کنترل پیش بین
Model Predictive Control

ارائه کننده: امیرحسین نیکوفرد

مهندسی برق و کامپیوتر دانشگاه خواجه نصیر

دانشگاه صنعتی خواجه نصیرالدین طوسی

Keywords

- In most commercial product acronyms we find several important keywords that define the MPC technologies
- Control
- Model
- Predictive
- Multivariable
- Robustness
- Constraints
- Optimization
- Identification

Models of Dynamic Systems

- **Goal:** Introduce **mathematical models** to be used in Model Predictive Control (MPC) describing the **behavior** of dynamic systems

- **Model classification:** state space/transfer function, linear/nonlinear, time-varying/time-invariant, continuous-time/discrete-time, deterministic/stochastic

- If not stated differently, we use deterministic models

Models of Dynamic Systems

- Models of physical systems derived from first principles are mainly: nonlinear, time-invariant, continuous-time, state space models (1)

- **Target models for standard MPC are mainly**:
- linear, **time-invariant**, discrete-time, state space models (2)
- Focus of this section is on how to 'transform' (1) to (2)

Nonlinear, Time-Invariant, Continuous-Time, State Space Models

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

$x \in \mathbb{R}^n$	state vector	$g(x,u) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$	system dynamics
$u \in \mathbb{R}^m$	input vector	$h(x,u) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^p$	output function
$y \in \mathbb{R}^p$	output vector		

- Very general class of models
- Higher order ODEs can be easily brought to this form (next slide)
- Analysis and control synthesis generally hard ⟶ ***linearization*** to bring it to linear, time-invariant (LTI), continuous-time, state space form

Nonlinear, Time-Invariant, Continuous-Time, State Space Models

- **Equivalence of one** n**-th order ODE and** n **1-st order ODEs**

$$x^{(n)} + g_n(x, \dot{x}, \ddot{x}, \ldots, x^{(n-1)}) = 0$$

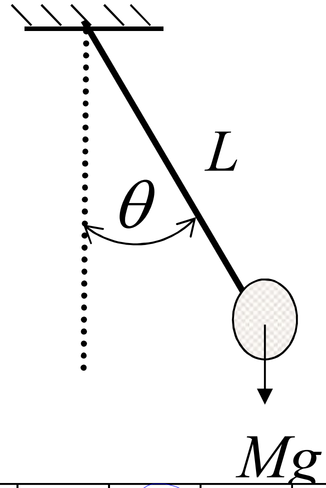
- Define

$$x_{i+1} = x^{(i)}, \quad i = 0, \ldots, n-1$$

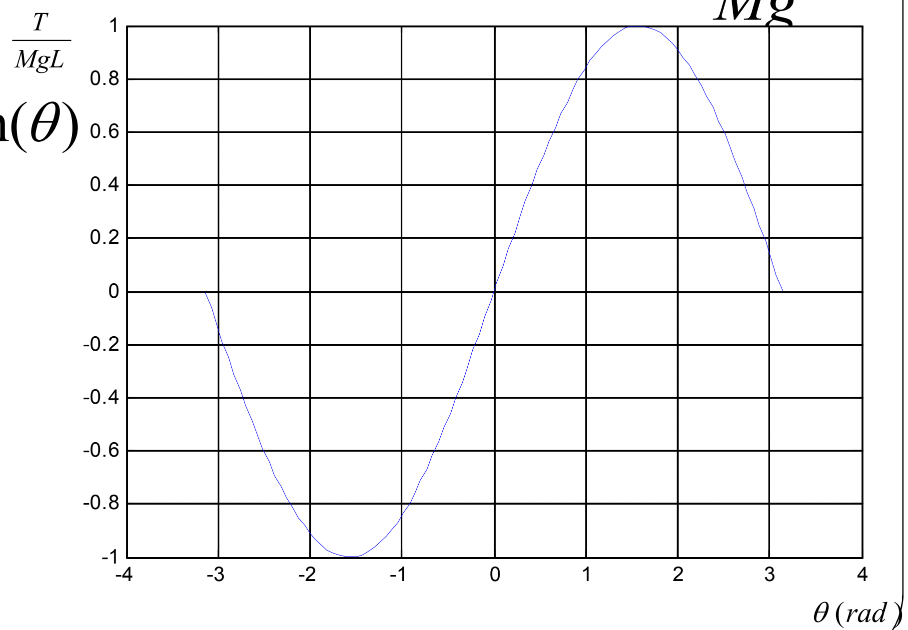
- Transformed system

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= x_3 \\
&\vdots \\
\dot{x}_{n-1} &= x_n \\
\dot{x}_n &= -g_n(x_1, x_2, \ldots, x_n)
\end{aligned}$$

Nonlinear, Time-Invariant, Continuous-Time, State Space Models



- **Example**: **Pendulum**
- Moment of inertia wrt. Rotational axis ML^2
- Torque caused by external force T
- Torque caused by gravity $MgL\sin(\theta)$



- System equation $ML^2\ddot{\theta} = T - MgL\sin(\theta)$
- Using $x_1 \triangleq \theta, x_2 \triangleq \dot{\theta} = \dot{x}_1$

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ \dfrac{-g\sin(x_1)}{L} + \dfrac{T}{ML^2} \end{bmatrix}$$

LTI Continuous-Time State Space Models

$$\dot{x} = A^c x + B^c u$$

$$y = Cx + Du$$

$x \in \mathbb{R}^n$ state vector

$u \in \mathbb{R}^m$ input vector

$y \in \mathbb{R}^p$ output vector

- Vast theory exists for the analysis and control synthesis of linear systems
- Exact solution:

$$x(t) = e^{A^c(t-t_0)}x_0 + \int_{t_0}^{t} e^{A^c(t-\tau)}Bu(\tau)\mathrm{d}\tau$$

LTI Continuous-Time State Space Models

- **Problem**: Most physical systems are nonlinear but linear systems are much better understood
- Nonlinear systems can be well approximated by a linear system in a 'small' neighborhood around a point in state space
- **Idea**: Control keeps the system around some operating point → replace nonlinear by a linearized system around operating point
- First order Taylor expansion of f( ) around $\mathbf{x}_0(t)$ and $\mathbf{r}_0(t)$

$$x_i(t) = f_i(\mathbf{x}_0, \mathbf{r}_0) + \sum_{j=1}^{n} \left(\left. \frac{\partial f_i}{\partial x_j} \right|_{\substack{\mathbf{x}_0 \\ \mathbf{r}_0}} (x_j - x_{0j}) \right) + \sum_{j=1}^{n} \left(\left. \frac{\partial f_i}{\partial r_j} \right|_{\substack{\mathbf{x}_0 \\ \mathbf{r}_0}} (r_j - r_{0j}) \right) + \ldots$$

LTI Continuous-Time State Space Models

- **Definition:** $\Delta x_i = x_i - x_{0i} \qquad \Delta r_i = r_i - r_{0i}$

$$\Delta x_i(t) = \sum_{j=1}^{n}\left(\left.\frac{\partial f_i}{\partial x_j}\right|_{\substack{\mathbf{x}_0\\ \mathbf{r}_0}} \Delta x_j\right) + \sum_{j=1}^{n}\left(\left.\frac{\partial f_i}{\partial r_j}\right|_{\substack{\mathbf{x}_0\\ \mathbf{r}_0}} \Delta r_j\right) + \dots$$

$$\Delta \dot{\mathbf{x}} = A^* \,\Delta \mathbf{x} + B^* \,\Delta \mathbf{r}$$

$$A^* = \left.\begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}\right|_{\substack{\mathbf{x}_0\\ \mathbf{r}_0}} \qquad B^* = \left.\begin{pmatrix} \frac{\partial f_1}{\partial r_1} & \frac{\partial f_1}{\partial r_2} & \cdots & \frac{\partial f_1}{\partial r_p} \\ \frac{\partial f_2}{\partial r_1} & \frac{\partial f_2}{\partial r_2} & \cdots & \frac{\partial f_2}{\partial r_p} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_n}{\partial r_1} & \frac{\partial f_n}{\partial r_2} & \cdots & \frac{\partial f_n}{\partial r_p} \end{pmatrix}\right|_{\substack{\mathbf{x}_0\\ \mathbf{r}_0}}$$

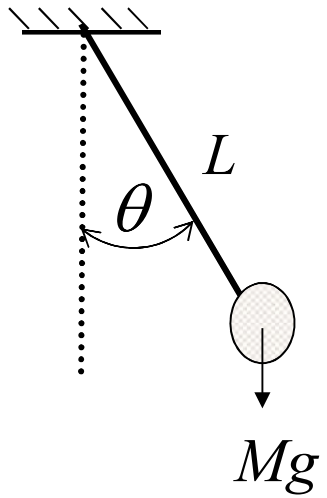
LTI Continuous-Time State Space Models

Linearization

- The linearized system is written in terms of deviation variables $\Delta x, \Delta u, \Delta y$
- Linearized system is only a good approximation for 'small' $\Delta x, \Delta u$
- Subsequently, instead of $\Delta x, \Delta u, \Delta y$,x, u and y are used for brevity

LTI Continuous-Time State Space Models

- **Example:** Linearization of pendulum equations



$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{-g\sin(x_1)}{L} + \frac{T}{ML^2} \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{-g\sin(x_1)}{L} + u \end{bmatrix} = g(x,u)$$

$$y = x_1 = h(x,u)$$

- Want to keep the pendulum around $x_s = (\pi/4, 0)' \rightarrow u_s = \frac{g}{l}\sin(\pi/4)$

$$A^c = \left.\frac{\partial g}{\partial x'}\right|_{\substack{x=x_s \\ u=u_s}} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{l}\cos(\pi/4) & 0 \end{bmatrix}, \quad B^c = \left.\frac{\partial g}{\partial u'}\right|_{\substack{x=x_s \\ u=u_s}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$C = \left.\frac{\partial h}{\partial x'}\right|_{\substack{x=x_s \\ u=u_s}} = [1 \quad 0], \quad D = \left.\frac{\partial h}{\partial u'}\right|_{\substack{x=x_s \\ u=u_s}} = 0$$

Nonlinear, Time-Invariant, Discrete-Time, State Space Models

- Nonlinear discrete-time systems are described by difference equations

$$x(k+1) = g(x(k), u(k))$$
$$y(k) = h(x(k), u(k))$$

$x \in \mathbb{R}^n$	state vector	$g(x,u) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$	system dynamics
$u \in \mathbb{R}^m$	input vector	$h(x,u) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^p$	output function
$y \in \mathbb{R}^p$	output vector		

LTI Discrete-Time, State Space Models

- Linear discrete-time systems are described by linear difference equations

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k) + Du(k)$$

- Inputs and outputs of a discrete-time system are defined onIy at discrete time points, i.e. its inputs and outputs are sequences defined for $k \in \mathbb{Z}^+$
- Discrete time systems describe either
  1. Inherently discrete systems, eg. bank savings account balance at the kth month $x(k+1) = (1+\alpha)x(k) + u(k)$
  2. 'Transformed' continuous-time system

LTI Discrete-Time, State Space Models

- Vast majority of controlled systems **not inherently** discrete-time systems
- Controllers almost always implemented using microprocessors
- **Finite computation** time must be considered in the control system design ⟶ discretize the continuous-time system
- **Discretization** is the procedure of obtaining an 'equivalent' discrete-time system from a continuous-time system
- The discrete-time model describes the state of the continuous-time system only at **particular instances**

Discrete-Time Model

We will use:

- Nonlinear Discrete Time

$$\begin{aligned} x(k+1) &= g(x(k), u(k)) \\ y(k) &= h(x(k), u(k)) \end{aligned}$$

- or LTI Discrete Time

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned}$$

- Discretization Methods
  1. Euler Discretization
  2. ZOH Discretization

Discrete-Time Model Stability

Theorem: Asymptotic Stability of Linear Systems

The LTI system

$$x(k+1) = Ax(k)$$

is globally asymptotically stable

$$\lim_{k \to \infty} x(k) = 0, \forall x(0) \in \mathbb{R}^n$$

if and only if $|\lambda_i| < 1$, $\forall i = 1, \cdots, n$ where λ_i are the eigenvalues of A. [1]

[1] for cont., time LTI systems $\dot{x} = Ax$, the conditions is $Re(\lambda_i) < 0$

Discrete-Time Model Stability

- We consider first the stability of a nonlinear, time-invariant, discrete-time system

$$x_{k+1} = g(x_k) \tag{1}$$

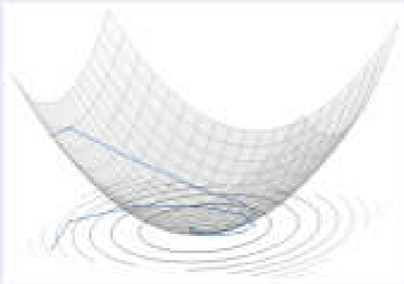
with an equilibrium point at 0

Definition: Lyapunov function

Consider the equilibrium point $x = 0$ of system (1). Let $\Omega \subset \mathbb{R}^n$ be a closed and bounded set containing the origin. A function $V : \mathbb{R}^n \to \mathbb{R}$, continuous at the origin, finite for every $x \in \Omega$, and such that

$$V(0) = 0 \text{ and } V(x) > 0, \ \forall x \in \Omega \setminus \{0\}$$
$$V(g(x_k)) - V(x_k) \leq -\alpha(x_k) \ \forall x_k \in \Omega \setminus \{0\}$$

where $\alpha : \mathbb{R}^n \to \mathbb{R}$ is continuous positive definite,



is called a *Lyapunov function*.

Discrete-Time Model Stability

Lyapunov theorem

Theorem: Lyapunov stability (asymptotic stability)

If a system (1) admits a Lyapunov function $V(x)$, then $x = 0$ is *asymptotically stable* in Ω.

Theorem: Lyapunov stability (global asymptotic stability)

If a system (1) admits a Lyapunov function $V(x)$ that additionally satisfies

$$\|x\| \to \infty \Rightarrow V(x) \to \infty,$$

then $x = 0$ is *globally asymptotically stable*.

Discrete-Time Model Stability

Remarks:

- Note that the Lyapunov theorems only provide **sufficient conditions**
- Lyapunov theory is a powerful concept for proving stability of a control system, but for general nonlinear systems it is usually **difficult** to find a Lyapunov function
- Lyapunov functions can sometimes be derived from **physical considerations**
- One common approach:
  - Decide on form of Lyapunov function (e.g., quadratic)
  - Search for parameter values e.g. via optimization so that the required properties hold
- For linear systems there exist constructive theoretical results on the existence of a **quadratic Lyapunov function**