LECTURE 10 & 11 : 31st Jan

Optimal control of discrete-time systems.

Consider a discrete-time dynamical system is state-space form:

$$\underline{x_{k+1} = f^k(x_k, u_k)}\,, \quad \text{where} \quad x_k \in \mathbb{R}^n : \text{state at time } k$$

$u_k \in \mathbb{R}^m$: input at time k

initial state x_0 is known.

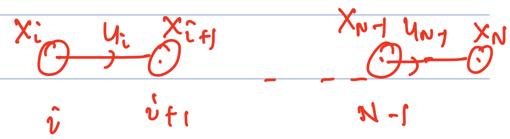
If f^k does not depend on time k, the system is time-invariant.
Otherwise it is time-varying.

If $f^k = A^k x_k + B^k u_k$, we say that the system is linear.

The goal of optimal control is to compute a sequence of inputs $(u_0, u_1, \ldots u_{N-1})$ such that the applied input and the resulting state trajectory $(x_0, x_1, x_2, \ldots x_N)$ together minimize some cost function or performance metric.

$$U_{i:N-1} = (u_i, u_{i+1}, \cdots u_{N-1})$$

$$X_{i:N} = (x_i, x_{i+1}, \cdots x_N)$$



$$J(U_{i:N-1}, X_{i:N}) = \underbrace{\phi(N, x_N)}_{\text{terminal cost}} + \sum_{k=i}^{N-1} \underbrace{L^k(x_k, u_k)}_{\text{stage cost}}$$

The overall optimization problem is written as **(DT-OPT)**

$$\min_{U_{i:N-1}, X_{i:N}} \quad \phi(N, x_N) + \sum_{k=i}^{N-1} L^k(x_k, u_k)$$

$$\text{s.t.} \quad \lambda_{k+1} : x_{k+1} = f^k(x_k, u_k), \quad k = i, i+1, \ldots, N-1$$

Given the above formulation, there are two paradigms that one can adopt.

(A) Solve the above problem using a suitable numerical optimization solver.

this philosophy forms the basis of model predictive control.

- (−) we require on board compute to numerically solve the above problem
- (−) the algorithm needs to converge to optimal solution quickly.
- (−) structure of the optimal control input is not exploited.
- (+) other constraints on states and inputs can be imposed.
- (−) we cannot handle infinite-horizon optimization problems.

(B) Synthesize optimal controller exploiting the optimality conditions. (offline)

- (−) for the most part, we will focus on problems without any constraints other than the dynamics.
- (+) exploit the structure of the problem
- (+) in some cases, we can analyze infinite-horizon problems $(N \to \infty)$

Two main analysis techniques:

(i) Calculus of variations

(ii) Dynamic Programming.

Let us start with the simplest form that we can think of.

LTI system: $x_{k+1} = Ax_k + Bu_k$

cost function: $\phi(N, x_N) = \frac{1}{2} x_N^T S x_N$

$L^k(x_k, u_k) = \frac{1}{2} x_k^T Q x_k + \frac{1}{2} u_k^T R u_k$

Let $N = i+1$.

Length of horizon : 1.

or $i=0$, $N=1$ $\qquad x_0$: known

$(Cx_k - x^{ref})^T Q (Cx_k - x^{ref})$ =

$$\begin{cases} \min\limits_{u_0, x_1} & \frac{1}{2} x_1^T S x_1 + \underbrace{\frac{1}{2} x_0^T Q x_0}_{\text{Known constant, can be dropped.}} + \frac{1}{2} u_0^T R u_0 \\ \text{s.t.} & x_1 = Ax_0 + Bu_0 \end{cases}$$

equivalently:

$$\min_{u_0} \frac{1}{2} \underbrace{[Ax_0 + Bu_0]^T S [Ax_0 + Bu_0]}_{x_0^T A^T S B u_0} + \frac{1}{2} u_0^T R u_0$$

equivalently:

$$\min_{u_0} \frac{1}{2} u_0^T (B^T S B) u_0 + \frac{1}{2} u_0^T R u_0 + x_0^T A^T S B u_0$$

$$= \min_{u_0} \frac{1}{2} u_0^T [R + B^T S B] u_0 + x_0^T A^T S B u_0$$

$$= \min_{x} \underbrace{\frac{1}{2} x^T G x + h^T x}_{f_0(x)}$$: unconstrained optimization problem

$\boxed{G = R + B^T S B}$ $\qquad$ let G: symmetric

First order necessary condition (FONC) $\nabla f_0(x) = 0$

$\Rightarrow Gx + h = 0 \quad \cdots (1)$

What can we say about solutions of (1) :

case (i): If G is invertible, $x^* = -G^{-1}h$ is the unique solution.

- if G is positive definite, x^* is a minimizer
- if G is negative " , x^* is a maximizer
- If G is indefinite, x^* is a saddle point

Recall that $G = R + \underline{B^T S B}$ $\quad \underline{y^T B^T} \underline{S} \underline{By} \geq 0$

- if $\underline{R \text{ is positive definite}}$ and $\underline{S \text{ is positive semidefinite}}$, then G is positive definite.

case (ii): Suppose G is not-invertible, i.e., $\underline{\det(G) = 0}$.

$\Rightarrow \exists\, \eta$ s.t. $G\eta = 0$

$\Rightarrow \eta^T G = 0$

(A) there is no solution to $Gx + h = 0$

(B) there is a solution $\bar{x}$ to $Gx + h = 0$

$\Rightarrow h = -G\bar{x}$

in case (B): the value of the cost function at $\bar{x}$ is

$$\frac{1}{2}\bar{x}^T G \bar{x} + h^T \bar{x} = -\frac{1}{2}\bar{x}^T G \bar{x}$$

if $\bar{x}$ satisfies $G\bar{x} + h = 0$,
then any $\bar{x} + \lambda\eta$ also satisfies $G(\bar{x} + \lambda\eta) + h = G\bar{x} + h = 0$,
$\lambda \in \mathbb{R}$

cost function $\frac{1}{2}(\bar{x} + \lambda\eta)^T G (\bar{x} + \lambda\eta) + h^T(\bar{x} + \lambda\eta)$

$$= \frac{1}{2}\bar{x}^T G \bar{x} + (-G\bar{x})^T(\bar{x} + \lambda\eta)$$

$$= \underline{-\frac{1}{2}\bar{x}^T G \bar{x}} - \underbrace{\bar{x}^T G \lambda\eta}_{=0}$$

$\Rightarrow$ If G is positive semidefinite $\bar{x}$ is minimizer.
There are infinitely many points satisfying FONC, but all with the same optimal value.

(A) there is no optimal solution

since $\det(G)=0$, $\exists\ \eta$ s.t. $G\eta=0$

we can choose η s.t. $\underline{h^T\eta<0}$.

[since if η s.t. $G\eta=0$, then $G(-\eta)=0$]

let $\lambda\eta$.

cost function: $\frac{1}{2}\underbrace{(\lambda\eta)^T G(\lambda\eta)}_{=0}+\underline{\lambda h^T\eta}\to-\infty$ when $\lambda\to\infty$

$\Rightarrow$ the optimization problem is unbounded, and there is no optimal solution.

Let us now examine the one step look ahead optimal control problem

$$\min_{u_0}\ \frac{1}{2}u_0^T\left[\underline{R+B^TSB}\right]u_0+\underline{(x_0^TA^TSB)}u_0=f(u_0)$$

Suppose R is symmetric, positive definite $R\succ 0$

S is ", positive semi-definite $S\succeq 0$

$R+B^TSB$ is invertible

$$\nabla f(u_0)=(R+B^TSB)u_0+B^TSAx_0$$

$$\underline{\nabla^2 f(u_0)}=\underline{R+B^TSB}\succ 0$$

the optimal input u_0^* satisfies $\nabla f(u_0^*)=0$

$$\Rightarrow (R+B^TSB)u_0^*+B^TSAx_0=0$$

$$\Rightarrow u_0^*=-\underbrace{(R+B^TSB)^{-1}B^TSA}x_0.$$

$$=-\underline{K^{opt}}x_0$$

$\Rightarrow$ optimal control input is a state-feedback control.

– there is no need to solve any optimization problem online. K^{opt} can be synthesized offline.

closed-loop dynamics: $x_{k+1} = (A - BK^{opt})x_k$

we would want $A - BK^{opt}$ should have all eigenvalues strictly inside unit circle, but that is only possible if (A, B) is stabilizable.

this relationship will be further explored in coming lectures.

The optimal value of the cost function is quadratic in x_0.

Before tackling the general (DT-OPT) problem, let us consider the following.

$$\begin{bmatrix} \min\limits_{x \in \mathbb{R}^n} & f(x) \\ \text{s.t.} & g_i(x) = 0, \quad i = 1, 2 \ldots m \end{bmatrix}$$

→ in optimization field, λ is used for multipliers of inequality constraints.

→ in optimal control field, λ is used as multipliers of equality constraints. which we will adopt.

Let λ_i be the multiplier of $g_i(x) = 0$

Let us define the Lagrangian:

$$L(x, \lambda) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x)$$

$$= f(x) + \lambda^T g(x), \quad g(x) = \begin{bmatrix} g_1(x) \\ g_2(x) \\ \vdots \\ g_m(x) \end{bmatrix} = 0$$

dual function:

$$\underline{d(\lambda)} = \inf_{x \in \mathbb{R}^n} \left[\underline{f(x) + \lambda^T g(x)} \right]$$

$$\leq f(\bar{x}) + \lambda^T g(\bar{x})$$

$$= \underline{f(\bar{x})}$$

for any feasible $\bar{x}$ which satisfies $g(\bar{x}) = 0$.

[If x^* is the minimizer of $L(x, \lambda)$ at λ^*, then x^* is an optimal solution]

FONC: If (x^*, λ^*) s.t. $\underline{\nabla_x L(x^*, \lambda^*) = 0}$

$x^* \in \mathbb{R}^n$, $\lambda^* \in \mathbb{R}^m$

$$\Rightarrow \left.\begin{aligned} & \underbrace{\nabla f(x^*) + (\lambda^*)^T \nabla g(x^*)}_{\in \mathbb{R}^n} = 0 \\ & \nabla_\lambda L(x^*, \lambda^*) = \underbrace{g(x^*)}_{\in \mathbb{R}^m} = 0 \end{aligned}\right\} \cdots (2)$$

For sufficiency,

$$\nabla_x^2 L(x, \lambda) = \underline{\nabla_x^2 f(x)} + \underline{\lambda^T \nabla_x^2 g(x)}$$

Note: If the optimization problem is convex, $\nabla_x^2 f(x) \succeq 0 \ \forall x$, and if we can find (x^*, λ^*) satisfying (2), then the corresponding x^* is a minimizer.

The goal is to solve for (x^*, λ^*) satisfying

$$\nabla_x L(x^*, \lambda^*) = \nabla f(x^*) + (\lambda^*)^T \nabla g(x^*) = 0$$

$$\nabla_\lambda L(x^*, \lambda^*) = \underline{g(x^*) = 0}$$

Lecture 12 & 13 : 2nd Feb.

The overall optimization problem is written as (DT-OPT)

$$\min_{u_{i:N-1},\, x_{i:N}} \quad \phi(N, x_N) + \sum_{k=i}^{N-1} L^k(x_k, u_k)$$

$$\text{s.t.} \quad \lambda_{k+1} : x_{k+1} = f^k(x_k, u_k), \quad k = i, i+1, \ldots, N-1$$

$f^k(x_k,u_k) - x_{k+1} = 0$

Lagrangian of the above problem is given by:

$$\sum_{k=i}^{N-1} \lambda_{k+1}^T x_{k+1} = \sum_{k=i+1}^{N} \lambda_k^T x_k$$

$$L(u_i, u_{i+1}, \ldots u_{N-1}, x_i, x_{i+1} \ldots x_N, \lambda_{i+1}, \lambda_{i+2} \ldots \lambda_N)$$

$$= \phi(N, x_N) + \sum_{k=i}^{N-1} \left[L^k(x_k, u_k) + \lambda_{k+1}^T \left(f^k(x_k, u_k) - x_{k+1} \right) \right]$$

$$= \phi(N, x_N) + \sum_{k=i+1}^{N-1} \left[L^k(x_k, u_k) + \lambda_{k+1}^T f^k(x_k, u_k) - \lambda_k^T x_k \right] - \lambda_N^T x_N + L^i(x_i, u_i) + \lambda_{i+1}^T f^i(x_i, u_i)$$

$$= \phi(N, x_N) - \lambda_N^T x_N + \underbrace{L^i(x_i, u_i) + \lambda_{i+1}^T f^i(x_i, u_i)} + \sum_{k=i+1}^{N-1} \Big[\underbrace{L^k(x_k, u_k) + \lambda_{k+1}^T f^k(x_k, u_k)}_{\text{Hamiltonian } H^k(x_k, u_k; \lambda_{k+1})} - \lambda_k^T x_k \Big]$$

Suppose $u^*_{i:N-1}, x^*_{i:N-1}, \lambda^*_{i+1:N}$ satisfy FONC for optimality. Then, the following are true.

$$\nabla_{\lambda_{k+1}} L = 0 \Rightarrow \boxed{f^k(x^*_k, u^*_k) = x^*_{k+1}}$$

$$\nabla_{u_k} L = 0 \Rightarrow \boxed{\nabla_{u_k} L^k(x^*_k, u^*_k) + (\lambda^*_{k+1})^T \nabla f_{u_k}(x^*_k, u^*_k) = 0, \quad \forall k = i, i+1, \ldots, N-1}$$

$$\nabla_{u_k} H^k = \frac{\partial H^k}{\partial u_k}(x_k, u_k; \lambda_{k+1}) = 0$$

$$\begin{cases} \nabla_{x_N} L = 0 \Rightarrow \underline{[\nabla \phi(N, x_N^*) - \lambda_N^*]} \, dx_N^* = 0 \\ \nabla_{x_i} L = 0 \Rightarrow \underline{[\nabla_{x_i} L^i(x_i^*, u_i^*) + (\lambda_{i+1}^*)^T \nabla_{x_i} f^i(x_i^*, u_i^*)]} \, dx_i^* = 0 \end{cases}$$

boundary conditions

$\nabla_{x_k} H^k(x_k, u_k; \lambda_{k+1})$

$$\nabla_{x_k} L = 0 \Rightarrow \boxed{\nabla_{x_k} L^k(x_k^*, u_k^*) + (\lambda_{k+1}^*)^T f^k(x_k^*, u_k^*) = \lambda_k^*}$$

CLASS TEST - I

1. Consider the function $f(x_1, x_2) = \frac{1}{2}x_1^2 + x_1 x_2 + x_2^2 + x_2$. Find all the critical points of this function and check if they are minimizers or maximizers.

2. Consider the problem
$$\min_{x \in \mathbb{R}^3} \frac{1}{2}(x_1^2 + x_2^2 + x_3^2)$$
$$\text{s.t.} \quad x_1 + x_2 + x_3 = 3$$
(i) Write the KKT conditions and find all points that satisfy it.
(ii) Determine an optimal solution of the above problem and see if there exists an unique optimal solution.

3. For the problem $\min_{x \in \mathbb{R}^n} f(x)$, with f being a convex function, show that $\nabla f(x) = 0$ is a sufficient condition of optimality.

Soln: If f is a convex function

$$f(y) \geq f(x) + \nabla f(x)^T (y - x). \quad \forall x, y \in \text{domain}$$

If x^* s.t. $\nabla f(x^*) = 0 \Rightarrow f(y) \geq f(x^*) \quad \forall y.$

Let us examine a specific problem.

we have a LTI system: $x_{k+1} = Ax_k + Bu_k$,

i.e., $f^k(x_k, u_k) = Ax_k + Bu_k$

Final state $x_N = r_N$ (fixed final state, r_N: reference)

Initial state x_i = fixed and given

cost function $J = \frac{1}{2}\sum_{k=i}^{N-1} u_i^T R\, u_i$,

i.e., $\phi(N, x_N) = 0$, $L^k(x_k, u_k) = \frac{1}{2} u_k^T R u_k$

- R is a positive definite
- "drive the state from initial state x_i to a final state x_n with minimal control effort"

If $R = I$, then cost function $J = \frac{1}{2}\sum_{k=i}^{N-1} u_i^T u_i$,

which is the energy of the input signal over time from i to $N-1$.

optimal control problem:

$$\min_{\substack{u_i, u_{i+1}, \ldots, u_{N-1} \\ x_{i+1}, \ldots, x_N}} \frac{1}{2}\sum_{k=i}^{N-1} u_i^T R u_i$$

s.t. $x_{k+1} = Ax_k + Bu_k$, $k = i, i+1, \ldots, N-1$

$x_N = r_N$

Let us write down the optimality conditions.

$L(u_{i:N-1}, x_{i+1:N}, \lambda_{i+1:N})$

$$= \sum_{k=i+1}^{N-1}\left(\frac{1}{2}u_k^T R u_k + \lambda_{k+1}^T\left[Ax_k + Bu_k\right] - \lambda_k^T x_k\right) - \lambda_N^T x_N$$

$+ \frac{1}{2} u_i^T R u_i + \lambda_{i+1}^T [A x_i + B u_i]$

Let $(u^*_{i,N-1}, x^*_{i:N}, \lambda^*_{i+1:N})$ satisfy FONC of optimality.

Then, $\nabla_{\lambda_{k+1}} L = 0 \Rightarrow x^*_{k+1} = A x^*_k + B u^*_k, \quad k = i, i+1 \dots N-1$

$\nabla_{u_k} L = 0 \Rightarrow R u_k + (\lambda_{k+1}^T B)^T = 0$

$\Rightarrow u^*_k = -R^{-1} B^T \lambda^*_{k+1}$

$\nabla_{x_k} L = 0 \Rightarrow \lambda_k = A^T \lambda_{k+1}$ $\quad \nabla_{x_k}\left(-\lambda_k^T x_k + (A^T \lambda_{k+1})^T x_k\right)$

We obtain:

state dynamics (forward)

$$\boxed{\begin{aligned} x^*_{k+1} &= A x^*_k - B R^{-1} B^T \lambda^*_{k+1} \quad &-(a) \\ \lambda^*_k &= A^T \lambda^*_{k+1} \quad &-(b) \end{aligned}}$$

λ_k: co-states

co-state dynamics (backward)

Boundary conditions: initial state x_i
final state $x_N = r_N$ (known)

$\boxed{\text{From now on, * is dropped and assume } i = 0}$

From (b):

$\lambda_{N-1} = A^T \lambda_N$

$\lambda_{N-2} = (A^T)^2 \lambda_N$

$\vdots$

$\lambda_k = (A^T)^{N-k} \lambda_N, \quad \lambda_0 = (A^T)^N \lambda_N$

$\lambda_{i+1} = (A^T)^{N-i-1} \lambda_N$

Recall: for a DT linear system, $x_1 = A x_0 + B u_0$

$x_2 = A^2 x_0 + A B u_0 + B u_1$

$x_k = A^k x_0 + \sum_{i=0}^{k-1} (A)^{k-1-i} B u_i$

Thus,

$$r_N = x_N = A^N x_0 + \sum_{i=0}^{N-1} (A)^{N-i-1} B\, u_i$$

$$= A^N x_0 + \sum_{i=0}^{N-1} (A)^{N-i-1} B \left(-R^{-1} B^T \lambda_{i+1}\right)$$

$$= A^N x_0 - \underbrace{\sum_{i=0}^{N-1} (A)^{N-i-1} B\, R^{-1} B^T (A^T)^{N-i-1}}_{G_{0,N-1}} \lambda_N$$

$M_0 \lambda_N + M_1 \lambda_N + \cdots$

$\left(\sum_i M_i\right) \lambda_N$

Given x_N and x_0, the only unknown is λ_N.

Thus, $G_{0,N-1} \lambda_N = -\left(r_N - A^N x_0\right)$

Observe that $G_{0,N-1} = \underbrace{\begin{bmatrix} B & AB & A^2B & \cdots & A^{N-1}B \end{bmatrix}}_{\substack{\text{Reachability or} \\ \text{controllability matrix} \\ \text{of order } N \,=\, C_N}} \begin{bmatrix} R^{-1} & & & 0 \\ & R^{-1} & & \\ & & \ddots & \\ 0 & & & R^{-1} \end{bmatrix} \begin{bmatrix} B^T \\ (AB)^T \\ \vdots \\ (A^{N-1}B)^T \end{bmatrix}$

$\Rightarrow G_{0,N-1} = C_N \begin{bmatrix} R^{-1} & & \\ & \ddots & \\ & & R^{-1} \end{bmatrix} C_N^T$: square matrix with number of rows $= n$, where n is state-dimension

↳ invertible if C_N has full row-rank

Thus, $G_{0,N-1}$ is guaranteed to be invertible if $N \geq n$ and the pair (A, B) is controllable.

If $G_{0,N-1}$ is invertible, $\lambda_N^* = -G_{0,N-1}^{-1}\left(r_N - A^N x_0\right)$.

(Unique sol^n)

state at time N when input is zero.

$\Rightarrow$ find all λ_k^*, $k < N$

$\Rightarrow$ find all u_k^*, $k < N$

optimal input at time k

$$u_k^* = -R_k^{-1} B^T \lambda_{k+1}^* = +R_k^{-1} B^T \underbrace{(A^T)^{N-k-1} G_{0,N-1}^{-1}}_{\text{depends on } k} \underbrace{(r_N - A^N x_0)}_{\substack{\text{error of desired final state} \\ \text{\& final state under natural response}}}$$

It is easy to verify that x_N under the above control law, will be equal to r_N.

Lecture 14th & 15th : 3rd Feb.

Discrete-time Linear Quadratic Regulator (LQR) (Finite-horizon)

$$\min_{u_{i:N-1}, x_{i:N}} \quad \frac{1}{2} x_N^T S_N x_N + \frac{1}{2} \sum_{k=i}^{N-1} \left[x_k^T Q_k x_k + u_k^T R_k u_k \right]$$

$$\text{s.t.} \quad x_{k+1} = A_k x_k + B_k u_k, \quad k = i, i+1, \ldots, N-1$$

$$x_i : \text{given / known}$$

Note: (i) DT system is linear, but possibly time-varying

(ii) $R_k \succ 0, \; Q_k \succeq 0 \quad \forall \, k = i, \ldots N-1, \quad S_N \succeq 0.$

(iii) Final state x_N is free.

The matrices S_N, Q_k, R_k are chosen by the practioner to prioritize either states or inputs; often chosen after some trial and error steps.

$$L(u_{i:N-1}, x_{i:N}, \lambda_{i+1:N}) = \frac{1}{2} x_N^T S_N x_N - \lambda_N^T x_N + \frac{1}{2} x_i^T Q_i x_i + \frac{1}{2} u_i^T R_i u_i + \lambda_{i+1}^T (A_i x_i + B_i u_i)$$

$$+ \frac{1}{2} \sum_{k=i+1}^{N-1} \left[x_k^T Q_k x_k + u_k^T R_k u_k + \lambda_{k+1}^T (A_k x_k + B_k u_k) - \lambda_k^T x_k \right]$$

FONC : (these conditions are also sufficient since DT-OPT is convex)

$\nabla_{u_k} L = R_k u_k + B_k^T \lambda_{k+1} = 0 \Rightarrow \boxed{u_k^* = -R_k^{-1} B_k^T \lambda_{k+1}^*}$

$\nabla_{x_N} L = S_N x_N - \lambda_N = 0 \Rightarrow \lambda_N^* = S_N x_N^*$

$\nabla_{x_k} L = Q_k x_k^* + A_k^T \lambda_{k+1}^* = \lambda_k^*, \quad k = i+1, \ldots, N-1$

$\nabla_{\lambda_k} L = x_{k+1}^* = A_k x_k^* + B_k u_k^*, \quad k = i \ldots, N-1$

$$(a) \cdots \left\{ \begin{aligned} x_{k+1} &= A_k x_k - B_k R_k^{-1} B_k^T \lambda_{k+1} \\ \lambda_k &= Q_k x_k + A_k^T \lambda_{k+1} \end{aligned} \right. \cdots (b)$$

, initial conditions
x_i : known
$\lambda_N = S_N x_N$

two point boundary value problems which are very difficult to solve in general

we hypothesize that $\lambda_k = S_k x_k \quad \forall k = i \ldots N-1$ for a suitable (symmetric) matrix S_k.

Then, (a) yields

$$x_{k+1} = A_k x_k - B_k R_k^{-1} B_k^T \left[S_{k+1} x_{k+1} \right]$$

$$\Rightarrow \left[I + B_k R_k^{-1} B_k^T S_{k+1} \right] x_{k+1} = A_k x_k$$

$$\Rightarrow x_{k+1} = \left[I + B_k R_k^{-1} B_k^T S_{k+1} \right]^{-1} A_k x_k \quad \text{--- (c)}$$

Substituting the above in (b), we obtain

$$S_k x_k = Q_k x_k + A_k^T S_{k+1} x_{k+1}$$

$$\Rightarrow S_k x_k = Q_k x_k + A_k^T S_{k+1} \left[I + B_k R_k^{-1} B_k^T S_{k+1}\right]^{-1} A_k x_k$$

$$\Rightarrow \boxed{S_k = Q_k + A_k^T S_{k+1} \left[I + B_k R_k^{-1} B_k^T S_{k+1}\right]^{-1} A_k} \quad (d)$$

— backward recursion, can be computed offline.

Since S_N is known from problem data, the values of S_k, $k \in \{0, 1, \dots N-1\}$ can be calculated from above relation.

The backward recursion (d) has several other forms that can be obtained using matrix inversion lemma. The following form is often used in numerical computations.

$$S_k = Q_k + A_k^T \left[S_{k+1} - S_{k+1} B_k \left(B_k^T S_{k+1} B_k + R_k\right)^{-1} B_k^T S_{k+1}\right] A_k .$$

Let us now focus on the optimal control input:

$$u_k = -R_k^{-1} B_k^T S_{k+1} x_{k+1} = -R_k^{-1} B_k^T S_{k+1} \left[A_k x_k + B_k u_k\right]$$

$$\Rightarrow \left(I + R_k^{-1} B_k^T S_{k+1} B_k\right) u_k = -R_k^{-1} B_k^T S_{k+1} A_k x_k$$

multiplying R_k on both sides, we obtain

$$\left(R_k + B_k^T S_{k+1} B_k\right) u_k = -B_k^T S_{k+1} A_k x_k$$

$$\Rightarrow u_k = -\underbrace{\left(R_k + B_k^T S_{k+1} B_k\right)^{-1} B_k^T S_{k+1} A_k}_{=: K_k} x_k = -K_k x_k$$

<u>Offline computations</u>

- Given S_N, at step $k = N-1, N-2 \dots 0$, compute:

$$K_k = \underline{\left(R_k + B_k^T S_{k+1} B_k\right)^{-1} B_k^T S_{k+1} A_k}$$

$$S_k = \underline{A_k^T S_{k+1} \left(A_k - B_k K_k\right) + Q_k}$$

during online deployment

at time 0, $u_0 = -K_0 x_0$ computed beforehand; observed state at time 0

⋮

k; $u_k = -K_k x_k$ observed state at time k.

Fact: The optimal cost $J_i^* = \frac{1}{2} x_i^T S_i x_i$.

Infinite-horizon LQR for DT-LTI systems

Consider the backward recursion step:

$$S_k = Q_k + A_k^T \left[S_{k+1} - S_{k+1} B_k \left(B_k^T S_{k+1} B_k + R_k \right)^{-1} B_k^T S_{k+1} \right] A_k$$

Suppose N is extremely large.

(Q) does the sequence S_k converge? i.e., $S_k \simeq S_{k+1}$ if $k \ll N$.

(Q) If so, is the resulting S_k symmetric? positive definite? positive semi-definite?

(Q) what are properties of the resulting control gain?

Assumptions: A_k, B_k, Q_k, R_k do not depend on k.

If S_k converges, i.e., $S_k = S_{k+1}$, then the limit S_∞ must satisfy

$$S_\infty = Q + A^T \left[S_\infty - S_\infty B \left(B^T S_\infty B + R \right)^{-1} B^T S_\infty \right] A.$$

Discrete Algebraic Riccati Equation (D-ARE).

(Q) When does D-ARE have a solution? Is the solution unique? does $S_k \to S_\infty$?

(Q) If we choose $K_\infty = \left(R + B^T S_\infty B \right)^{-1} B^T S_\infty A$, and apply $u_k = -K_\infty x_k$ $\forall k$, then is the closed-loop system $x_{k+1} = (A - BK_\infty) x_k$ stable?

Static state-feedback controller.

Recall the following properties of LTI systems: $\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k + Du_k \end{aligned}$

(i) If (A,B) is controllable, then we can find a gain K such that all the eigenvalues of $(A-BK)$ can be assigned at any desired locations of the complex plane.

(ii) (A,B) is stabilizable $\Leftrightarrow$ $\exists$ gain K s.t. all the eigenvalues of $(A-BK)$ lie strictly inside unit circle.

$\Rightarrow$ $x_{k+1} = (A-BK)x_k$ is globally asymptotically stable.

(iii) If (A,C) is observable, then we can find L s.t. all the eigenvalues of $(A-LC)$ can be arbitrarily assigned.

(iv) (A,C) is detectable $\Leftrightarrow$ $\exists$ gain L s.t. all the eigenvalues of $(A-LC)$ lie strictly inside unit circle.

Theorem: (1) If (A,B) is stabilizable, then for every S_N, there exists a bounded limiting solution S_∞ of D-ARE and S_∞ is positive semidefinite.

(2) Let $\sqrt{Q} = C$, i.e., $Q = C^TC$. Let (A,C) be observable. Then,
(A,B) is stabilizable $\Leftrightarrow$ $\exists$ unique soln S_∞ of D-ARE, $S_\infty > 0$
$x_{k+1} = (A-BK_\infty)x_k$ is GAS.

(3) Let (A,C) be detectable. Then,
(A,B) is stabilizable $\Leftrightarrow$ $\exists$ unique soln S_∞ of D-ARE, $S_\infty \geq 0$
Closed-loop system $x_{k+1} = (A-BK_\infty)x_k$ is GAS.

Take home message: Given problem data, check whether (A,B) is stabilizable, $(A,\sqrt{Q})$ is detectable. If yes, solve for S_∞, use $u_k = -K_\infty x_k$ as input. The resulting closed-loop system has origin as GAS.

Recall: Detectability is the minimum requirement for Luenberger observer design.

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + L\left[y_k - C\hat{x}_k\right] \qquad \begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k \end{aligned}$$

$$e_k := x_k - \hat{x}_k$$

$$e_{k+1} = (A-LC)e_k$$

The above idea is relevant for optimal control because

$$\frac{1}{2}x_k^T Q x_k + \frac{1}{2}u_k^T R u_k = \underline{\frac{1}{2}y_k^T y_k}, \text{ where}$$

$$y_k = \begin{bmatrix} \sqrt{Q} \\ 0 \end{bmatrix} x_k + \begin{bmatrix} 0 \\ \sqrt{R} \end{bmatrix} u_k$$

- If $(A,\sqrt{Q})$ is not detectable, then there are unstable modes that are not present in the output y_k.
- minimizing $\frac{1}{2}y_k^T y_k$ will not affect the states that correspond to unstable modes.
- So we can ensure that all the states are being driven to origin.

Infinite-horizon LQR:

$$\min_{\substack{u_0, u_1, \dots \\ x_1, x_2, \dots}} \quad \frac{1}{2} \sum_{k=0}^{\infty} \left[x_k^T Q x_k + u_k^T R u_k \right]$$

$$\text{s.t.} \quad x_{k+1} = A x_k + B u_k, \quad k \geq 0.$$

Optimal solution: $\underline{u_k = -K_\infty x_k}$ assuming

(A, B) stabilizable,

$(A, \sqrt{Q})$ detectable.

Optimal value: $\frac{1}{2} x_0^T S_\infty x_0$

Recap:

- Formulated discrete-time optimal control over finite time horizon
- Derived first-order necessary conditions of optimality
  - case 1: Fixed terminal state : $(\dot{K}(\pi_N - A^N x_0))$
  - case 2: Free terminal state : $(u_k = K_k x_k)$
- Specialized to Linear systems with quadratic cost f^n such that the overall problem is convex.
  - $\Rightarrow$ FONC are sufficient
- Specialized to LTI systems, infinite horizon under suitable assumptions on A, B, Q.

linear state feedback is a consequence of quadratic cost & linear dynamics.

$$\left\{ \begin{array}{l} L = \frac{1}{2} u^T R u + u^T A x \\ \nabla_u = R u + A x \Rightarrow u^* = -R^{-1} A x \end{array} \right.$$

LECTURES 16 & 17 : 9th Feb.

Alternative choice of cost functions

(A) minimum "fuel" control : $J_i \doteq \sum_{k=i}^{N-1} |u_i|$

(B) minimum "time" :

$$\begin{cases} \min\limits_{u_i, \ldots u_{N-1}} & \boxed{N} \\ \text{s.t.} & x_{k+1} = f^k(x_k, u_k), \\ & x_i : \text{given} \\ & x_N = r_N \text{ given} \end{cases}$$

(C) Tracking problem : given $r_0, r_1, r_2 \ldots r_N$, and we wish to find $(u_0, u_1, \ldots u_{N-1})$ s.t. the output $y_k = h^k(x_k, u_k)$ is "close to" r_k.

$$\begin{cases} \min\limits_{\substack{u_{0:N-1} \\ x_{0:N}}} & \frac{1}{2}(y_N - r_N)^T S_N (y_N - r_N) + \sum_{k=i}^{N-1} \frac{1}{2}(y_k - r_k)^T Q (y_k - r_k) + \frac{1}{2} u_k^T R u_k \\ \text{s.t.} & x_{k+1} = f^k(x_k, u_k) \\ & y_k = h^k(x_k, u_k) \\ & x_i : \text{given} \end{cases}$$

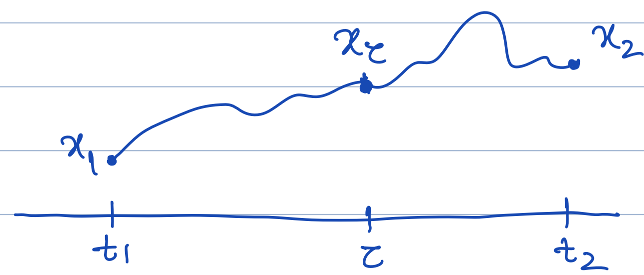
Let us now discuss an alternative approach to solve optimal control problems.

Dynamic Programming

Suppose we are solving an optimal control problem from time t_1 to t_2.

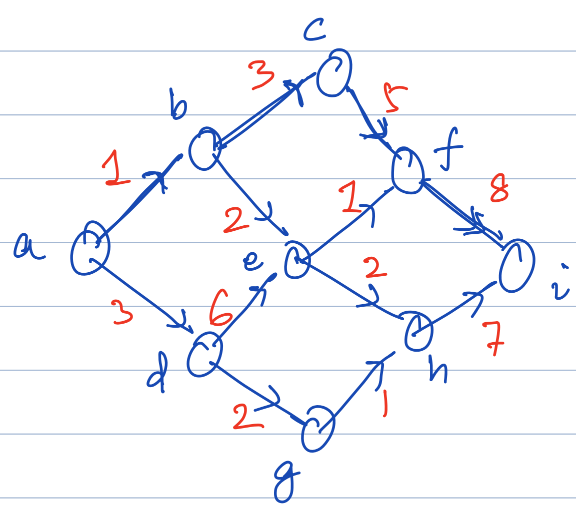
Let $x^*(t)$ be the optimal state trajectory.

Let $\tau \in [t_1, t_2]$, the state at τ is $x^*(\tau)$



If we now focus on the optimal control problem from τ to t_2, then its solution coincides with the solution of the original problem from τ to t_2.

Illustrative example: Find shortest path from a to i.



$17 = (a-b-c-f-i)$

$(a,b), (b,c), (c,f), (f,i)$

$(a-b-e-f-i)$

$18 = (a-d-e-f-i)$

$(a-d-e-h-i)$

$(a-b-e-h-i)$,

and so on

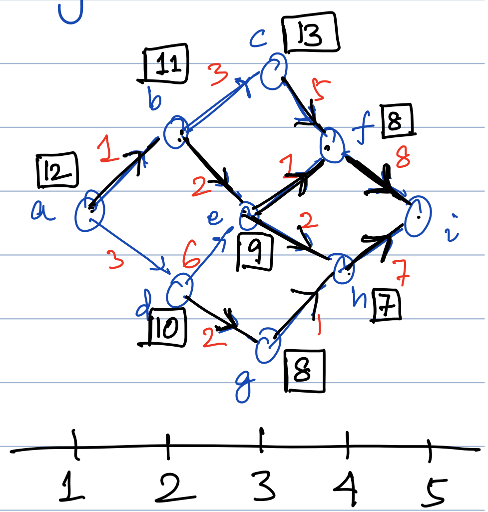
length of a path = sum of length of each edge/link included in that path

Brute force solution: i) find all paths from a to i

ii) compute length of all those paths

iii) choose the path with smallest length

Let us now compute shortest path using backward Induction.



Stage 4:

Shortest path from f to i : (f,i)

length = 8

For node h, length of shortest path 7

Stage 3: to Stage 4 :

For node c : only one path to nodes in stage 4.

length = 5 (length of (c,f)) + shortest path length from f onwards

= 5 + 8 = 13

For node e : if (e,f) is chosen, path length : 1+8 = 9

if (e,h) is chosen, " : 2+7 = 9

We have two optimal paths from (a) to (i), both with length 12.

a - b - e - f - i

a - b - e - h - i

Note: In the above example, number of stages : finite

number of states in every stage: finite

number of available action/input at every state: finite

In theory, we need a function $J_k^*(\bar{x})$ for every time k which gives us the optimal value of the problem from stage k to N with state at time k being $\bar{x}$.

Next Monday: 5pm - 6pm : Class Test

$f^k(0.1, 0.5) = 0.137 \rightarrow 0.1$

Now, consider the DT-OPT problem

$$\min_{\substack{u_{i:N-1}\\ x_{i:N}}} \quad \underbrace{\phi(N, x_N)} + \sum_{k=i}^{N-1} \underbrace{L^k(x_k, u_k)}$$

$$\text{s.t.} \quad x_{k+1} = f^k(x_k, u_k)$$

Start from N

define optimal value $J_N^*(\bar{x}) = \underline{\phi(N, \bar{x})}$.

at intermediate time k, we assume that $\underline{J_{k+1:N}^*(:)}$ is known.
with state $x_k = \bar{x}$

we solve $\quad \min_{u_k} \left[L^k(\bar{x}, u_k) + J_{k+1:N}^*(x_{k+1}) \right]$

where $x_{k+1} = f^k(\bar{x}, u_k)$

$$\underline{J_{k:N}^*(\bar{x})} = \min_{u_k} \left[L^k(\bar{x}, u_k) + J_{k+1:N}^*\left(f^k(\bar{x}, u_k)\right) \right]$$

$$\underline{u_k^*(\bar{x})} \in \operatorname*{argmin}_{u_k} \left[L^k(\bar{x}, u_k) + J_{k+1:N}^*\left(f^k(\bar{x}, u_k)\right) \right]$$

Let us apply this technique to DT LQR problem.

$$\min_{u_{i:N-1}, x_{i:N}} \quad \frac{1}{2} x_N^T S_N x_N + \frac{1}{2} \sum_{k=i}^{N-1} \left[x_k^T Q_k x_k + u_k^T R_k u_k \right]$$

$$\text{s.t.} \quad x_{k+1} = A_k x_k + B_k u_k, \quad k = i, i+1, \dots, N-1$$

x_i : given / known

at time N: $\quad \underline{J_N^*(\bar{x}) = \frac{1}{2} \bar{x}^T S_N \bar{x}}$, if $J_k^*(\bar{x}) = \frac{1}{2} \bar{x}^T S_k \bar{x}$, then

at time $k-1$:
with state $\bar{x}$: $\quad \min_{\bar{u}} \underline{\left[\frac{1}{2} \bar{x}^T Q_{k-1} \bar{x} + \frac{1}{2} \bar{u}^T R_{k-1} \bar{u} + J_k^*(A_{k-1}\bar{x} + B_{k-1}\bar{u}) \right]}$

$$= \min_{\bar{u}} \left[\frac{1}{2} \bar{x}^T Q_{K-1} \bar{x} + \frac{1}{2} \bar{u}^T R_{K-1} \bar{u} + \frac{1}{2} \left(A_{K-1}\bar{x} + B_{K-1}\bar{u} \right)^T S_K \left(A_{K-1}\bar{x} + B_{K-1}\bar{u} \right) \right]$$

$$= \min_{\bar{u}} \left[\frac{1}{2} \bar{u}^T \left[R_{K-1} + B_{K-1}^T S_k B_{K-1} \right] \bar{u} + (\bar{x})^T \left[A_{K-1}^T S_K B_{K-1} \right] \bar{u} + \frac{1}{2} \bar{x}^T \left[Q_{K-1} + A_{K-1}^T S_K A_{K-1} \right] \bar{x} \right]$$

$$= \min_{\bar{u}} g(\bar{u})$$

$$\nabla_u g(\bar{u}) = \left[R_{K-1} + B_{K-1}^T S_K B_{K-1} \right] \bar{u} + B_{K-1}^T S_K A_{K-1} \bar{x} = 0$$

$$\Rightarrow \bar{u} = - \left[R_{K-1} + B_{K-1}^T S_K B_{K-1} \right]^{-1} B_{K-1}^T S_K A_{K-1} \bar{x}$$

$$= - K_{K-1} \bar{x}$$

+ve def.
if $S_K \succeq 0$.

— which has the same expression we derived earlier.

It remains to show that $J_{K-1}^*(\bar{x}) = \frac{1}{2} \bar{x}^T S_{K-1} \bar{x}$ for some S_{K-1}.

we substitute $\bar{u} = -K_{K-1}\bar{x}$ in the cost function to obtain

$$\frac{1}{2} \bar{x}^T Q_{K-1} \bar{x} + \frac{1}{2} \bar{x}^T \left(K_{k-1}^T R_{k-1} K_{k-1} \right) \bar{x} + \frac{1}{2} \left(\left(A_{K-1} - B_{K-1} K_{K-1} \right) \bar{x} \right)^T S_k \left(A_{k-1} - B_{K-1} K_{k-1} \right) \bar{x}$$

$$= \frac{1}{2} \bar{x}^T S_{K-1} \bar{x}, \text{ where}$$

$$S_{K-1} = Q_{k-1} + K_{k-1}^T R_{k-1} K_{k-1} + \left(A_{k-1} - B_{k-1} K_{k-1} \right)^T S_k \left(A_{k-1} - B_{k-1} K_{k-1} \right).$$

If $S_N \succeq 0$, $Q_k \succeq 0 \; \forall k \Rightarrow S_{k-1} \succeq 0$ (symmetric, positive semi def)

Discrete-time Optimal tracking for LTI system

System model: $x_{k+1} = Ax_k + Bu_k$

$y_k = Cx_k$: output

given: reference output $(r_0, r_1, \dots r_N)$

and we wish to achieve $y_k \simeq r_k \;\; \forall k$.

$$\min_{\substack{u_{0:N-1}\\ x_{0:N}}} \; \underline{\frac{1}{2}(y_N - r_N)^T P (y_N - r_N)} + \frac{1}{2}\sum_{k=0}^{N-1}\left[\underline{(y_k - r_k)^T Q (y_k - r_k)} + \underline{u_k^T R u_k}\right] \quad \epsilon I$$

s.t. $x_{k+1} = Ax_k + Bu_k$

$y_k = Cx_k$

x_0 : given

if $y, r \in \mathbb{R}^2$

$$\begin{bmatrix} y_{1,k} - r_{1,k} \\ y_{2,k} - r_{2,k} \end{bmatrix}^T \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \begin{bmatrix} \cdot \\ \cdot \end{bmatrix}$$

Standard assumptions: $P \succeq 0,\; Q \succeq 0,\; R \succ 0$.

Let us apply dynamic programming to solve this.

at stage N:

$$J_N^*(\bar{x}) = \frac{1}{2}(C\bar{x} - r_N)^T P (C\bar{x} - r_N)$$

$$= \frac{1}{2}\bar{x}^T C^T P C \bar{x} + \frac{1}{2} r_N^T P r_N - \underline{r_N^T P C \bar{x}}$$

$$= \frac{1}{2}\bar{x}^T S_N \bar{x} - \underbrace{\bar{x}^T v_N} + q_N$$

$S_N = C^T P C$, $v_N = C^T P r_N$, $q_N = \frac{1}{2} r_N^T P r_N$

does not depend on $\bar{x}$

can we find S_{N-1}, v_{N-1}, q_{N-1} s.t.

$$\underline{J^*_{N-1}(\bar{x}) = \frac{1}{2}\bar{x}^T S_{N-1}\bar{x} - \bar{x}^T v_{N-1} + q_{N-1}} \quad \text{——} \quad (a)$$

From dynamic programming, we know if state at time N-1 is $\bar{x}$

$$J^*_{N-1}(\bar{x}) = \min_{\bar{u}}\left[\frac{1}{2}(C\bar{x}-r_{N-1})^T Q(C\bar{x}-r_{N-1}) + \frac{1}{2}\bar{u}^T R\bar{u} + J^*_N(A\bar{x}+B\bar{u})\right] =: \min_{\bar{u}} g(\bar{u})$$

$$g(\bar{u}) = \frac{1}{2}\bar{x}^T C^T Q C\bar{x} + \frac{1}{2} r_{N-1}^T Q r_{N-1} - r_{N-1}^T Q C\bar{x}$$
$$+ \frac{1}{2}\bar{u}^T R\bar{u} + \frac{1}{2}(A\bar{x}+B\bar{u})^T S_N(A\bar{x}+B\bar{u}) - (A\bar{x}+B\bar{u})^T v_N + q_N$$

$$= (\text{terms that do not depend on } u)$$
$$+ \frac{1}{2}\bar{u}^T[R+B^T S_N B]\bar{u} + (\bar{u})^T B^T S_N A\bar{x} - (\bar{u})^T B^T v_N$$

$$\nabla_u g(\bar{u}) = [R+B^T S_N B]\bar{u} + [B^T S_N A\bar{x} - B^T v_N] = 0$$

$$\Rightarrow \bar{u}^* = -[R+B^T S_N B]^{-1} B^T S_N A\bar{x}$$
$$+ [R+B^T S_N B]^{-1} B^T v_N$$
$$=: -K_{N-1}\bar{x} + K^v_{N-1} v_N$$

It remains to show that substituting $\bar{u}^*$ in the cost function yields $J^*_{N-1}(\bar{x})$ as given in (a).

$$\bar{u}^* = -K_{N-1}\bar{x} + K^v_{N-1} v_N$$

LECTURES 18 & 19 : 10th Feb

Let us substitute $\bar{u}^*$ in $g(\bar{u})$, which yields

$$g(\bar{u}) = \frac{1}{2}\bar{x}^T C^T Q C \bar{x} + \frac{1}{2} r_{N-1}^T Q r_{N-1} - r_{N-1}^T Q C \bar{x} + q_N$$

$$+ \frac{1}{2}\bar{u}^T \left[R + B^T S_N B\right]\bar{u} + (\bar{u})^T B^T S_N A \bar{x} - (\bar{u})^T B^T v_N$$

$$+ \frac{1}{2}\bar{x}^T A^T S_N A \bar{x} - (\bar{x})^T A v_N$$

$$= \frac{1}{2}\bar{x}^T \left[C^T Q C + A^T S_N A\right]\bar{x} + \frac{1}{2} r_{N-1}^T Q r_{N-1} + q_N$$

$$- (\bar{x})^T \left[A v_N + C^T Q r_{N-1}\right]$$

$$+ \frac{1}{2}\left(-K_{N-1}\bar{x} + K_{N-1}^v v_N\right)^T \left(R + B^T S_N B\right)\left(-K_{N-1}\bar{x} + K_{N-1}^v v_N\right)$$

$$+ \left(-K_{N-1}\bar{x} + K_{N-1}^v v_N\right)^T \left[B^T S_N A \bar{x} - B^T v_N\right]$$

needs to be simplified to

$$= \frac{1}{2}\bar{x}^T S_{N-1}\bar{x} + q_{N-1} - \bar{x}^T v_{N-1}$$

$$\cancel{(R + B^T S_N B)^{-1}} B^T S_N A$$

$$S_{N-1} = C^T Q C + A^T S_N A + K_{N-1}^T \cancel{(R + B^T S_N B)} \underline{\underline{K_{N-1}}}$$

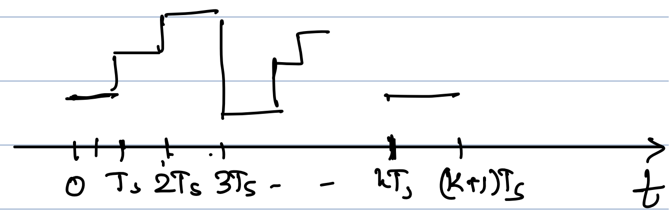
$$- 2K_{N-1}^T B^T S_N A$$

$$= C^T Q C + A^T S_N A - K_{N-1}^T B^T S_N A$$

$$= C^T Q C + (A - BK_{N-1})^T S_N A$$

Discretizing a continuous-time System

Consider a continuous-time system $\dot{x} = f(x,u,t)$



Discretization is the process of approximating the above system trajectory by the trajectory of a discrete-time system.

Let T_s be the sampling time.

Zero-order hold: $u(t) = u_k$ if $t \in [kT_s, (k+1)T_s)$

discrete-time system: $x^d_{k+1} = f^d(x^d_k, u_k, k)$

The goal is to determine f^d to achieve $x^d_k \cong x(kT_s)$

For LTI systems, we can find f^d to be a DT linear system.

$\dot{x} = Ax + Bu$, $\qquad x_{k+1} = A^d x_k + B^d u_k$

$\downarrow$ $u(t) = u_k$, $t \in [kT_s, (k+1)T_s)$.

$$x(t) = e^{At} x(0) + \int_0^t e^{A(t-\tau)} Bu(\tau)\, d\tau$$

$$x(t) = e^{A(t-kT_s)} x(kT_s) + \left[\int_{kT_s}^{t} e^{A(t-\tau)} B\, d\tau\right] u_k$$

$$x^d_{k+1} = x((k+1)T_s)$$

$$= e^{AT_s} x^d_k + \left(\int_{kT_s}^{(k+1)T_s} e^{A(t-\tau)} B\, d\tau\right) u_k \rightarrow B^d$$

$A^d \leftarrow$

$$= e^{AT_s} x^d_k + \left(\int_0^{T_s} e^{A(t-\tau)} B\, d\tau\right) u_k$$

Clarification on duality

$$\min_{x} \quad -x_1^2 - x_2^2$$
$$\text{s.t.} \quad x_1^2 + x_2^2 - 1 \leq 0 \;:\; \lambda \in \mathbb{R}$$

$$(1+\lambda+\mu)x_1^2 + 2\mu x_2^2 - 5\lambda + 6\mu$$

$$L(x,\lambda) = -x_1^2 - x_2^2 + \lambda[x_1^2 + x_2^2 - 1]$$
$$= [\lambda - 1][x_1^2 + x_2^2] - \lambda =$$

$$d(\lambda) = \begin{cases} -\lambda & \text{for } \lambda \geq 1 \\ -\infty & \text{for } \lambda < 1 \end{cases}$$

dual optimization $\max_{\lambda \geq 0} d(\lambda) = \max_{\lambda \geq 1} -\lambda = -1$

$$L(x,\lambda,\mu) = (1+\lambda+\mu)x_1^2 + 2\mu x_2^2 - 5\lambda + 6\mu$$

$$d(\lambda) = \begin{cases} -5\lambda + 6\mu & \text{if } 1+\lambda+\mu \geq 0, \; \mu \geq 0 \\ -\infty & \text{otherwise} \end{cases}$$

dual opt:

$$\left| \begin{array}{l} \max_{\lambda,\mu} \quad -5\lambda + 6\mu \\ \text{s.t.} \quad \lambda \geq 0, \; \mu \geq 0, \; 1+\lambda+\mu \geq 0 \end{array} \right|$$

Programming demo: on Laptop