, 0_{p_m}, G_1, \ldots, G_n) \in \mathbf{C}^{r\times r}, G_i = G_i^* \in \mathbf{C}^{q_i \times q_i}\right\}$$

$D \in \mathcal{D}$ and $G \in \mathcal{G}$ are called scaling matrices.

1. Standard form 2. μ-analysis 3. Computing μ 4. Applications

μ UPPER BOUND: mixed real/complex uncertainties

Let $\alpha > 0$. If there exist matrices $D \in \mathcal{D}$ and $G \in \mathcal{G}$ which satisfy one of the following relations:

$$\overline{\sigma}\left((I + G^2)^{-\frac{1}{4}}\left(\frac{DMD^{-1}}{\alpha} - jG\right)(I + G^2)^{-\frac{1}{4}}\right) \leq 1 \qquad (1)$$

$$M^* DM + j(GM - M^* G) \leq \alpha^2 D \qquad (2)$$

then $\mu_{\mathbf{\Delta}}(M) \leq \alpha$.

- The problem of minimizing α can be solved:
  $\rightarrow$ optimally using an LMI solver
  $\rightarrow$ faster but suboptimally using a gradient descent algorithm
- The complex μ upper bound is recovered if $G = 0 \Rightarrow$ the mixed μ upper bound is less pessimistic.

μ UPPER BOUND: mixed real/complex uncertainties

The gap between $\mu_{\mathbf{\Delta}}(M)$ and its upper bound can be large, but is usually much smaller than with the crude bound $\overline{\sigma}(M)$.

Example: $$M = \begin{bmatrix} 0 & 1.9+0.2i & 01+0.1i & \\ 1+0.1i & 0 & 0.5-0.1i & 0 \\ 2.1-0.2i & 0 & 1+0.1i & 0 \\ 0 & -2-0.3i & 0 & -1-0.1i \end{bmatrix}$$

$$\mathbf{\Delta} = \left\{ \begin{bmatrix} \delta_1 I_2 & 0 \\ 0 & \delta_2 I_2 \end{bmatrix}, \delta_i \in \mathbf{R} \right\}$$

- crude upper bound ($\overline{\sigma}(M)$): 3.12
- complex upper bound (D scaling only): 2.42
- mixed upper bound (D, G scalings): 1.77

If $2n+m \leq 3$, the mixed μ upper bound is exactly equal to $\mu_{\mathbf{\Delta}}(M)$.

μ UPPER BOUND: mixed real/complex uncertainties

In case of highly repeated real scalars, i.e. $\delta_i I_{q_i}$ with $q_i \gg 1$:

- the number of decision variables in D_i, G_i can be large $(2q_i^2)$
  $\Rightarrow$ high computational time
- the gap between $\mu_{\mathbf{\Delta}}(M)$ and its upper bound can be large
  $\Rightarrow$ pessimistic result

Importance of a simple (and possibly minimal) LFT model!

Example: $y = [1 - \delta_1^2\delta_2^2 + \delta_1^2 + 2\delta_1\delta_2^2 - 2\delta_1 - \delta_2^2]^{-1}\, u$

1. no preprocessing $\Rightarrow \mathbf{\Delta} = \{\mathrm{diag}(\delta_1 I_6, \delta_2 I_6), \delta_i \in \mathbf{R}\}$
2. $y = [(1-\delta_1)^2(1-\delta_2^2)]^{-1}\, u \Rightarrow \mathbf{\Delta} = \{\mathrm{diag}(\delta_1 I_2, \delta_2 I_2), \delta_i \in \mathbf{R}\}$

	LMI solver		gradient algorithm	
	bound	CPU time	bound	CPU time
model 1	1.02	20.22 s	1.16	0.16 s
model 2	1.00	0.17 s	1.00	0.05 s

μ LOWER BOUND: mixed real/complex uncertainties

1. $\mu_{\mathbf{\Delta}}(M) = \max_{\Delta \in B(\mathbf{\Delta})} \rho_R(\Delta M)$

   → choose some matrices $\tilde{\Delta}_1, \ldots, \tilde{\Delta}_K$ in $B(\mathbf{\Delta})$

   → a μ lower bound is obtained as $\max_{k=1\ldots K} \rho_R(\tilde{\Delta}_k M)$

   → simple but not very efficient method (2^n values must be considered just to test the vertices of $B(\mathbf{\Delta})$)

2. $\mu_{\mathbf{\Delta}}(M) = \max_{Q \in \mathcal{Q}} \rho_R(QM)$ where $\mathcal{Q}$ is the set of all $\Delta \in B(\mathbf{\Delta})$ such that $\Delta_i^* \Delta_i = I_{p_i}$

   → non-concave problem ⇒ a local maximum (μ lower bound) is computed using an iterative procedure (power algorithm)

   → very fast and accurate method except for purely real uncertainties (convergence problems ⇒ other methods exist)

COMPUTATION OF $\mu_{\mathbf{\Delta}}(M)$: summary

- no analytical expression of $\mu_{\mathbf{\Delta}}(M)$ in the general case
- computing $\mu_{\mathbf{\Delta}}(M)$ is NP hard in the general case
  ⇒ computation of upper and lower bounds
  ⇒ evaluation of the gap between the bounds
- upper bound: convex optimization tools (LMI solver) to compute the global minimum of a convex optimization problem
- lower bound: heuristics to compute a local maximum of a non-concave optimization problem
- the described methods are implemented in the function **mussv** of the Robust Control Toolbox of Matlab
- several other methods exist:
  → specific algorithms for real, complex or mixed uncertainties
  → polynomial-time or exponential-time algorithms (the latter can be useful for problems with only few uncertainties)
  → tradeoff between accuracy and computational time

COMPUTATION OF k_r: standard approach

The exact computation of k_r or k_s is NP hard in the general case, so both lower and upper bounds are computed instead.

But even computing these bounds is a challenging problem with an infinite number of frequency-domain constraints.

It is usually solved on a finite frequency grid $(\omega_i)_{i\in[1,M]}$ and estimates of the robust stability margins are then obtained as:

$$\frac{1}{\max\limits_{i\in[1,M]}(\overline{\mu}_{\mathbf{\Delta}}(M(j\omega_i)))} \leq k_r \leq \frac{1}{\max\limits_{i\in[1,M]}(\underline{\mu}_{\mathbf{\Delta}}(M(j\omega_i)))}$$

$$\frac{1}{\max\limits_{i\in[1,M]}(\overline{\nu}_{\mathbf{\Delta}_s}(M(j\omega_i)))} \leq k_s \leq \frac{1}{\max\limits_{i\in[1,M]}(\underline{\nu}_{\mathbf{\Delta}_s}(M(j\omega_i)))}$$

1. Standard form 2. μ-analysis 3. Computing μ 4. Applications

COMPUTATION OF k_r: standard approach

Problem: the grid must contain the most critical frequency point for which the maximal value of $\mu_{\mathbf{\Delta}}$ or $\nu_{\mathbf{\Delta}_s}$ is reached. If not:

- The upper bound on k_r or k_s can be very poor, notably for flexible systems, whose $\mu_{\mathbf{\Delta}}$ or $\nu_{\mathbf{\Delta}_s}$ plot often exhibits very high and narrow peaks.
- Even worse, the lower bound can be over-evaluated, *i.e.* be larger than the real value of k_r or k_s.

Unfortunately, the aforementioned critical frequency is usually unknown!

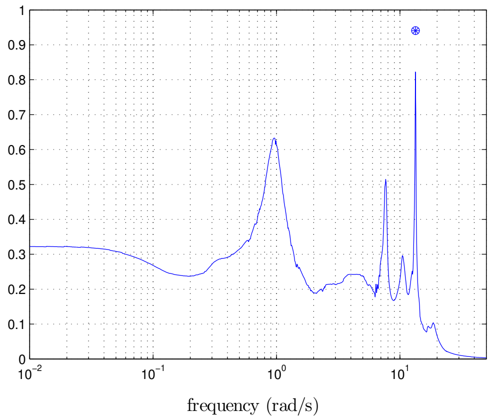
In this context, the considered frequency grid must be sufficiently dense, which can lead to a prohibitive computational cost.

But even so, it is still possible to miss a critical frequency (especially for purely real uncertainties, since $\omega \rightarrow \mu_{\mathbf{\Delta}}(M(j\omega))$ can be discontinuous)...

COMPUTATION OF k_r: standard approach

Example: aeronautical system with badly-damped flexible modes

- μ upper bounds are computed on a very fine frequency grid (1000 points) $\Rightarrow$ the largest value is 0.82.
- a μ lower bound of 0.94 and a destabilizing value of Δ are obtained.



A sharp peak is missed although the frequency grid is very dense.

COMPUTATION OF k_r: standard approach

Example: $M = \begin{bmatrix} 1 & m \\ -m & 1 \end{bmatrix}$, $m \in \mathbf{R}$, $\mathbf{\Delta} = \{\delta I_2, \delta \in \mathbf{R}\}$.

$$\det(I - M\Delta) = (1-\delta)^2 + m^2\delta^2$$

$$\Rightarrow \begin{cases} \mu_{\mathbf{\Delta}}(M) = 0 \text{ if } m \neq 0 \\ \mu_{\mathbf{\Delta}}(M) = 1 \text{ if } m = 0 \end{cases}$$

The functions $M \to \mu_{\mathbf{\Delta}}(M)$ and $\omega \to \mu_{\mathbf{\Delta}}(M(j\omega))$, can be discontinuous if $\mathbf{\Delta} \subset \mathbf{R}^{r\times r}$ (purely real uncertainties).

These functions are continuous in the presence of complex uncertainties.

LOWER BOUND ON k_r: improved approach

A μ upper bound α_i and matrices D_i, G_i are computed for a frequency ω_i.

$\alpha_i \leftarrow (1+\epsilon)\alpha_i$ is then slightly increased to enforce a strict inequality:

$$\overline{\sigma}\left((I+G_i^2)^{-1/4}\left(\frac{D_i M(\omega_i) D_i^{-1}}{\alpha_i} - jG_i\right)(I+G_i^2)^{-1/4}\right) < 1$$

The key step is to compute the largest frequency interval $I(\omega_i) \ni \omega_i$ s.t.:

$$\forall \omega \in I(\omega_i),\ \ \overline{\sigma}\left((I+G_i^2)^{-1/4}\left(\frac{D_i M(\omega) D_i^{-1}}{\alpha_i} - jG_i\right)(I+G_i^2)^{-1/4}\right) \leq 1$$

→ α_i is a guaranteed μ upper bound on the whole frequency interval $I(\omega_i)$, and not only for the single frequency ω_i.

→ an iterative procedure is applied until $\cup I(\omega_i) = \mathbf{R} \Rightarrow [\max_i \alpha_i]^{-1}$ is a lower bound on k_r, i.e. a guaranteed robustness margin.

UPPER BOUND ON k_r: improved approach

Main features of standard methods (e.g. power algorithm):

+ constructive heuristics which compute worst-case uncertainties,
− frequency is fixed $\Rightarrow$ worst-cases can be missed even with a fine grid,
+ fast and accurate results in the complex and mixed cases,
− convergence problems in the purely real case (lower bound equal to 0).

Key idea of the improved method: to obtain in the purely real case a tight μ lower bound over the whole frequency range rather than at a fixed frequency.

→ first search a perturbation Δ which brings one pole of the system near a chosen frequency point on the imaginary axis (good initial guess),

→ then consider Δ as a fictitious feedback gain allowing to move this pole freely through the imaginary axis to obtain a destabilizing perturbation.

UPPER BOUND ON k_r: improved approach

2-step procedure at each point ω_i of a rough frequency grid:

1. apply the power algorithm to a regularized μ problem obtained by adding a small amount ϵ of complex uncertainties Δ_C to the real uncertainties Δ:

$$M_{reg}(j\omega_i) = \begin{bmatrix} M(j\omega_i) & \sqrt{\epsilon}M(j\omega_i) \\ \sqrt{\epsilon}M(j\omega_i) & \epsilon M(j\omega_i) \end{bmatrix} \quad \text{and} \quad \Delta_{reg} = \mathrm{diag}(\Delta, \Delta_C)$$

2. extract the real part Δ^* of the resulting worst-case uncertainty ; compute $\tilde{\Delta} \in \mathbf{\Delta}$ which minimizes $\overline{\sigma}\left(\Delta^* + \tilde{\Delta}\right)$ and moves one pole of $M(s) - (\Delta^* + \tilde{\Delta})$ through the imaginary axis.

- Step 2 can be recast as a linear programming problem $\Rightarrow$ very fast.
- The imaginary axis is crossed at a point $j\tilde{\omega}_i \neq j\omega_i$, which usually corresponds to a peak value on the μ plot $\Rightarrow$ tight μ lower bound and destabilizing uncertainty, *i.e.* tight upper bound on k_r.

UPPER & LOWER BOUND ON k_r: improved approach

The aforementioned algorithms have been implemented in the Systems Modeling Analysis and Control (SMAC) Toolbox for Matlab.

Available online: *http://w3.onera.fr/smac/smart*

Brief summary

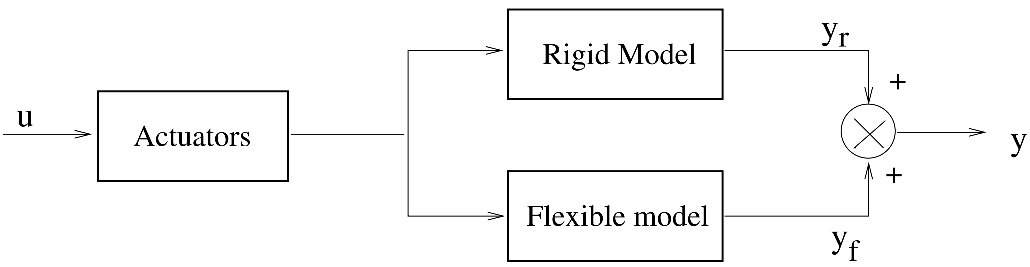
- A lower bound on k_r (μ upper bound) provides a guaranteed robustness margin.
- An upper bound on k_r (μ lower bound) provides a destabilizing uncertainty and measures the conservatism of the lower bound.

OUTLINE

1. Computation of a standard form
2. Introduction to μ-analysis
3. Computation of the structured singular value μ
4. Application to a passenger aircraft

PASSENGER AIRCRAFT: open-loop model

The open-loop plant is composed of both a rigid and a flexible part:



Rigid model: linearized lateral model with uncertainties in the 14 stability derivatives.

$$
\begin{aligned}
\dot{\beta} &= Y_\beta \beta + (Y_p + sin\alpha_0)p + (Y_r - cos\alpha_0)r + \tfrac{g}{V}\phi + Y_{\delta p}\delta p + Y_{\delta r}\delta r \\
\dot{p} &= L_\beta \beta + L_p p + L_r r + L_{\delta p}\delta p + L_{\delta r}\delta r \\
\dot{r} &= N_\beta \beta + N_p p + N_r r + N_{\delta r}\delta r \\
\dot{\phi} &= p + tan\theta_0\ r
\end{aligned}
$$

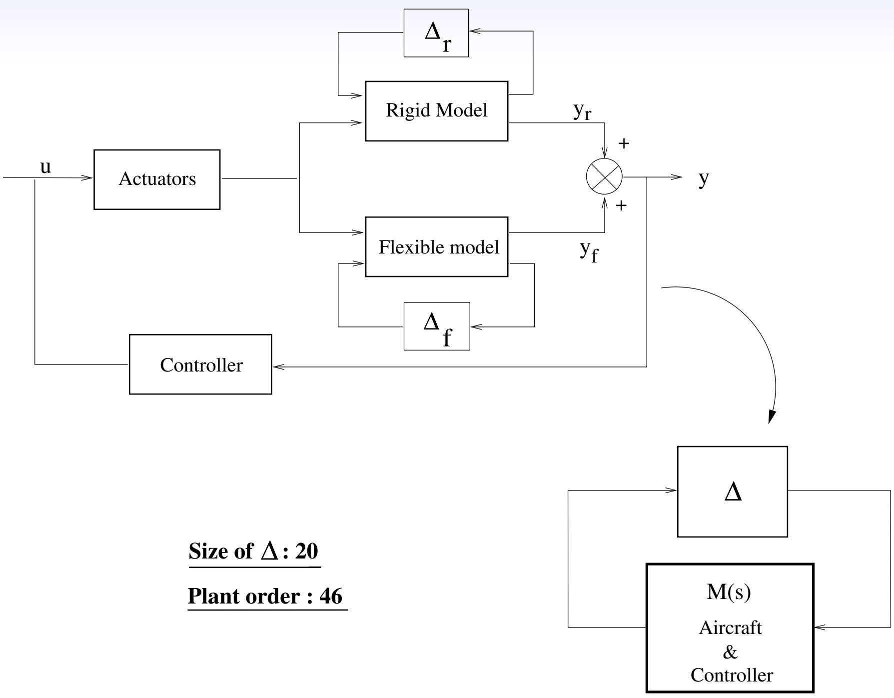
PASSENGER AIRCRAFT: open-loop model

Flexible model: 6 poorly damped bending modes with uncertainties in the natural frequencies.

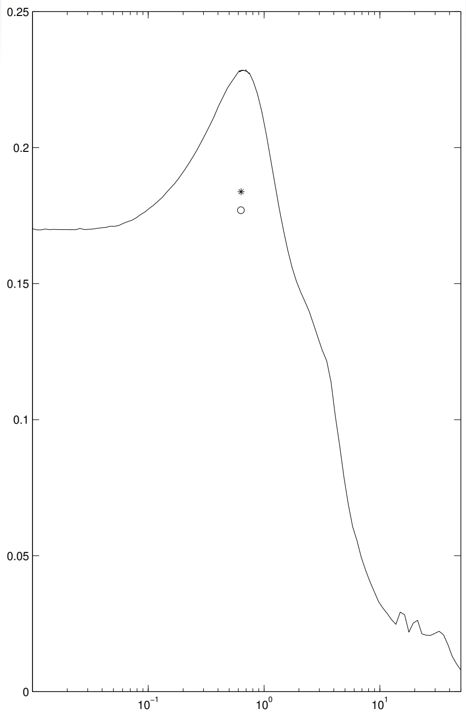
	damping ratio	natural frequency (rad/s)
1	$1.56\,10^{-2}$	14.3
2	$2.16\,10^{-2}$	13.5
3	$2.42\,10^{-2}$	12.5
4	$3.29\,10^{-2}$	7.35
5	$5.07\,10^{-2}$	14.1
6	$5.09\,10^{-2}$	8.62

Controller: observer-based state-feedback control law.

PASSENGER AIRCRAFT: closed-loop LFR



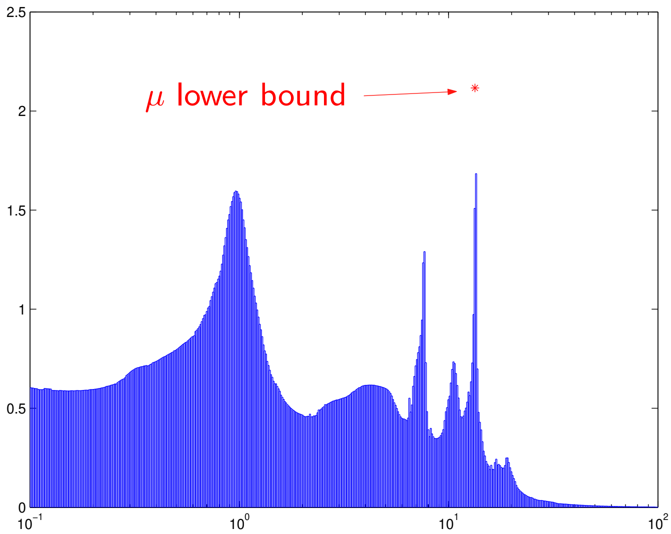
RIGID AIRCRAFT: standard μ upper bound



- Normalization: $\delta_i \in [-1,\ 1] \Rightarrow$ variation of $\pm$ 10% of the stability derivarives around nominal values.
- Analysis: 100 points within the interval $[0, 50\, rad/s]$.
- μ upper bound = 0.229 for $\omega = 0.70\, rad/s \Rightarrow$ stability guaranteed for variations of $\pm 10\%/0.229 \approx \pm 43.5\%$.
- μ lower bound = 0.184 for $\omega = 0.63\, rad/s \Rightarrow$ gap between the bounds = 19%.

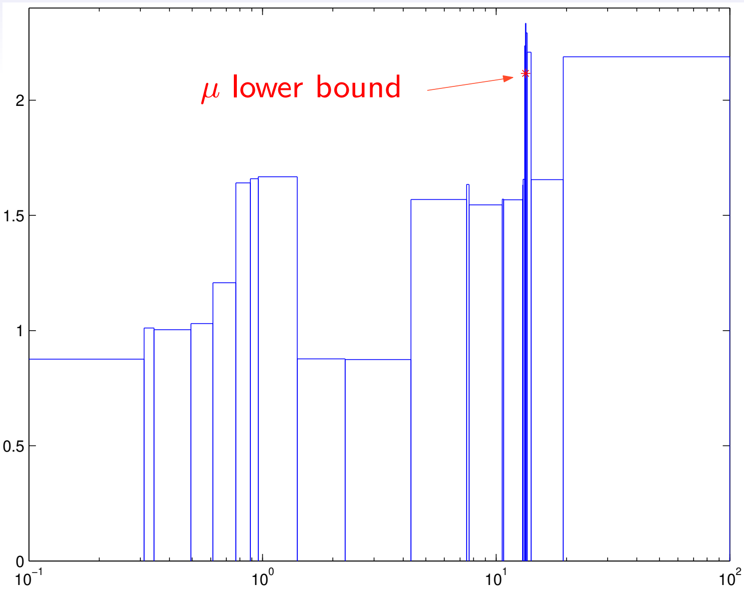
FLEXIBLE AIRCRAFT: standard μ upper bound

500 points are considered within the interval $[0, 100\, rad/s]$.



The highest value of the peak is not detected and the robustness margin is considerably overestimated (CPU time = 28s)!

FLEXIBLE AIRCRAFT: improved μ upper bound



A reliable and accurate value of the robustness margin is computed: no peak missed, low gap between bounds (10%), low computation time (6s).

CONCLUSION

μ-analysis is a very mature robustness analysis technique, which has been successfully applied to challenging industrial applications:

- high-order systems (including flexible modes)
- numerous uncertainties
- low computational time
- low conservatism

Matlab tools:

- modeling $\rightarrow$ SMAC/GSS Toolbox (ONERA), Robust control toolbox
- analysis $\rightarrow$ SMAC/SMART Toolbox (ONERA), Robust control toolbox

Limitations:

- highly-repeated real uncertainties (use of branch-and-bound algorithms)
- time-varying parameters and nonlinearities

Matlab workshops

ONERA
THE FRENCH AEROSPACE LAB

ENAC

isae
Institut Supérieur de l'Aéronautique et de l'Espace

Optimal control: Lab-work

Optimal rendez-vous trajectory

Let us consider a space station S orbiting around the earth (orbit center: O) and a servicing vehicle M in the orbital plane. The objective is to dock the (active) chaser M on the (passive) target S.

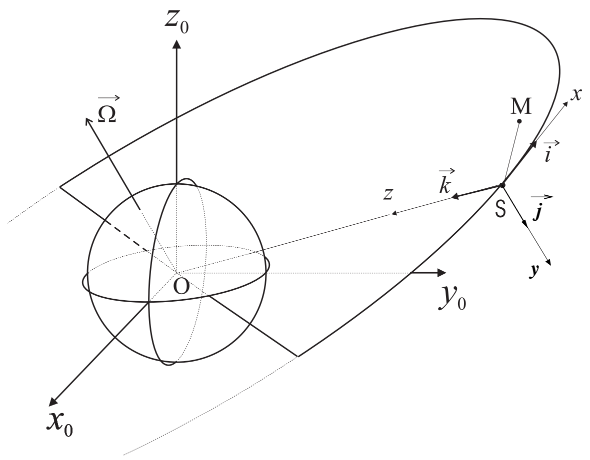


Figure 1: Frame definition

The differential equations governing the relative dynamics of M w.r.t. S in the orbital plane (simplified HILL-CLOHESSY-WHITSHIRE equations) projected in the local orbital frame $\left\{S,\vec{i},\vec{j},\vec{k}\right\}$ reads:

$$\ddot{x} - 2\omega\dot{z} = \varphi_x$$
$$\ddot{z} + 2\omega\dot{x} - 3\omega^2 z = \varphi_z$$

where x and z are the coordinates of M in the local orbital plane.

The 2 control inputs are the specific propulsion forces φ_x and φ_z (resp. tangential and radial) of M.

1) Give a state space representation of this system:

$$\dot{X} = \mathbf{A}X + \mathbf{B}u$$

where $X^T = (z, x, \dot{z}, \dot{x})$ and $u^T = (\varphi_z, \varphi_x)$.

2) Is the system controllable using:
a) the radial thrust φ_z only ?
b) the tangential thrust φ_x only ?
In the following, it is assumed that only one control input can be used. Which one (φ_x or φ_z) ?.

3) At the initial time $t = 0$, the space vehicle M has an initial state $X(0) = X_0$, assumed to be measured. The objective is to dock M on S at a given time $t = T$ while minimizing the fuel consumption of M. The problem is thus to find the state optimal trajectories $\widehat{X}(t)$ and the optimal control $\widehat{u}(t)$[1] such that $X(T) = \mathbf{0}$ while minimizing the performance index:

$$\mathscr{C} = \int_0^T \frac{1}{2} u^2 dt$$

3.1) In a first step, one wishes to develop a generic MATLAB function solving the two point boundary-value problem in the general case of a linear system, a quadratic index and a finite time horizon. Fill the code of the following `twopbvp.m` function:

[1] In this case u is a dimension 1 signal since only one control input is used.

```
function [K_t,P_t,phi_t]=twopbvp(T,t,a,b,q,r)
%TWOPBVP   Two point boundary-value Problem (LQ problem with finite horizon T
%          and null final state).
%
%        * K_t=twopbvp(T,t,A,B,Q,R) compute, at current time t (in [0, T[),
%          the optimal gain K_t for the LQ problem:
%
%          - System:
%              .
%              x = Ax + Bu  with negative state feedback: u(t)=-K_t(t)*x(t),
%
%          - Performance index
%                      /T
%              J = 0.5 | {x'Qx + u'Ru} dt
%                     /0
%          - Hard constraint:
%              x(T)=0
%
%        * [K_t,P_t]=twopbvp(T,t,A,B,Q,R) computes also P_t: the (semi-
%          definite positive) solution at current time t of the associated
%          Riccati equation:
%               .                              -1
%              P_t = -P_t A - A' P_t + P_t B R  B'P_t - Q
%
%        * [K_t,P_t,phi_t]=twopbvp(T,t,....) computes also the transistion
%          matrix phi_t at current time t on the optimal trajectory,
%          such that:
%                        x(t)=phi_t x(0)


error(nargchk(6,6,nargin));
error(abcdchk(a,b));

if t>T, disp('Input argument problem: t>T !!!');K_t=[];P_t=[];phi_t=[];return,end

[m,n] = size(a);[mb,nb] = size(b);[mq,nq] = size(q);[mr,nr] = size(r);
if (m ~= mq) | (n ~= nq)
       error('A and Q must be the same size');
end
if (mr ~= nr) | (nb ~= mr)
       error('B and R must be consistent');
end

H=[a -b*inv(r)*b';-q -a'];
eH_t=expm(H*(T-t));
...
```

3.2) Numerical application: $\omega = \frac{2\pi}{T_{orb}}\,(rd/s)$, $T = \frac{1}{4}T_{orb}$ and $T_{orb} = 5400\,(s)$.
For the 4 following initial states:

1. $z(0) = \dot{z}(0) = \dot{x}(0) = 0$ et $x(0) = 1000\,(m)$,
2. $z(0) = \dot{z}(0) = \dot{x}(0) = 0$ et $x(0) = -1000\,(m)$,
3. $x(0) = \dot{x}(0) = \dot{z}(0) = 0$ et $z(0) = 1000\,(m)$,
4. $x(0) = \dot{x}(0) = \dot{z}(0) = 0$ et $z(0) = -1000\,(m)$,

compute $\widehat{\mathscr{C}}$, $\widehat{u}(t)$, $\widehat{X}(t)$ and plot, using the given user-function `plotresults.m`:

- the 4 optimal state trajectories $\widehat{z}(t), \widehat{x}(t), 100\widehat{\dot{z}}(t), 100\widehat{\dot{x}}(t)$ in the same graph,
- the optimal control response $\widehat{u}(t)$,
- the optimal trajectory in the local orbital plane $\widehat{x} = F(\widehat{z})$.

Comment these responses.
3.3) Knowing that:

- the chaser M has a total energy of 1 (with the same unit than the performance index $\mathscr{C}$),
- at $t = 0$, M is exactly on the same orbit than S ($z(0) = 0$) with a null relative velocity ($\dot{x}(0) = \dot{z}(0) = 0$),

what is the maximal distance $|x(0)|_{max}$ for the rendez-vous to be possible?
3.4) go back to questions 3.2) et 3.3) considering now that the 2 control inputs (φ_z and φ_x) can be used and the new performance index:

$$\mathscr{C} = \int_0^T \frac{1}{2}(\varphi_z^2 + \varphi_x^2)dt \ .$$

Exercises on H_∞ control with Matlab

1 H_∞ design

Let us consider the second order model of an unstable aerospace vehicle:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{s^2 - 1}$$

The objective is to design an H_∞ feedback controller $K(s)$ to meet the template depicted in Figure 1 on the sensitivity function S. The model G is also subject to an additive uncertainty Δ whose an upper bound in the frequency-domain is plotted in Figure 2.

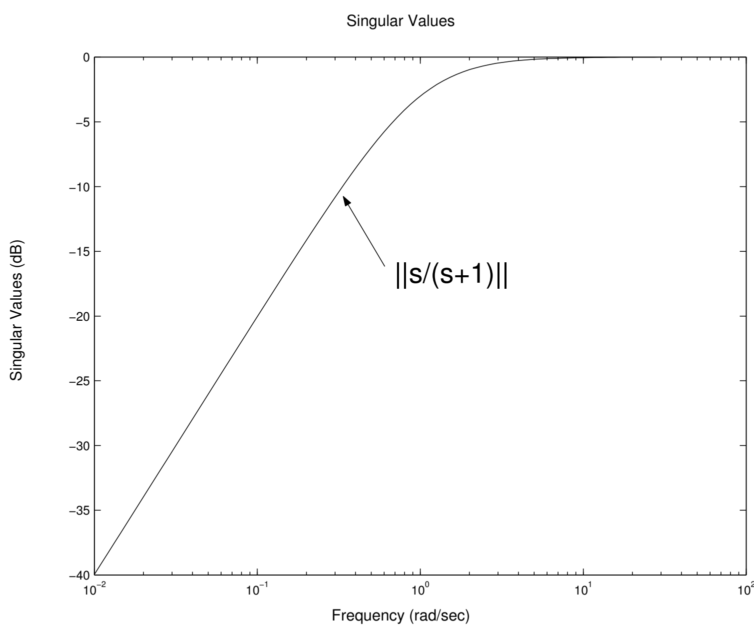


Figure 1: Template on the sensitivity function S.

- Translate this control problem using H_∞ specifications and describe it using a block-diagram representation,
- Propose a set of weighting filters from Figures 2 et 1,
- Compute the state-space representation of the corresponding standard problem $P(s)$,
- Design the controller $K(s)$ using the Matlab Robust Control Toolbox:
  - firstly, using function `hinfsyn` (full order controller design),
  - secondly, using function `systune` (fixed-structure controller design), a multi-objective formulation of the control problem and a judicious controller structure,

- compare and comment the 2 solutions[1].

- Redesign, analyze and comment the controller obtained assuming that the measurement of $\dot{y}$ is also available.

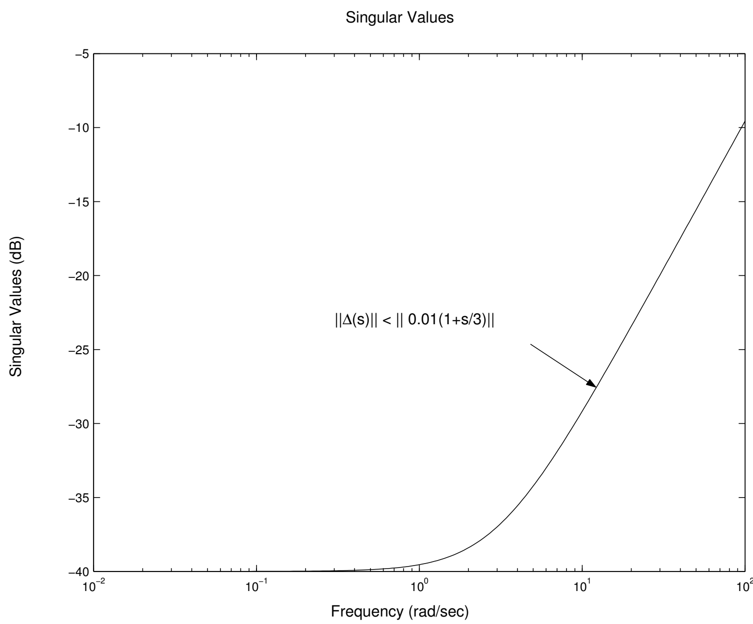


Figure 2: Uncertainty Δ upper bound.

[1]That is, plot the frequency-domain responses (function `sigma`) of the constrained closed-loop transfers and their templates.

Robustness analysis of a spark ignition engine

Clément Roos
ONERA

The term spark-ignition engine refers to internal combustion engines, usually petrol engines, where the combustion process of the air-fuel mixture is ignited by a spark from a spark plug. This is in contrast to compression-ignition engines, typically diesel engines, where the heat generated from compression is enough to initiate the combustion process, without needing any external spark.

A block diagram of the linearized open-loop engine model is shown in Figure 1. It is composed of three main blocks: the filling dynamics of the manifold chamber, the combustion process and the rotational dynamics of the engine. The inputs are the duty cycle of the throttle valve $D(t)$ and the spark advance position $A(t)$. The outputs are the relative air pressure of the manifold $P(t)$ and the engine speed $N(t)$. Finally, $K_1, \ldots, K_7$ are static gains and J corresponds to the polar moment of inertia. Note that $1/J$ will be replaced with K_8 in the sequel for the ease of notation.

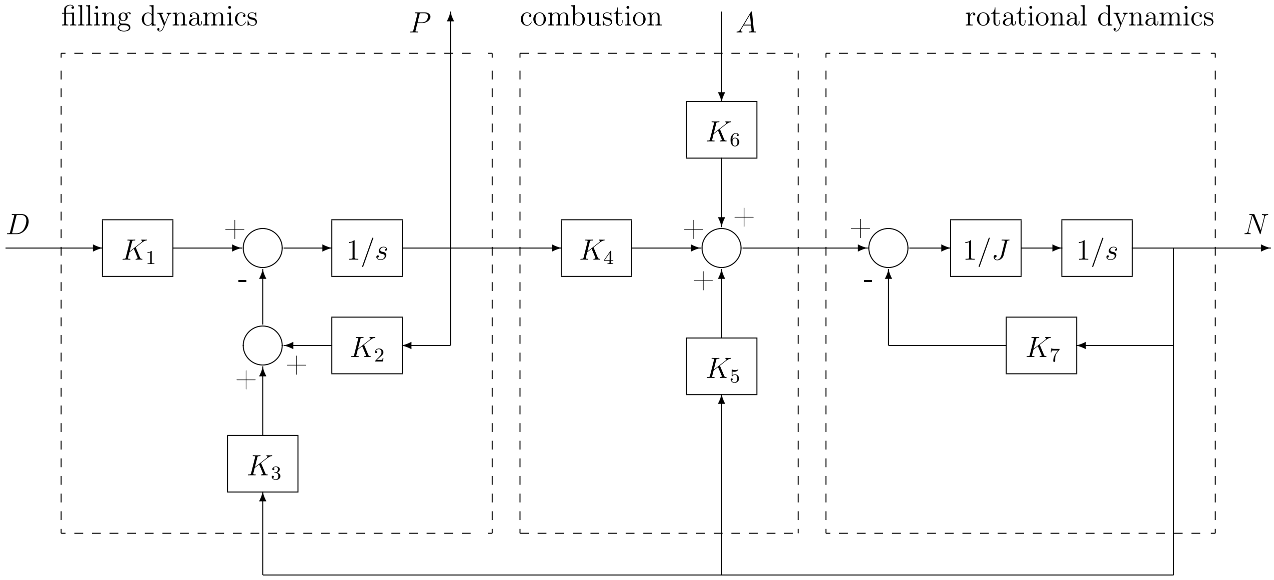


Figure 1: Block diagram of the linearized open-loop engine model

The engine can work at many different operating points. Three of them are considered here, which correspond to various working conditions. Point A represents the completely unloaded engine at idle speed. Point B represents a slightly loaded engine at idle speed. Finally, point C represents the case in which the first gear is engaged. Standard identification techniques are used to obtain three sets of parameters, which are given in Table 1.

Parameters	K_1	K_2	K_3	K_4	K_5	K_6	K_7	$K_8 = 1/J$
Point A	2.1608	0.1027	0.0357	0.5607	2.1999	3.8429	2.0283	1.0000
Point B	3.4329	0.1627	0.1139	0.2539	1.7993	1.8078	1.8201	1.0000
Point C	2.1608	0.1027	0.0357	0.5607	1.7993	3.8429	1.8201	0.1000

Table 1: Parameters values for the three operating points of interest

A controller is designed for operating point B, which is the most commonly occurring. The whole closed-loop interconnection is represented in Figure 2 and the controller gains are given below:

$$G = \begin{bmatrix} 0.0081 & 0.0872 & 0.1586 & -0.1202 \\ 0.0187 & 0.1826 & 0.0848 & -0.0224 \end{bmatrix}$$

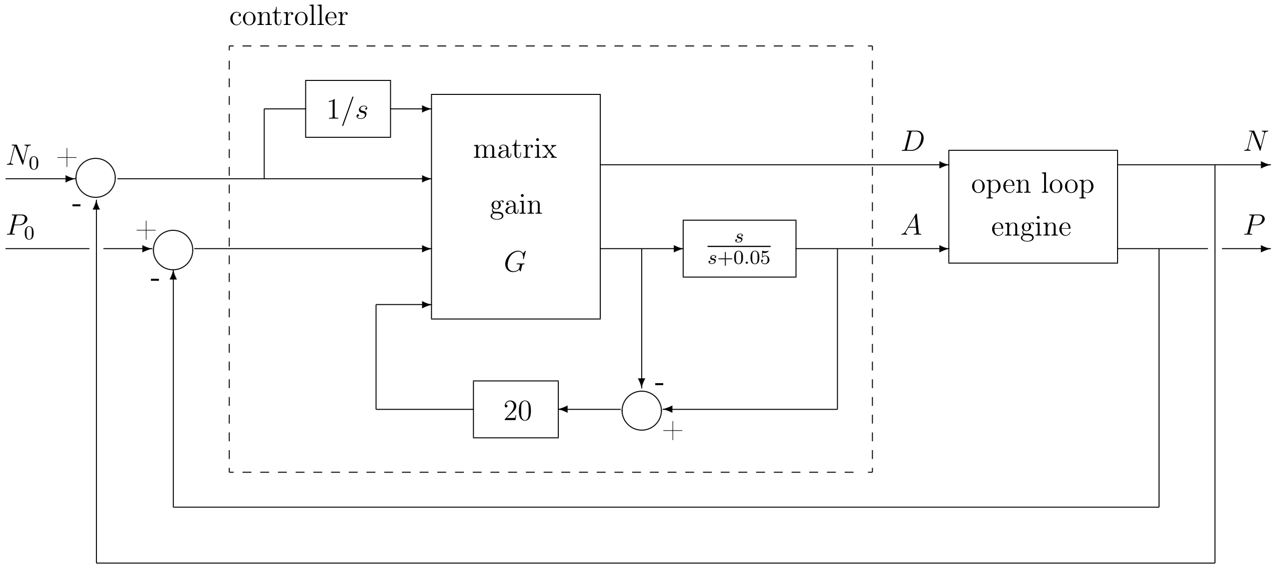


Figure 2: Block diagram of the closed-loop engine model

For each $i = 1, \ldots, 8$, the values of K_i at operating points A, B and C are denoted K_i^A, K_i^B and K_i^C respectively. The assumption is made that during the normal operation of the engine, K_i may take all values between $K_i^- = \min(K_i^A, K_i^B, K_i^C)$ and $K_i^+ = \max(K_i^A, K_i^B, K_i^C)$. Hence, the range of variation of K_i is simply the interval $\left[K_i^-, K_i^+\right]$. As a result, the considered parametric domain $\mathbf{K}$ is a box defined as:

$$\mathbf{K} = \left\{(K_1, \ldots, K_8) \in \mathbf{R}^8 : K_i^- \leq K_i \leq K_i^+\right\}$$

The objective is to determine whether the closed-loop interconnection of Figure 2 is stable for all $K \in \mathbf{K}$.

Instructions

You have to work **in pairs**. Evaluation will be performed on the basis of a report in **pdf format**, which should include all necessary comments, figures and pieces of Matlab code. All answers should be **justified**, but try to be **concise (maximum 10 pages)**! The report should be uploaded on the LMS **the day of the workshop before midnight**.

In addition to standard Matlab tools, two libraries of the Systems Modeling Analysis and Control (SMAC) Toolbox are used in this work:

- the GSS library allows to build Linear Fractional Representations and can be downloaded at *http://w3.onera.fr/smac/gss*,
- the SMART library allows to compute bounds on the structured singular value μ and can be downloaded at *http://w3.onera.fr/smac/smart*.

There is no tricky question, even if you have not understood all the underlying theory. Nevertheless, you must be **rigorous** and make no mistake when handling the numerical values of the parameters, or you might quickly get into trouble. It is also important to **read and understand the Matlab documentation** before you use a function. For example, if you do not know how to use the function `fun`, first type `help fun` (or `help gss/fun` in case of an overloaded function).

1 Modeling

Q1.1 - By hand, compute the state-space representation of the open-loop engine shown in Figure 1 with state vector $x = [N \; P]^T$, input vector $u = [D \; A]^T$ and output vector $y = x$. Find the expressions of $\delta_1, \ldots, \delta_7$ such that:

$$\dot{x} = \begin{bmatrix} \delta_5\delta_7 & \delta_4\delta_7 \\ \delta_3 & \delta_2 \end{bmatrix} x + \begin{bmatrix} 0 & \delta_6\delta_7 \\ \delta_1 & 0 \end{bmatrix} u \tag{1}$$

Using Matlab, define the parameters $\delta_1, \ldots, \delta_7$ with their ranges of variation $[\delta_1^-, \delta_1^+], \ldots, [\delta_7^-, \delta_7^+]$ and their nominal values $\dfrac{\delta_1^- + \delta_1^+}{2}, \ldots, \dfrac{\delta_7^- + \delta_7^+}{2}$ (function `gss`). Remember that the gains $K_1, \ldots, K_8$ can vary independently of each other. Create the open-loop parametric model (overloaded function `ss`).

Q1.2 - Compute a state-space representation of the controller shown in Figure 2. The state matrices can be obtained either by hand or by using Simulink and the function `linmod`.

Q1.3 - Create the closed-loop parametric model $\Sigma_1(\delta)$ represented in Figure 2 (overloaded function `feedback`). A Linear Fractional Representation `sys` is obtained, where the LTI model $M(s)$ and the structure of the block-diagonal matrix $\Delta = \text{diag}(\delta_1 I_{q_1}, \ldots, \delta_7 I_{q_7})$ are given by the fields `sys.M` and `sys.D` respectively. Give the order of $M(s)$ and the values of $q_1, \ldots, q_7$ (overloaded function `size`).

2 Stability analysis

Q2.1 - Check closed-loop stability at the operating points A, B and C (overloaded function `eval`).

Q2.2 - Evaluate the stability of the closed-loop parametric model $\Sigma_1(\delta)$ with 1000 simulations (function `dbsample`). Plot the eigenvalues of all samples on the same figure. In case an unstable sample is found, give the corresponding values of $\delta_1, \ldots, \delta_7$.

Q2.3 - Evaluate the stability of the closed-loop parametric model $\Sigma_1(\delta)$ with μ-analysis:

- Remove all exogenous inputs and outputs of $\Sigma_1(\delta)$ to obtain an LFR for stability analysis.
- Compute μ upper and lower bounds on the whole frequency range (functions `muub` and `mulb`). You can use the following Matlab code:

```
optub.lmi=1;optub.tol=0;
ubnd=muub(sys,[],optub);
[lbnd,wc,pert,iodesc]=mulb(sys);
```

- Conclude about the stability of the closed-loop model on the parametric domain $\mathbf{K}$. Note that the Δ matrix is normalized in `muub` and `mulb`, so stability is ensured on $\mathbf{K}$ if and only if the largest value of μ over the whole frequency range is less than 1. In case an unstable configuration is found, give the corresponding values of $\delta_1, \ldots, \delta_7$ (output argument `iodesc` of function `mulb`) and check your result using the overloaded function `eval`.

Q2.4 - Compare the results of Q2.2 and Q2.3.

3 Adjustment of the combustion chamber

Following the conclusions of Q2.2 and Q2.3, some adjustments are performed inside the combustion chamber. New values are identified for K_5 and K_6, as shown in Table 2.

Parameters	K_5	K_6
Point A	2.0183	4.4962
Point B	1.7993	2.0247
Point C	1.7993	4.4962

Table 2: New parameters values for the three operating points of interest

Q3.1 - Define the parameters $\delta_1, \ldots, \delta_7$ with their new ranges of variation. Create the new closed-loop parametric model $\Sigma_2(\delta)$.

Q3.2 - Evaluate the stability of the closed-loop parametric model $\Sigma_2(\delta)$ with μ-analysis. Conclude about the stability of the closed-loop model on the parametric domain $\mathbf{K}$.

4 Improvement of the analysis results

4.1 By applying branch-and-bound

The gap between the μ upper and lower bounds can be reduced by applying a branch-and-bound algorithm. The idea is to partition the uncertainty domain in more and more subsets until the gap between the highest lower bound and the highest upper bound computed on all subsets becomes less than a user-defined thresholdϵ. Such an algorithm is known to converge, *i.e.* ϵ can be reduced to an arbitrarily small value. However, it suffers from an exponential growth of computational complexity as a function of the number of real uncertainties. The choice of ϵ allows to handle the tradeoff between accuracy and computational time.

Q4.1 - Apply the branch-and-bound algorithm to the closed-loop parametric model $\Sigma_2(\delta)$ created in Section 3. You can use the following Matlab code:

```
optbb.maxgap=0.05;
[lbnd,wc,pert,iodesc]=mubb(sys,[],optbb);
```

Conclude about the stability of the closed-loop model on the parametric domain $\mathbf{K}$.

4.2 By reducing the size of $\boldsymbol{\Delta}$

The gap between the μ upper and lower bounds strongly depends on the considered LFR. Although this is not an exact rule, the trend is as follows: the larger the size of Δ, the larger the gap.

Q4.2 - By carefully reading the description of the engine, determine whether the LFRs used until now are minimal. If no, determine the smallest possible values of $q_1, \ldots, q_7$ in the matrix Δ. Find a suitable factorization of equation (1), which leads to this minimal representation. Create the new open-loop parametric model using the numerical values of Section 3 and the corresponding closed-loop parametric model $\Sigma_3(\delta)$.

Q4.3 - Evaluate the stability of the closed-loop parametric model $\Sigma_3(\delta)$ with μ-analysis. Conclude about the stability of the closed-loop model on the parametric domain $\mathbf{K}$. Compare the results with those obtained in Q3.2 and Q4.1.