2E1252
Control Theory and Practice

Lecture 12:
Model predictive control

Mikael Johansson
School of Electrical Engineering
KTH, Stockholm, Sweden

 Mikael Johansson mikaelj@ee.kth.se

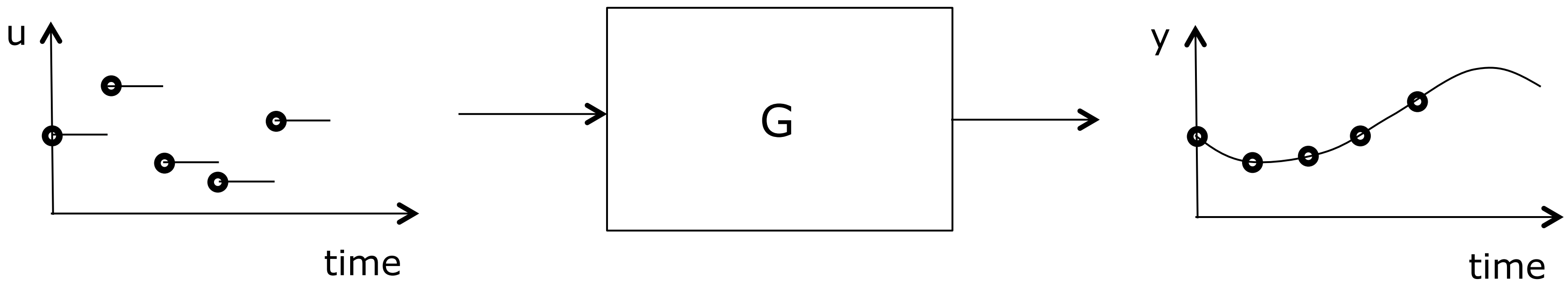
Learning aims

After this lecture, you should

- know how to transform a continuous-time linear system to discrete-time
- be able to formulate and solve a finite-horizon LQR problem
  - by minimizing a quadratic form, or
  - via dynamic programming
- be able to characterize the stationary optimal solution
- understand the principle behind receding-horizon optimal control

Mikael Johansson mikaelj@ee.kth.se

Computer-controlled systems



Output sampled every h seconds, control constant between samples

- how does state evolve between sampling instances?

Mikael Johansson mikaelj@ee.kth.se

Plant dynamics at sampling instants

Recall that

$$\dot{x}(t) = Ax(t) + Bu(t) \Rightarrow x(t+h) = e^{Ah}x(t) + \int_{s=0}^{h} e^{As}Bu(s)\,ds$$

so if u is held constant during sample interval $u(t) = u_t,\ t \in [t, t+h)$

$$x(t+h) = A_D x(t) + B_D u_t \qquad \left(A_D = e^{Ah},\ B_D = \int_{s=0}^{h} e^{As}B\,ds \right)$$

$$y(t) = Cx(t) + Du_t$$

A discrete-time linear system!

Discrete-time linear systems

For notational convenience, we drop reference to physical time and write

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k + Du_k \end{aligned}$$

where

- $\{u_0, u_1, \ldots\}$ is an **input sequence**
- $\{y_0, y_1, \ldots\}$ is the **output sequence**
- $\{x_0, x_1, \ldots\}$ is the **state evolution**

System is stable if all eigenvalues of A are less than one in magnitude

Discrete-time linear systems

Some system theory for discrete-time linear systems (Book Ch. 2.2, 3.7, 4)

System is controllable if $S(A,B) = \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix}$ is full rank.

System is observable if

$$O(A,C) = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

has full rank

Observer-based controllers have the form

$$\begin{aligned} \hat{x}_t &= A\hat{x}_t + Bu_t + K(y_t - \hat{y}_t) \\ u_t &= -L\hat{x}_t \end{aligned}$$

Mikael Johansson mikaelj@ee.kth.se

Finite-horizon LQR problem

Find control sequence

$$U = \{u_0, \ldots, u_{N-1}\}$$

that minimizes the quadratic cost function

$$J(U) = \sum_{k=0}^{N-1} (x_k^T Q_1 x_k + u_k^T Q_2 u_k) + x_N^T Q_f x_N$$

for given state cost, control cost, and final cost matrices

$$Q = Q^T \geq 0, \quad R = R^T > 0, \quad Q_f = Q_f^T \geq 0$$

N is called the **horizon** of the problem. Note the final state cost.

Finite-time LQR via least-squares

Note that $X = (x_0, \ldots, x_N)$ is a linear function of x_0 and $U = (u_0, \ldots, u_{N-1})$

$$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ B & 0 & \cdots & 0 \\ AB & B & 0 & \cdots \\ \vdots & \vdots & & \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} + \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}$$

Can express as

$$X = GU + Hx_0$$

where $G \in \mathbb{R}^{Nn \times Nm}, \ H \in \mathbb{R}^{Nn \times n}$

Finite-time LQR via least-squares

Can express finite-horizon cost as

$$J(U) = X^T \underbrace{\begin{bmatrix} Q_1 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & \vdots \\ \vdots & 0 & Q_1 & 0 \\ 0 & \cdots & 0 & Q_f \end{bmatrix}}_{\overline{Q}_1} X + U^T \underbrace{\begin{bmatrix} Q_2 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & \vdots \\ \vdots & 0 & Q_2 & 0 \\ 0 & \cdots & 0 & Q_2 \end{bmatrix}}_{\overline{Q}_2} U =$$

$$= (GU + Hx_0)^T \overline{Q}_1 (GU + Hx_0) + U^T \overline{Q}_2 U =$$

$$= U^T (G^T \overline{Q}_1 G + \overline{Q}_2) U + 2x_0^T H^T \overline{Q}_1 GU + x_0^T H^T \overline{Q}_1 H x_0 =$$

$$:= U^T P_{LQ} U + 2q_{LQ}^T U + r_{LQ}$$

so optimal control is

$$U^\star = -P_{LQ}^{-1} q_{LQ}$$

for which

$$J(U^\star) = r_{LQ} - q_{LQ}^T P_{LQ}^{-1} q_{LQ}$$

Mikael Johansson mikaelj@ee.kth.se

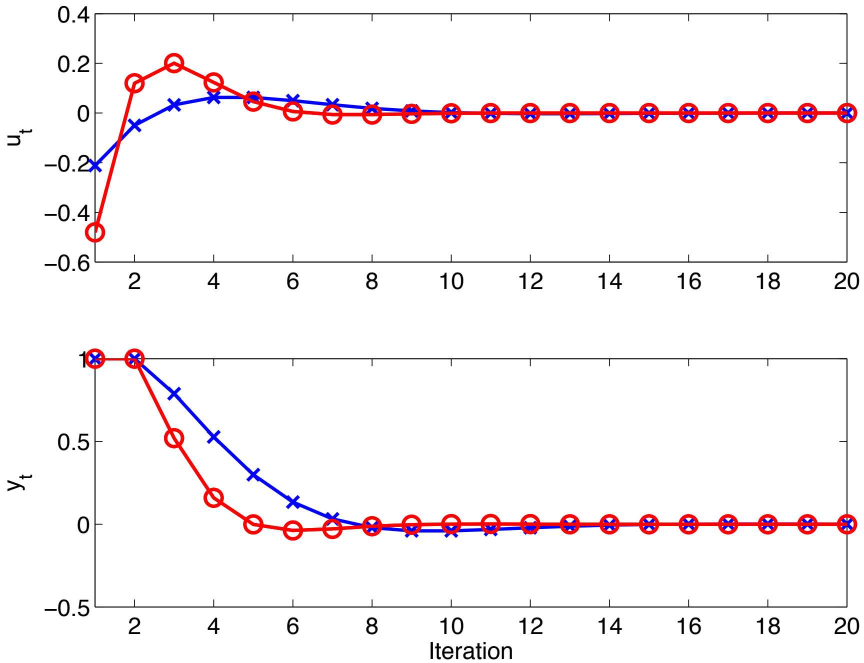
Example

LQR problem for system

$$x_{k+1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_t + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_t, \qquad C = \begin{bmatrix} 1 & 0 \end{bmatrix} x_t$$

$$Q_1 = Q_f = C^T C, \qquad R = \rho I$$

with horizon length 20. Results for $\rho = 10$ (blue) and $\rho = 1$ (red)



Mikael Johansson mikaelj@ee.kth.se

LQR via dynamic programming

Optimal LQ control can be found recursively using **Dynamic Programming**

For $t = 0, \ldots, N$ define the **value function** $V_t : \mathbb{R}^n \mapsto \mathbb{R}$ by

$$V_k(z) = \min_{u_k, \cdots, u_{N-1}} \sum_{t=k}^{N-1} \left(x_t^T Q_1 x_t + u_t^T Q_2 u_t\right) + x_N^T Q_f x_N$$

$$\text{subject to} \quad x_{t+1} = A x_t + B u_t, \; x_k = z$$

$V_k(z)$ gives the minimum LQR cost-to-go, starting from state z at time k

Note that

- $V_0(x_0)$ is the minimal LQR cost (from state x_0 at time 0)
- the cost-to-go with no time left is the quadratic final state cost

$$V_N(z) = z^T Q_f z$$

Dynamic programming principle

Assume that we know $V_{t+1}(z)$, what is the optimal choice for u_t ?

The choice of u_t affects

- cost incurred in current step (through $u_t^T Q_2 u_t$)
- the next state x_{t+1} (hence, the cost-to-go from x_{t+1})

Dynamic programming (DP) principle

$$V_t(z) = \min_w \left(z^T Q_1 z + w^T Q_2 w + V_{t+1}(Az + Bw) \right)$$

Follows from the fact that we can minimize in any order

$$\min_{w_1,\dots,w_k} f(w_1,\dots,w_k) = \min_{w_1} \underbrace{\left(\min_{w_2,\dots,w_k} f(w_1,\dots,w_k) \right)}_{\text{a function of } w_1}$$

Hamilton-Jacobi-Bellman equation

The recursion

$$V_t(z) = z^T Q_1 z + \min_w \left(w^T Q_2 w + V_{t+1}(Az + Bw) \right)$$

is called the Dynamic Programming, Bellman or Hamilton-Jacobi equation

Any minimizing w gives optimal control at time t

$$u_t^\star = \operatorname*{argmin}_w \left(w^T Q_2 w + V_{t+1}(Az + Bw) \right)$$

The HJB equation for LQR

Assume that $V_{t+1}(z) = z^T P_{t+1} z$ for some $P_{t+1} = P_{t+1}^T \geq 0$ (holds for $t+1 = N$)

Then,

$$\begin{aligned} V_t &= z^T Q_1 z + \min_w \left(w^T Q_2 w + (Az + Bw)^T P_{t+1} (Az + Bw) \right) = \\ &= z^T Q_1 z + \min_w \left(w^T (Q_2 + B^T P_{t+1} B) w + 2 z^T A^T P_{t+1} B w + z^T A^T P_{t+1} A z \right) = \\ &= z^T \left(Q_1 + A^T P_{t+1} A - A^T P_{t+1} B (Q_2 + B^T P_{t+1} B)^{-1} B P_{t+1} A \right) z := z^T P_t z \end{aligned}$$

with optimal control

$$u_t^\star = -(Q_2 + B^T P_{t+1} B)^{-1} B^T P_{t+1} A x_t$$

Mikael Johansson mikaelj@ee.kth.se

Summary of LQR via DP

1. set $P_N = Q_f$

2. for $t = N, N-1, \ldots, 1$

$$P_{t-1} := Q_1 + A^T P_t A - A^T P_t B (Q_2 + B^T P_t B)^{-1} B^T P_t A$$

3. for $t = 0, 1, \ldots, N-1$

$$L_t := (Q_2 + B^T P_{t+1} B)^{-1} B^T P_{t+1} A$$
$$u_t^\star = -L_t x_t$$

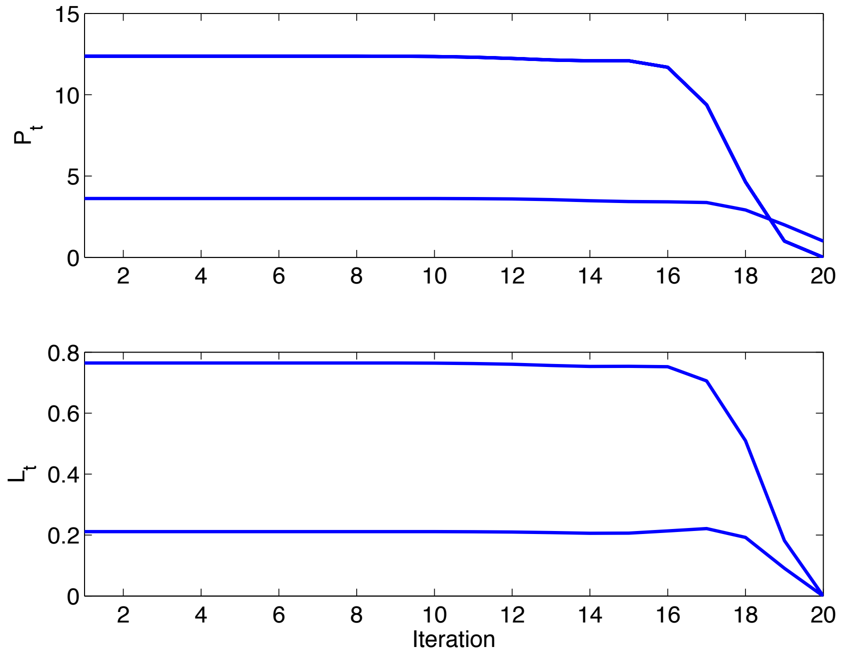
Notes:

- optimal control is a linear function of the state
- recursion for minimum cost-to-go runs backwards in time

Mikael Johansson mikaelj@ee.kth.se

Example

Same system as earlier. Investigate how elements of P and L converge



Rapid convergence to stationarity as t drops below horizon N!

Steady-state regulator

Usually, P_t converges rapidly as t decreases below N

The stationary solution satisfies

$$P = Q_1 + A^T PA - A^T PB(Q_2 + B^T PB)^{-1} B^T PA$$

(called the discrete-time algebraic Riccati equation)

When N is large, and t is not too close to N, optimal input approaches

$$u_t = -Lx_t \qquad L = (Q_2 + B^T PB)^{-1} B^T PA$$

(perfect agreement when N is infinite). A linear state feedback!

Receding horizon LQR

Consider the cost function

$$J_k(u_k, \ldots, u_{k+K-1}) = \sum_{t=k}^{k+K-1} (x_t^T Q_1 x_t + u_t^T Q_2 u_t) + x_{k+K}^T Q_f x_{k+K}$$

Here, K is called the **horizon**, and if

$$(u_k^\star, \ldots, u_{k+K-1}^\star)$$

minimizes J_k, then $u_k^\star$ is called **K-step optimal receding horizon control**

Receding-horizon control:

- at time k, find input sequence that minimizes K-step ahead LQR cost (starting at time k)
- then apply only the first element of the input sequence

Closed-loop system

If horizon tends to infinity, then control coincides with stationary LQR

- closed-loop system

$$x_{t+1} = (A - BL)x_t$$

  stable under mild conditions ((A,B) controllable, (Q,A) observable)

In general, optimal K-step ahead LQR control is time-varying and closed-loop not necessarily stable if horizon too short.

Example

Consider a discrete-time system with

$$A = \begin{bmatrix} 1 & 1 \\ 2 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

and the the receding horizon control problem given by

$$Q_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad Q_2 = 1, \quad Q_f = Q_1$$

The one-step optimal receding-horizon control is

$$u_t = -\underbrace{(1 + B^T B)^{-1} B^T A}_{L} x = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

which yields an unstable closed-loop system

$$x_{t+1} = (A - BL)x_t = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} x_t$$

Closed-loop system cont'd

Two ways to ensure closed-loop stability:

1. use a different terminal cost. In particular $Q_f = P$ where

   $$P = Q_1 + A^T PA - A^T PB(Q_2 + B^T PB)^{-1} B^T PA$$

   (i.e. P solves the discrete-time ARE) ensures stability.

   Why? Receding horizon-control is then (independent of t)

   $$u_t = -Lx_t \qquad L = (Q_2 + B^T PB)^{-1} B^T PA$$

   and the associated closed-loop system is stable since observability and controllability conditions are met.

2. Use longer horizon, so that control approaches stationary optimal

Mikael Johansson mikaelj@ee.kth.se

Summary

"Sampling of systems"

- convert continuous-time system model to discrete-time
- states agree at sampling instances (if control is held constant)

The finite-time linear quadratic regulator

- by quadratic optimization or dynamic programming
- stationary solution and infinite-horizon LQR

Receding-horizon principle

- compute optimal input sequence over finite-time horizon, apply first element in sequence, re-optimize next sampling instant
- stability if sufficient horizon, or "right" terminal constraint

Next week: model predictive control (receding horizon with constraints)

Mikael Johansson mikaelj@ee.kth.se

Acknowledgements

Slides borrow heavily from Stanford EE363 Winter 2008-2009 (Prof. Boyd).