Pole placement control: state space and polynomial approaches
Lecture 1

O. Sename

1Gipsa-lab, CNRS-INPG, FRANCE
Olivier.Sename@gipsa-lab.fr
www.gipsa-lab.fr/~o.sename

November 7, 2017
Outline

Introduction

Modelling of dynamical systems

Stability

Controllability

Observability

Concerning discrete-time systems

Some properties
References

Some interesting books:

Course Objectives

Main objective: Pole placement control design method: state space and polynomial approaches.

- Study of system properties in view of control/observation
- Synthesis of state feedback controller for state space linear systems
- Design of state observers (estimators)
- Calculation of a 2 degrees-of-freedom controller using a polynomial approach

for continuous-time and discrete-time linear systems as well.
Why state space equations?

- dynamical systems where physical equations can be derived: electrical engineering, mechanical engineering, aerospace engineering, microsystems, process plants ....
- include physical parameters: easy to use when parameters must be changed for the design
- State variables have physical meaning.
- Allow for including non linearities (state constraints)
- Easy to extend to Multi-Input Multi-Output (MIMO) systems
- Advanced control design methods are based on state space equations (reliable numerical optimisation tools)
Some physical examples
Modelling of dynamical systems
Many dynamical systems can be represented by Ordinary Differential Equations (ODE) as

\[
\begin{align*}
\dot{x}(t) &= f((x(t), u(t), t), \quad x(0) = x_0 \\
y(t) &= g((x(t), u(t), t) \quad (1)
\end{align*}
\]

where \( f \) and \( g \) are non-linear functions.
Example: Lateral vehicle model

The dynamical equations are as follows:

\[
\begin{align*}
\text{NL Horizontal chassis dynamics} \\
\begin{aligned}
a_x &= \dot{v}_x - \psi \dot{v}_y \\
a_y &= \dot{v}_y + \psi \dot{v}_x \\
m a_y &= 2 F_{tyf} \cos(\delta_f) + 2 F_{tyr} + F_{yd} \\
l_y \ddot{\psi} &= 2 F_{tyf} \cos(\delta_f) l_f - 2 F_{tyr} l_r + M_{yd}
\end{aligned}
\end{align*}
\]

Assumption

\[F_{tyf} \cos(\delta_f) \gg F_{txf} \sin(\delta_f)\]

\[
\text{NL Lateral Tire Forces} \\
\begin{aligned}
\alpha_f &= \arctan\left(\frac{v_y + l_f \psi}{v_x}\right) - \delta_f \\
\alpha_r &= \arctan\left(\frac{v_y - l_r \psi}{v_x}\right)
\end{aligned}
\]
Vehicle model - synopsis

\[
\begin{bmatrix}
\dot{x}_s, \dot{y}_s \\
\ddot{x}_s, \ddot{y}_s \\
F_{sz}, z_{us}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\dot{z}_s, z_s \\
\dot{z}_{us}, z_{us}
\end{bmatrix}
\rightarrow
\text{Suspensions}
\rightarrow
\begin{bmatrix}
F_{sx, y, z}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
x_s \\
y_s \\
z_s
\end{bmatrix}
(external disturbances)

\text{(suspensions control)}

\begin{bmatrix}
\dot{x}_s, \dot{y}_s, \dot{z}_s \\
\ddot{x}_s, \ddot{y}_s, \ddot{z}_s \\
\psi, \nu, v, F_{sz}, z_{us}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\dot{\theta} \\
\dot{\phi} \\
\dot{\psi}
\end{bmatrix}
(vehicle dynamics)

\text{(braking & steering control)}

\begin{bmatrix}
[T_{bij}, \delta]
\end{bmatrix}

\text{(road characteristics)}

\begin{bmatrix}
\mu_{ij}, z_{r_{ij}}
\end{bmatrix}
(tire, wheel dynamics)

\begin{bmatrix}
\dot{x}_s, \dot{y}_s \\
\ddot{x}_s, \ddot{y}_s \\
\lambda_{ij}, \beta_{ij}, \omega_{ij}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
F_{sx, y, z}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\dot{\theta} \\
\dot{\phi} \\
\dot{\psi}
\end{bmatrix}
(external disturbances)

\text{(tire, wheel dynamics)}

\text{(vehicle dynamics)}
Vehicle model - dynamical equations

\[
\begin{align*}
\ddot{x}_s &= \frac{\left(F_{txfr} + F_{txfl}\right) \cos(\delta) + \left(F_{txrr} + F_{txrl}\right) - \left(F_{tyfr} + F_{tyfl}\right) \sin(\delta) + m\dot{\psi} \dot{y}_s - F_{dx}}{m} \\
\dot{y}_s &= \frac{\left(F_{tyfr} + F_{tyfl}\right) \cos(\delta) + \left(F_{tyrr} + F_{tyrl}\right) + \left(F_{txfr} + F_{txfl}\right) \sin(\delta) - m\dot{\psi} \dot{x}_s - F_{dy}}{m} \\
\ddot{z}_s &= \frac{- \left(F_{szfl} + F_{szfr} + F_{szrl} + F_{szrr} + F_{dz}\right)}{m_s} \\
\ddot{\lambda}_{ij} &= \frac{v_{ij} - R_{ij} \omega_{ij} \cos(\beta_{ij})}{\max(v_{ij}, R_{ij} \omega_{ij} \cos(\beta_{ij}))} \\
\dot{\omega}_{ij} &= \frac{- RF_{txij} \left(\mu, \lambda, F_n\right) + T_{b_{ij}}}{l_w} \\
\beta_{ij} &= \arctan\left(\frac{\dot{x}_{ij}}{\dot{y}_{ij}}\right)
\end{align*}
\]
A continuous-time LINEAR state space system is given as:

\[
\begin{cases}
\dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0 \\
y(t) = Cx(t) + Du(t)
\end{cases}
\]  

- \( x(t) \in \mathbb{R}^n \) is the system state (vector of state variables),
- \( u(t) \in \mathbb{R}^m \) the control input
- \( y(t) \in \mathbb{R}^p \) the measured output
- \( A, B, C \) and \( D \) are real matrices of appropriate dimensions
- \( x_0 \) is the initial condition.

\( n \) is the order of the state space representation.

Matlab: \( \text{ss}(A,B,C,D) \) creates a SS object SYS representing a continuous-time state-space model.
Systems definition

A discrete-time state space system is as follows:

\[
\begin{align*}
   x((k+1)h) &= A_d x(kh) + B_d u(kh), \quad x(0) = x_0 \\
   y(kh) &= C_d x(kh) + D_d u(kh)
\end{align*}
\]

(3)

where \( h \) is the sampling period.

Matlab : \texttt{ss}(A_d,B_d,C_d,D_d,h) creates a SS object SYS representing a discrete-time state-space model.
Recall Laplace & Z-transform

From Transfer Function to State Space

<table>
<thead>
<tr>
<th>$H(s)$ to state space</th>
<th>$H(z)$ to state space</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{\dot{X}}{U} = \text{den}(s)$</td>
<td>$\frac{\dot{X}}{U} = \text{den}(z)$</td>
</tr>
<tr>
<td>$\frac{Y}{X} = \text{num}(s)$</td>
<td>$\frac{Y}{X} = \text{num}(z)$</td>
</tr>
<tr>
<td>$\dot{X} = AX + BU$</td>
<td>$X_{k+1} = A_d X_k + B_d U_k$</td>
</tr>
<tr>
<td>$Y = CX + DU$</td>
<td>$Y_k = C_d X_k + D_d U_k$</td>
</tr>
<tr>
<td>$Y(s) = \left[ C [s I - A]^{-1} B + D \right] U(s)$</td>
<td>$Y(z) = \left[ C_d [z I - A_d]^{-1} B_d + D_d \right] U(z)$</td>
</tr>
</tbody>
</table>
Stability
Stability

Definition

An equilibrium point $x_{eq}$ is stable if, for all $\rho > 0$, there exists a $\eta > 0$ such that:

$$\|x(0) - x_{eq}\| < \eta \implies \|x(t) - x_{eq}\| < \rho, \forall t \geq 0$$

Definition

An equilibrium point $x_{eq}$ is **asymptotically stable** if it is stable and, there exists $\eta > 0$ such that:

$$\|x(0) - x_{eq}\| < \eta \implies x(t) \rightarrow x_{eq}, \text{ when } t \rightarrow \infty$$

These notions are equivalent for linear systems (not for non linear ones).
Stability Analysis

The stability of a linear state space system is analyzed through the characteristic equation $\det(sI_n - A) = 0$. The system poles are then the eigenvalues of the matrix $A$. It then follows:

**Proposition**

A system $\dot{x}(t) = Ax(t)$, with initial condition $x(0) = x_0$, is stable if $\text{Re}(\lambda_i) < 0$, $\forall i$, where $\lambda_i$, $\forall i$, are the eigenvalues of $A$.

Using Matlab, if SYS is an SS object then `pole(SYS)` computes the poles $P$ of the LTI model SYS. It is equivalent to compute `eig(A)`. 
Stability Analysis - Lyapunov

The stability of a linear state space system can be analysed through the Lyapunov theory. It is the basis of all extension of stability for non linear systems, time-delay systems, time-varying systems ...

**Theorem**

A system $\dot{x}(t) = Ax(t)$, with initial condition $x(0) = x_0$, is asymptotically stable at $x = 0$ if and only if there exist some matrices $P = P^T > 0$ and $Q > 0$ such that:

$$A^T P + PA = -Q \quad (4)$$

see *lyap* in MATLAB.

**Proof:** The Lyapunov theory says that a linear system is stable if there exists a continuous function $V(x)$ s.t.:

$$V(x) > 0 \text{ with } V(0) = 0 \text{ and } \dot{V}(x) = \frac{dV}{dx} \leq 0$$

A possible Lyapunov function for the above system is: $V(x) = x^T Px$
Controllability
Controllability

Controllability refers to the ability of controlling a state-space model using state feedback.

Definition

*Given two states* \( x_0 \) *and* \( x_1 \), the system \( (2) \) *is controllable if there exist* \( t_1 > 0 \) *and a piecewise-continuous control input* \( u(t), \ t \in [0, t_1] \), *such that* \( x(t) \) *takes the values* \( x_0 \) *for* \( t = 0 \) *and* \( x_1 \) *for* \( t = t_1 \).

Proposition

*The controllability matrix is defined by* \( C = [B, A.B, \ldots, A^{n-1}.B] \). *Then system* \( (2) \) *is controllable if and only if* \( \text{rank}(C) = n \).

*If the system is single-input single output (SISO), it is equivalent to* \( \text{det}(C) \neq 0 \).

Using **Matlab**, if SYS is an SS object then **ctrb(SYS)** returns the controllability matrix of the state-space model SYS with realization \((A,B,C,D)\). This is equivalent to **ctrb(sys.a,sys.b)**

Exercises

Test the controllability of the previous examples: DC motor, suspension, inverted pendulum.
More on controllability

- Other criteria
- Definition of stabilizability
- About the decomposition into the controllable/uncontrollable subspaces: use of ctrbf
Observability
Observability refers to the ability to estimate a state variable, from the knowledge of the system input and output variables.

**Definition**

A linear system (2) is completely observable if, given the control and the output over the interval \( t_0 \leq t \leq T \), one can determine any initial state \( x(t_0) \).

It is equivalent to characterize the non-observability as:

A state \( x(t) \) is not observable if the corresponding output vanishes, i.e. if the following holds: \( y(t) = \dot{y}(t) = \ddot{y}(t) = \ldots = 0 \).
Where does observability come from?

Compare the transfer function of the two different systems

\[
\dot{x} = -x + u \\
y = 2x
\]

and

\[
\dot{x} = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix} x + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u \\
y = \begin{bmatrix} 2 & 0 \end{bmatrix} x
\]
Proposition

The observability matrix is defined by $\mathcal{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$. Then system (2) is observable if and only if $\text{rank}(\mathcal{O}) = n$.

If the system is single-input single output (SISO), it is equivalent to $\det(\mathcal{O}) \neq 0$.

Using Matlab, if SYS is an SS object then $\text{obsv}(\text{SYS})$ returns the observability matrix of the state-space model SYS with realization (A,B,C,D). This is equivalent to $\text{OBSV(sys.a,sys.c)}$.

Exercises

Test the observability of the previous examples: DC motor, suspension, inverted pendulum.
Analysis of different cases, according to the considered number of sensors.
Concerning discrete-time systems
State space analysis (discrete-time systems)

Stability
A system (state space representation) is stable iff all the eigenvalues of the matrix $A_d$ are inside the unit circle.

Controllability definition

Definition
Given two states $x_0$ and $x_1$, the system (3) is controllable if there exist $K > 0$ and a sequence of control samples $u_0, u_1, \ldots, u_{K-1}$, such that $x_k$ takes the values $x_0$ for $k = 0$ and $x_1$ for $k = K_1$.

Observability definition

Definition
The system (3) is said to be completely observable if every initial state $x(0)$ can be determined from the observation of $y(k)$ over a finite number of sampling periods.
State space analysis (2)

Controllability
The system is controllable iff
\[ C(A_d, B_d) = \text{rg}[B_d A_d B_d \ldots A_d^{n-1} B_d] = n \]

Observability
The system is observable iff
\[ O(A_d, C_d) = \text{rg}[C_d C_d A_d \ldots C_d A_d^{n-1}]^T = n \]

Duality
Observability of \((C_d, A_d) \Leftrightarrow\) Controllability of \((A_d^T, C_d^T)\).
Controllability of \((A_d, B_d) \Leftrightarrow\) Observability of \((B_d^T, A_d^T)\).
Some properties
Minimality

Definition

A state space representation of a linear system (2) of order \( n \) is said to be minimal if it is controllable and observable.

In this case, the corresponding transfer function \( G(s) \) is of minimal order \( n \), i.e. is irreducible (no cancellation of poles and zeros). When the transfer function is not of minimal order, there exists non controllable or non observable modes.
Kalman decomposition

- When the linear system (2) is not completely controllable or observable, it can be decomposed as shown.
- It is of course very important, in that case, to study the stability of the non controllable and non observable modes, which refers to as the **stabilizability** and **detectability**, respectively.
- Use `ctrbf` and `obsvf` in Matlab.