

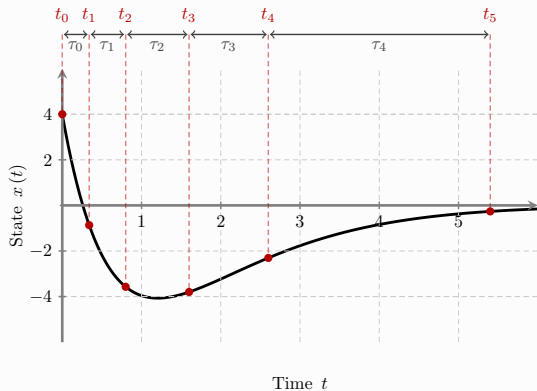
# Optimal Control

*Martina Maggio*

Saarland University  
Department of Computer Science

October 30, 2025

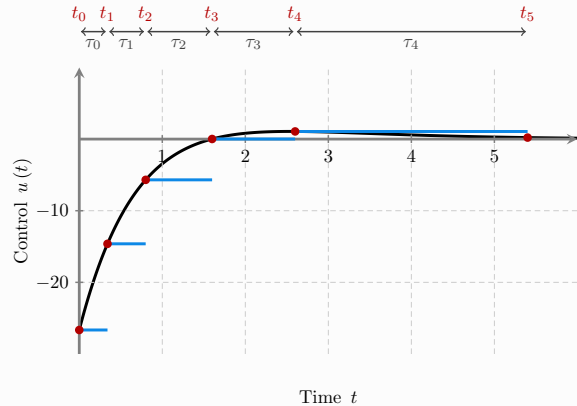
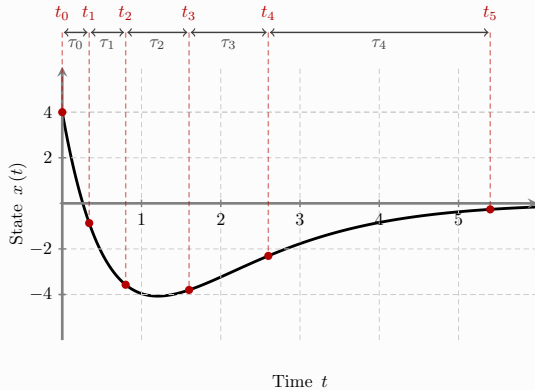
# SAMPLING



## Sampling

- ▶  $\{t_0, t_1, \dots, t_N\}$ : Sampling pattern
- ▶  $t_k$ : Sampling instants
- ▶  $\tau_k = t_{k+1} - t_k$ : Interarrival times

# SAMPLING



# QUESTIONS

## Question

left plot jumps with \only. how to fix?



## THE PROBLEM

Given

$$\begin{cases} \dot{x}(t) &= A x(t) + B u(t) \\ x(0) &= x_0 \end{cases}$$

- ▶ We restrict  $u(t)$  to be in the class of piecewise constant functions and denote it as  $\bar{u}(t)$
- ▶ Finite horizon  $[0, T]$
- ▶ Number  $N$  of control updates

find

- ▶ Sampling instants  $0 = t_1 < \dots < t_N = T$
- ▶ Piecewise constant  $\bar{u}(t) = u_k, \quad \forall t \in [t_{k-1}, t_k)$

that minimizes the performance index

$$\text{▶ } \mathcal{J}(\bar{u}) = \int_0^T (x'(t) Q x(t) + \bar{u}'(t) R \bar{u}(t)) dt + x'(T) S x(T)$$



# COMMENTS

Comment

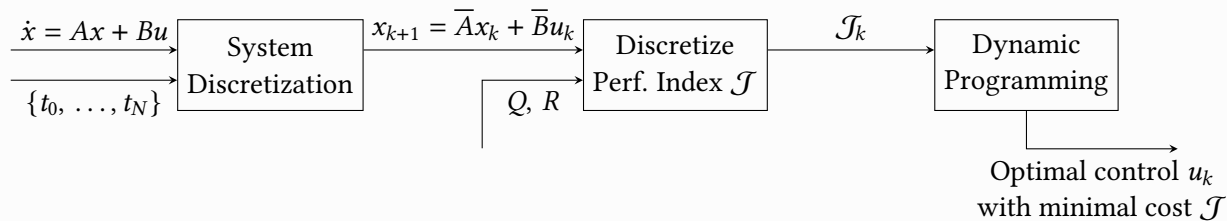
"find" looks weird?

## QUADRATIC PERFORMANCE INDEX

$$\mathcal{J}(\bar{u}) = \int_0^T \left( x'(t) \underbrace{Q}_{\succeq} x(t) + \bar{u}'(t) \underbrace{R}_{\succ} \bar{u}'(t) \right) dt + x'(T) \underbrace{S}_{\succeq} x(T)$$
$$\text{s.t. } \begin{cases} \dot{x}(t) &= A x(t) + B u(t) \\ x(0) &= x_0 \end{cases}$$

- ▶  $x(t) \in \mathbb{R}^n$ : system state
- ▶  $u(t) \in \mathbb{R}^m$ : system input
- ▶  $A, B, Q, R, S$ : constant matrices

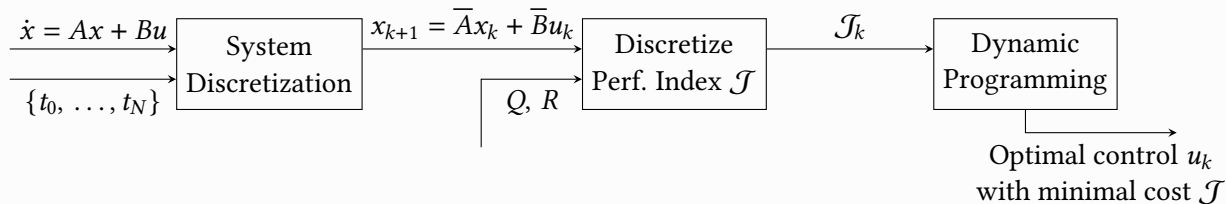
## PICTURE





# OUTLINE

- ▶ Computation of optimal control  $u(t)$  for continuous-time systems
- ▶ Discretization Process for given  $\{t_0, \dots, t_N\}$ 
  1. System Discretization
  2. Computation of optimal control  $(u_k)_{k \in \{0, \dots, N\}}$  for the discrete-time system



# OUTLINE

- ▶ Computation of optimal control  $u(t)$  for continuous-time systems
- ▶ Discretization Process for given  $\{t_0, \dots, t_N\}$ 
  1. System Discretization
  2. Computation of optimal control  $(u_k)_{k \in \{0, \dots, N\}}$  for the discrete-time system
- ▶ Sampling Density and Sampling Method Cost
- ▶ Sampling Methods for finding  $\{t_0, \dots, t_N\}$ 
  1. Periodic sampling
  2. Lebesgue sampling
  3. Quantization-based sampling
- ▶ Results

## OPTIMAL CONTROL: CONTINUOUS-TIME SYSTEMS

$$\text{minimize}_u \int_0^T \left( x'(t) Q x(t) + u'(t) R u(t) \right) dt + x(T)' S x(T)$$

Riccati equation for continuous-time systems

$$\begin{cases} \dot{K}(t) &= K(t) B R^{-1} B' K(t) - A' K(t) - K(t) A - Q, \\ K(T) &= S \end{cases}$$

which gives us the optimal control  $u(t) = -R^{-1} B' K(t) x(t)$

with achieved cost

$$\mathcal{J}_\infty = x_0' K(0) x_0$$



## OPTIMAL CONTROL: DISCRETE-TIME SYSTEMS

Given the time points  $\{t_0, \dots, t_N\}$ , with interarrivals  $\tau_k, k \in \{0, \dots, N\}$ , the discrete-time state space equations are:

$$\begin{cases} x_{k+1} &= \bar{A}_k x_k + \bar{B}_k \\ x(0) &= x_0 \end{cases}$$

$$\bar{A}_k = \Phi(\tau_k)$$

$$\Phi(\tau) = e^{A\tau}$$

State Matrix

$$\bar{B}_k = \Gamma(\tau_k)$$

$$\Gamma(\tau) = \int_0^\tau e^{A(\tau-t)} dt B$$

Input Matrix

## PERFORMANCE INDEX FOR DISCRETE-TIME SYSTEMS

$$\begin{aligned}
 & \int_0^T (x'(t) Q x(t) + \bar{u}'(t) R \bar{u}(t)) dt + x'(T) S x(T) \\
 &= \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} (x'(t) Q x(t) + \bar{u}'(t) R \bar{u}(t)) dt + x'(T) S x(T) \\
 &= \sum_{k=0}^{N-1} \left[ \int_{t_k}^{t_{k+1}} (\Phi x_k + \Gamma u_k)' Q (\Phi x_k + \Gamma u_k) dt + \int_{t_k}^{t_{k+1}} u_k' R u_k \right] + x'(T) S x(T) \\
 &= x_k' \underbrace{\left( \int_{t_k}^{t_{k+1}} \Phi' Q \Phi dt \right)}_{\bar{Q}} x_k + \underbrace{u_k' \left( \int_{t_k}^{t_{k+1}} \Gamma' Q \Gamma dt + \underbrace{\int_{t_k}^{t_{k+1}} R dt}_{\tau_k R} \right)}_{\bar{R}} u_k + 2x_k' \underbrace{\left( \int_{t_k}^{t_{k+1}} \Phi' Q \Gamma dt \right)}_{\bar{P}} u_k + x'(T) S x(T)
 \end{aligned}$$

## PERFORMANCE INDEX FOR DISCRETE-TIME SYSTEMS

$$\begin{aligned} & \int_0^T (x'(t) Q x(t) + \bar{u}'(t) R \bar{u}(t)) dt + x'(T) S x(T) \\ &= x'_k \underbrace{\left( \int_{t_k}^{t_{k+1}} \Phi' Q \Phi dt \right)}_{\bar{Q}} x_k + u'_k \underbrace{\left( \int_{t_k}^{t_{k+1}} \Gamma' Q \Gamma dt + \underbrace{\int_{t_k}^{t_{k+1}} R dt}_{\tau_k R} \right)}_{\bar{R}} u_k + 2x'_k \underbrace{\left( \int_{t_k}^{t_{k+1}} \Phi' Q \Gamma dt \right)}_{\bar{P}} u_k + x'(T) S x(T) \end{aligned}$$

$$\mathcal{J}(\bar{u}) = \sum_{k=0}^{N-1} \left( x'_k \bar{Q} x_k + u'_k \bar{R} u_k + 2x'_k \bar{P} u_k \right) + x'_N S x_N$$

# COMMENTS

## Comment

- ▶ Different brackets for integrals due to underbrace.
- ▶  $\Gamma$  and  $\Phi$  without args? Double check!

# DYNAMIC PROGRAMMING: BELLMAN EQUATION

## Bellman Equation

For  $k \in \{0, \dots, N\}$ , we define the function  $\mathcal{J}_k : \mathbb{R}^n \rightarrow \mathbb{R}$ , which gives the minimal cost achievable from stage  $k$  onward, given the state  $x_k$  as

$$\begin{cases} \mathcal{J}_k(x_k) = \min_u \left[ \underbrace{x_k' \bar{Q}_k x_k}_{\text{State penalty}} + \underbrace{u' \bar{R}_k u}_{\text{Control penalty}} + \underbrace{2x_k' \bar{P}_k u}_{\text{Cross term}} + \mathcal{J}_{k+1}(x_{k+1}) \right] & \text{for } k \in \{0, \dots, N-1\} \\ \mathcal{J}_N(x) = x' S x & . \end{cases}$$

**Intuition:** define a *backward recursive function* that gives the minimal achievable cost from the current state onward.





# DYNAMIC PROGRAMMING: BELLMAN EQUATION

## Bellman Equation

For  $k \in \{0, \dots, N\}$ , we define the function  $\mathcal{J}_k : \mathbb{R}^n \rightarrow \mathbb{R}$ , which gives the minimal cost achievable from stage  $k$  onward, given the state  $x_k$  as

$$\begin{cases} \mathcal{J}_k(x_k) = \min_u \left[ \underbrace{x_k' \bar{Q}_k x_k}_{\text{State penalty}} + \underbrace{u' \bar{R}_k u}_{\text{Control penalty}} + \underbrace{2x_k' \bar{P}_k u}_{\text{Cross term}} + \mathcal{J}_{k+1}(x_{k+1}) \right] & \text{for } k \in \{0, \dots, N-1\} \\ \mathcal{J}_N(x) = x' S x & . \end{cases}$$

$$x_{k+1} = \bar{A}_k x_k + \bar{B}_k u_k$$

# QUESTIONS

## Question

same issue: the first block moves

## DYNAMIC PROGRAMMING: QUADRATIC FORM

$\mathcal{J}_k(x)$  is a quadratic form of  $x$

$$\mathcal{J}_k(x) = x' \bar{K}_k x$$

*Unknown at this point!*



$$\mathcal{J}_{k+1}(\bar{A}_k x + \bar{B}_k u) = (\bar{A}_k x + \bar{B}_k u)' \bar{K}_{k+1} (\bar{A}_k x + \bar{B}_k u)$$



Plug into **Bellman equation**

$$\mathcal{J}_k(x) = x' \underbrace{(\bar{Q}_k + \bar{A}_k' \bar{K}_{k+1} \bar{A}_k)}_{\hat{Q}_k} x + u' \underbrace{(R_k + \bar{B}_k' \bar{K}_{k+1} \bar{B}_k)}_{\hat{R}_k} u + 2x' \underbrace{(\bar{P}_k + \bar{A}_k' \bar{K}_{k+1} \bar{B}_k)}_{\hat{S}_k} u$$

## DYNAMIC PROGRAMMING: QUADRATIC FORM

$\mathcal{J}_k(x)$  is a quadratic form of  $x$



$$\mathcal{J}_k(x) = x' \bar{K}_k x$$

*Unknown at this point!*



$$\mathcal{J}_{k+1}(\bar{A}_k x + \bar{B}_k u) = (\bar{A}_k x + \bar{B}_k u)' \bar{K}_{k+1} (\bar{A}_k x + \bar{B}_k u)$$



Plug into **Bellman equation**

$$\mathcal{J}_k(x) = x' \hat{Q}_k x + u' \hat{R}_k u + 2x' \hat{S}_k u$$

# DYNAMIC PROGRAMMING: MINIMIZATION

We aim to **minimize** the cost:

$$\min_u \left( x' \hat{Q}_k x + u' \hat{R}_k u + 2x' \hat{S}_k u \right)$$

**Optimality** condition:

$$\frac{\partial}{\partial u}(\cdot) = 0 \quad \implies \quad \bar{R}_k u + \bar{S}_k' x = 0$$

$\Downarrow$  Solving for  $u$

$$u_k^* = -\hat{R}_k^{-1} \hat{S}_k' x_k$$



## DYNAMIC PROGRAMMING: RICCATI SOLUTION

We plug  $u_k^*$  into the **Bellman equation**:

$$\mathcal{J}_k(x) = x' \left( \hat{Q}_k - \hat{B}_k \hat{R}_k^{-1} \hat{B}_k' \right) x$$



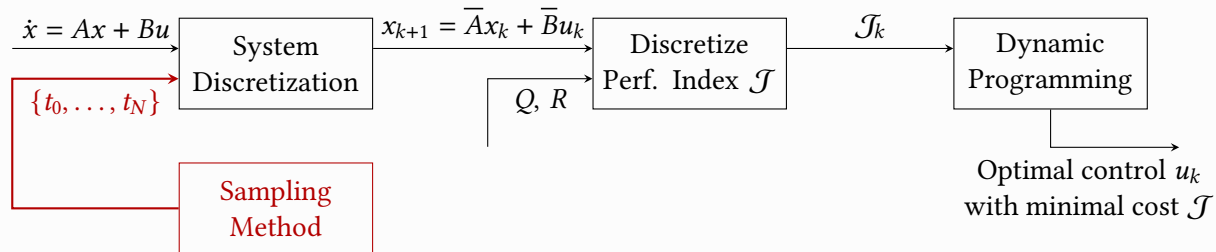
Recall  $\mathcal{J}_k(x) = x' \bar{K}_k x$

$$\bar{K}_k = \hat{Q}_k - \hat{S}_k \hat{R}_k^{-1} \hat{S}_k'$$

Minimal cost equals

$$\mathcal{J} = x_0' \bar{K}_0 x_0$$

## PICTURE



# COMPARING SAMPLING METHODS: SAMPLING DENSITY

## Sampling Density

Given  $x_0$ ,  $A$ ,  $B$ ,  $Q$ ,  $R$ , and  $S$ , and interval length  $T$ , and a number of samples  $N$ , the *sampling density*  $\sigma_{N,m} : [0, T] \rightarrow \mathbb{R}^+$  of any sampling method  $m$  is defined as

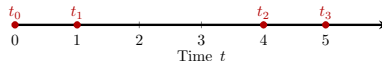
$$\sigma_{N,m}(t) = \frac{1}{N \tau_k} \quad \forall t \in [t_k, t_{k+1}), \quad k \in \{0, \dots, N-1\}$$

- ▶ Temporal distribution of sampling instants
- ▶ Sampling density is normalized

## Example

Consider  $T = 5$  units,  $N = 4$  samples, and the following sampling instants

- ▶  $\sigma_4(t) = \frac{1}{4 \cdot 1} = \frac{1}{4}, \quad \forall t \in [0, 1)$
- ▶  $\sigma_4(t) = \frac{1}{4 \cdot 3} = \frac{1}{12}, \quad \forall t \in [1, 4)$





# COMPARING SAMPLING METHODS: SAMPLING DENSITY

## Sampling Density

Given  $x_0$ ,  $A$ ,  $B$ ,  $Q$ ,  $R$ , and  $S$ , and interval length  $T$ , and a number of samples  $N$ , the *sampling density*  $\sigma_{N,m} : [0, T] \rightarrow \mathbb{R}^+$  of any sampling method  $m$  is defined as

$$\sigma_{N,m}(t) = \frac{1}{N \tau_k} \quad \forall t \in [t_k, t_{k+1}), \quad k \in \{0, \dots, N-1\}$$

- ▶ Temporal distribution of sampling instants
- ▶ Sampling density is normalized

## Asymptotic Sampling Density

To remove the dependency on  $N$ , we define the *asymptotic sampling density* as  $\sigma_m : [0, T] \rightarrow \mathbb{R}^+$  as the limit

$$\sigma_m(t) = \lim_{N \rightarrow \infty} \sigma_{N,m}(t)$$

# COMPARING SAMPLING METHODS: SAMPLING DENSITY

## Sampling Density

Given  $x_0$ ,  $A$ ,  $B$ ,  $Q$ ,  $R$ , and  $S$ , and interval length  $T$ , and a number of samples  $N$ , the *sampling density*  $\sigma_{N,m} : [0, T] \rightarrow \mathbb{R}^+$  of any sampling method  $m$  is defined as

$$\sigma_{N,m}(t) = \frac{1}{N \tau_k} \quad \forall t \in [t_k, t_{k+1}), \quad k \in \{0, \dots, N-1\}$$

- ▶ Temporal distribution of sampling instants
- ▶ Sampling density is normalized

## Asymptotic Sampling Density

To remove the dependency on  $N$ , we define the *asymptotic sampling density* as  $\sigma_m : [0, T] \rightarrow \mathbb{R}^+$  as the limit

$$\sigma_m(t) = \lim_{N \rightarrow \infty} \sigma_{N,m}(t)$$

# COMPARING SAMPLING METHODS: NORMALIZED COST

## Normalized Cost

---

Given  $x_0$ ,  $A$ ,  $B$ ,  $Q$ ,  $R$ , and  $S$ , and interval length  $T$ , and a number of samples  $N$ , the *normalized cost* of any sampling method  $m$  is defined as

$$c_{N,m} = \frac{N^2}{T^2} \frac{\mathcal{J}_{N,m} - \mathcal{J}_\infty}{\mathcal{J}_\infty}$$

where  $\mathcal{J}_{N,m}$  is the minimal cost of the sampling method  $m$  with  $N$  samples, and  $\mathcal{J}_\infty$  is the minimal cost of the continuous-time system.

## Example

---

TODO

# COMPARING SAMPLING METHODS: NORMALIZED COST

## Normalized Cost

Given  $x_0$ ,  $A$ ,  $B$ ,  $Q$ ,  $R$ , and  $S$ , and interval length  $T$ , and a number of samples  $N$ , the *normalized cost* of any sampling method  $m$  is defined as

$$c_{N,m} = \frac{N^2}{T^2} \frac{\mathcal{J}_{N,m} - \mathcal{J}_\infty}{\mathcal{J}_\infty}$$

where  $\mathcal{J}_{N,m}$  is the minimal cost of the sampling method  $m$  with  $N$  samples, and  $\mathcal{J}_\infty$  is the minimal cost of the continuous-time system.

## Asymptotic Normalized Cost

To remove the dependency on  $N$ , we define the *asymptotic normalized cost* as the limit

$$c_m = \lim_{N \rightarrow \infty} c_{N,m}$$

## PERIODIC SAMPLING

We divide the interval  $[0, T]$  into  $N$  parts of equal size

$$\tau_k = \tau = \frac{T}{N}, \quad k \in \{0, \dots, N-1\}$$

$$t_k = k \cdot \frac{T}{N}, \quad k \in \{0, \dots, N\}$$

For  $N \in \mathbb{N}$ , we get the constant sampling density

$$\sigma_{per,N}(t) = \frac{1}{N \cdot t_k} = \frac{1}{N} \cdot \frac{N}{T} = \frac{1}{T}$$

## PERIODIC SAMPLING: OPTIMAL CONTROL

For sampling period  $\tau$ , the solution  $\bar{K}(\tau)$  of the discrete-time Riccati equation can be determined analytically as

$$\bar{K}(\tau) = K_{\infty} + X \cdot \frac{\tau^2}{2} + o(\tau^2)$$

where  $K_{\infty}$  is the solution of the continuous-time Riccati equation, and  $X$  is the solution of Lyapunov equation<sup>1</sup>

*... informally, optimal controller of the discrete-time can be expressed as the continuous-time solution  $K_{\infty}$  plus a correction term that is proportional to the square of the sampling period  $\tau$ .*

<sup>1</sup> Melzer, Stuart M., and Benjamin C. Kuo. "Sampling period sensitivity of the optimal sampled data linear regulator." Automatica 7.3 (1971): 367-370.

# QUESTIONS

Question

footnote not visible!

## PERIODIC SAMPLING: ASYMPTOTIC NORMALIZED COST

$$c_{N,\text{per}} = \frac{N^2}{T^2} \frac{\mathcal{J}_{N,\text{per}} - \mathcal{J}_\infty}{\mathcal{J}_\infty}$$

$$\bar{K}(\tau) = K_\infty + X \cdot \frac{\tau^2}{2} + o(\tau^2)$$

$$\begin{aligned} c_{\text{per}} &= \lim_{N \rightarrow \infty} \frac{N^2}{T^2} \frac{x_0' \bar{K}(\tau) x_0 - x_0' K_\infty x_0}{x_0' K_\infty x_0} \\ &= \lim_{N \rightarrow \infty} \frac{N^2}{T^2} \frac{x_0' \left( K_\infty + X \cdot \frac{\tau^2}{2} + o(\tau^2) \right) x_0 - x_0' K_\infty x_0}{x_0' K_\infty x_0} \\ &= \lim_{N \rightarrow \infty} \frac{N^2}{T^2} \frac{x_0' \left( K_\infty + X \cdot \frac{T^2}{2N^2} + o\left(\frac{T^2}{2N^2}\right) \right) x_0 - x_0' K_\infty x_0}{x_0' K_\infty x_0} \\ &= \lim_{N \rightarrow \infty} \frac{N^2}{T^2} \frac{x_0' \frac{XT^2}{2N^2} x_0 + x_0' o(N^{-2}) x_0}{x_0' K_\infty x_0} \\ &= \frac{x_0' X x_0}{2 x_0' K_\infty x_0} \end{aligned}$$



## PERIODIC SAMPLING: EXAMPLE

$$c_{\text{per}} = \frac{x_0' X x_0}{2 x_0' K_{\infty} x_0}$$

### Example

For a first-order system ( $n = 1$ ), wlog  $B = R = 1$ , we obtain

$$X = \frac{1}{12}(K_{\infty} - A)K_{\infty}^2,$$
$$K_{\infty} = A + \sqrt{A^2 + Q}.$$

which gives us the asymptotic normalized cost

$$c_{\text{per}} = \frac{1}{24}A\sqrt{A^2 + Q} + A^2 + Q.$$

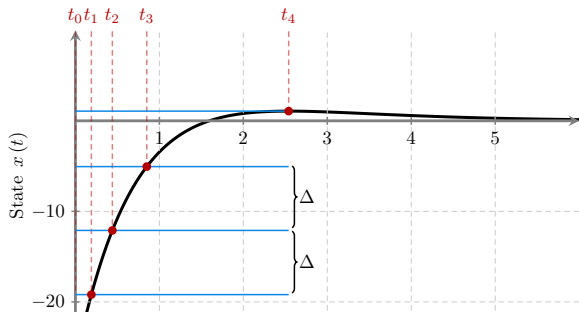
## DETERMINISTIC LEBESGUE SAMPLING

- **Intuition:** Sample more frequently where the control changes faster
- Sample whenever the optimal  $u$  changes by a fixed threshold  $\Delta$ , so after any sampling instant  $t_k$ , the next  $t_{k+1}$  is determined s.t.

$$\| u(t_{k+1}) - u(t_k) \| = \Delta$$

where  $u$  is the optimal continuous-time input

### Example ( $\Delta = 7$ )



## DETERMINISTIC LEBESGUE SAMPLING

$$\| u(t_{t+1}) - u(t_k) \| = \Delta$$

In the case of a scalar input ( $m = 1$ ) and a given number  $N$  of sampling instance in  $[0, T]$ , we compute the sampling instants  $t_k$

$$\begin{aligned} \int_{t_k}^{t_{k+1}} |\dot{u}(t)| dt &= |u(t_{k+1}) - u(t_k)| = \Delta \\ &= \frac{1}{N} \cdot \underbrace{\int_0^T |\dot{u}(t)| dt}_{N \cdot \Delta} \end{aligned}$$

## DETERMINISTIC LEBESGUE SAMPLING: SAMPLING DENSITY

$$\int_{t_k}^{t_{k+1}} |\dot{u}(t)| dt = \frac{1}{N} \cdot \int_0^T |\dot{u}(t)| dt$$

$$\begin{aligned}\sigma_{\text{dls}}(t) &= \frac{1}{N \cdot \tau_k} \\ &= \frac{\int_{t_k}^{t_{k+1}} |\dot{u}(t)| dt}{\int_0^T |\dot{u}(t)| dt} \cdot \frac{1}{\tau_k} \\ &= \frac{\int_{t_k}^{t_{k+1}} |\dot{u}(t)| dt}{\tau_k} \cdot \frac{1}{\int_0^T |\dot{u}(t)| dt} \\ &= \frac{|u(t_{k+1}) - u(t_k)|}{t_{k+1} - t_k} \cdot \frac{1}{\int_0^T |\dot{u}(t)| dt}\end{aligned}$$

- Sampling intervals vary depending on  $|\dot{u}(t)|$

# QUESTIONS

## Question

with Eqarray doesn't look good here, why?