Lecture 4. Optimal Control

Deterministic Continuous Finite Time

Controller Design Problem (Finite Horizon)

- Given a state space **X**, a control space **U**, an observation space **Y**, an initial state, noise models in stochastic case
- We need to design a control law that minimizes the objective function

	Deterministic Model	**Stochastic Model**
Continuous time	$\boxed{\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x,u) \\ J &= \int_{t=0}^{T} \delta^t l_t(x,u,\dot{x})dt \end{aligned}}$	$\begin{aligned} \dot{x} &= f(x,u,w) \\ y &= h(x,u,v) \\ J &= \int_{t=0}^{T} \delta^t l_t(x,u,\dot{x})dt \end{aligned}$
Discrete time	$\begin{aligned} x_{k+1} &= f(x_k,u_k) \\ y_k &= h(x_k,u_k) \\ J &= \sum_{k=0}^{N} \delta^k l_k(x_k,u_k,x_{k+1}) \end{aligned}$	$\begin{aligned} x_{k+1} &= f(x_k,u_k,w_k) \\ y_k &= h(x_k,u_k,v_k) \\ J &= \sum_{k=0}^{N} \delta^k l_k(x_k,u_k,x_{k+1}) \end{aligned}$

Continuous Time Optimal Control

- Problem formulation and examples
- Dynamic programming
- Maximum principle
- Linear quadratic regulator

Problem Formulation

- Deterministic continuous time
- State feedback
- Finite horizon
- No discount
- Controllable System

$$\min_{u_t, t\in[0,T)} J = \int_{t=0}^{T} l_t(x_t, u_t)dt + l_T(x_T)$$

$$\dot{x} = f(x, u)$$

Problem Formulation

- Decision variables: the control trajectory $u_t, t \in [0, T)$
- Known parameters:
  - Initial state: x_0
  - Run-time costs: l_t
  - Terminal cost: l_T

$$\min_{u_t, t\in[0,T)} J = \int_{t=0}^{T} l_t(x_t, u_t)dt + l_T(x_T)$$

$$\dot{x} = f(x, u)$$

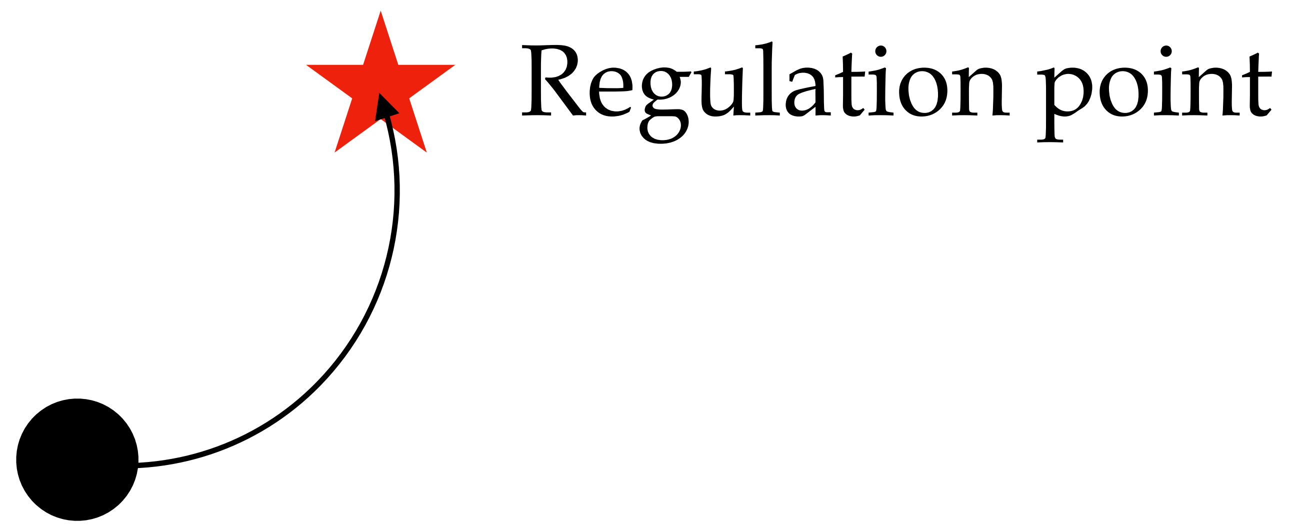
Solutions

- Open-loop solution:
  - A trajectory of future controls $u_t, t \in [0, T)$
- Closed-loop solution:
  - A control policy $u_t = q_t(x_t), \forall t$
  - More robust to disturbances

Example 1: Regulation Problem

- Dynamic model (double integrator)

$$\dot{x} = Ax + Bu$$



- Cost (quadratic)

$$J = \int_{t=0}^{T} [x_t^T Q_t x_t + u_t^T R_t u_t] dt + x_T^T S_T x_T$$

- This is a linear quadratic regulation problem

Continuous Time vs. Discrete Time

- Physical systems can be more precisely described using continuous time ODE models
  - Examples: ground vehicles, aerial vehicles, robots
- Continuous time models can be viewed as an extreme case of discrete time models where we set the sampling time to infinitely small.

Continuous Time vs. Discrete Time

- Many observation and control systems are implemented in discrete time. In those systems, designers usually start from discrete time models.
- Implementation media: digital signal processor, field-programmable gate array (FPGA), etc.

Continuous Time Optimal Control

- Problem formulation and examples
- Dynamic programming
- Maximum principle
- Linear quadratic regulator

Optimal Solution

$$\min_{u_t, t\in[0,T)} J = \int_{t=0}^{T} l_t(x_t, u_t)dt + l_T(x_T) \quad \longrightarrow \quad J_0(x_0; u_{0:T})$$

$$\dot{x} = f(x, u)$$

***We will not explicit write this equality constraint if it is clear from the context.**

- Let $u^*_{0:T}$ be an optimal control trajectory
- Together with x_0, the optimal control sequence determines the corresponding state sequence $x^*_{0:T}$
- The optimal cost corresponding to the optimal sequence is denoted $J^*(x_0)$

Principle of Optimality

- Consider a subproblem: we start at x_t^* at time t and wish to minimize the "cost-to-go" from time t to time T over $u_{t:T}$,

$$\min_{u_{t:T}} \int_{\tau=t}^{T} l_\tau(x_\tau, u_\tau) d\tau + l_T(x_T)$$

- Then the truncated optimal control sequence $u_{t:T}^*$ is optimal for this subproblem.

- We can also use proof by contradiction as discussed in the discrete time case.

Utilizing Principle of Optimality

$$J_t(x_t; u_{t:T}) = \int_t^{t'} l_\tau(x_\tau, u_\tau) d\tau + J_{t'}(x_{t'}; u_{t':T})$$

Take minimization on both sides

$$\min_{u_{t:T}} J_t(x_t; u_{t:T}) = \min_{u_{t:t'}} \left\{ \int_t^{t'} l_\tau(x_\tau, u_\tau) d\tau + \min_{u_{t':T}} J_{t'}(\boxed{x_{t'}}; u_{t':T}) \right\}$$

This term depends on $u_{t:t'}$

These are not two independent minimization problems

But they can be solved sequentially

Cost-to-Go and Value Function

Cost-to-go at time t: a function that depends on the current state and all future control inputs

$$J_t(x_t; u_{t:T}) = \int_{\tau=t}^{T} l_\tau(x_\tau, u_\tau) d\tau + l_T(x_T)$$

Value function (optimal cost-to-go): a time-varying function that depends on the current state (assuming optimal actions will be taken in the future)

$$V(t, x_t) = \min_{u_{t:T}} J_t(x_t; u_{t:T})$$

Utilizing Principle of Optimality

$$\underbrace{\min_{u_{t:T}} J_t(x_t; u_{t:T})}_{V(t,x_t)} = \min_{u_{t:t'}} \left\{ \int_t^{t'} l_\tau(x_\tau, u_\tau) d\tau + \underbrace{\min_{u_{t':T}} J_{t'}(x_{t'}; u_{t':T})}_{V(t',x_{t'})} \right\}$$

$V(t', x_{t'})$ optimal cost-to-go

$$0 = \min_{u_{t:t'}} \left\{ \int_t^{t'} l_\tau(x_\tau, u_\tau) d\tau + V(t', x_{t'}) - V(t, x_t) \right\}$$

$$0 = \lim_{t' \to t^+} \min_{u_{t:t'}} \left\{ \frac{1}{t'-t} \int_t^{t'} l_\tau(x_\tau, u_\tau) d\tau + \frac{V(t', x_{t'}) - V(t, x_t)}{t'-t} \right\}$$

Utilizing Principle of Optimality

$$0 = \lim_{t' \to t^+} \min_{u_{t:t'}} \left\{ \frac{1}{t'-t} \int_t^{t'} l_\tau(x_\tau, u_\tau) d\tau + \frac{V(t', x_{t'}) - V(t, x_t)}{t'-t} \right\}$$

Under smoothness assumptions of all functions

$$\lim_{t' \to t^+} \frac{1}{t'-t} \int_t^{t'} l_\tau(x_\tau, u_\tau) d\tau = l_t(x_t, u_t)$$

$$\lim_{t' \to t^+} \frac{V(t', x_{t'}) - V(t, x_t)}{t'-t} = \frac{\partial}{\partial t} V(t, x_t) + \frac{\partial}{\partial x} V(t, x_t) \dot{x}_t$$

Hamilton-Jacobi-Bellman equation

$$0 = \frac{\partial}{\partial t} V(t, x_t) + \min_{u_t} \{ l_t(x_t, u_t) + \frac{\partial}{\partial x} V(t, x_t) f(x_t, u_t) \}$$

Hamilton-Jacobi-Bellman Equation

$$0 = \underbrace{\frac{\partial}{\partial t} V(t, x_t)}_{\dot{V}(t, x_t)} + \min_{u_t} \{ l_t(x_t, u_t) + \underbrace{\frac{\partial}{\partial x} V(t, x_t)}_{\nabla_x V(t, x_t)} f(x_t, u_t) \}$$

- HJB equation is a nonlinear partial differential equation in the value function.

- Its discrete-time version is the Bellman equation.

$$J_k^*(x_k) = \min_{u_k} \left\{ l_k(x_k, u_k) + J_{k+1}^*(x_{k+1}) \right\}$$

Hamilton-Jacobi-Bellman Equation

William Hamilton

Carl Jacobi

Richard Bellman

Dynamic Programing

- Solving the optimal control recursively by solving the HJB equation
- At time T, we have

$$V(T, x_T) = l_T(x_T)$$

- We solve the following optimization backward in time

$$\dot{V}(t, x_t) = -\min_{u_t}\{l_t(x_t, u_t) + \nabla_x V(t, x_t) f(x_t, u_t)\}$$

and obtain the optimal control policy

$$q_t^*(x_t) = \arg\min_{u_t}\{l_t(x_t, u_t) + \nabla_x V(t, x_t) f(x_t, u_t)\}$$

Dynamic Programing

- Through this backward pass, we can only obtain the control policy as a mapping from the state to the control.

- The optimal control sequence needs to be obtained in a forward pass considering the initial state x_0

$$\begin{aligned} x_0^* &= x_0 \\ u_t^* &= q_t^*(x_t^*), \forall t \in [0, T) \\ \dot{x}_t^* &= f(x_t^*, u_t^*), \forall t \in [0, T] \end{aligned}$$

Continuous Time Optimal Control

- Problem formulation and examples
- Dynamic programming
- Maximum principle
- Linear quadratic regulator

Connection between HJB and Hamiltonian

- Hamiltonian

$$H(t, x_t, u_t, \lambda_t) := l_t(x_t, u_t) + \lambda_t^T f(x_t, u_t)$$

- HJB

$$\dot{V}(t, x_t) = -\min_{u_t}\{l_t(x_t, u_t) + \nabla_x V(t, x_t) f(x_t, u_t)\}$$

- We have

$$\lambda_t = \nabla_x V(t, x_t)$$

Connection between HJB and Hamiltonian

- Similar to the derivation in discrete time case, the co-state λ_t is the Lagrangian multiplier for the dynamic equation $\dot{x} = f(x, u)$
- How does the co-state evolve in time?

Dynamics of the Co-State

$$\dot{V}(t, x_t) = -\min_{u_t}\{l_t(x_t, u_t) + \nabla_x V(t, x_t) f(x_t, u_t)\}$$

Using the Partial Derivative Lemma, under optimal control u_t

$$-\frac{\partial^2}{\partial x \partial t} V(t, x_t) = \frac{\partial}{\partial x} l_t(x_t, u_t) + \frac{\partial^2}{\partial x^2} V(t, x_t) f(x_t, u_t) + [\frac{\partial}{\partial x} f(x_t, t_t)]^T \frac{\partial}{\partial x} V(t, x_t)$$

Using the following identity

$$-\frac{d}{dt}\frac{\partial}{\partial x} V(t, x_t) = \frac{\partial^2}{\partial t \partial x} V(t, x_t) + \frac{\partial^2}{\partial x^2} V(t, x_t) f(x_t, u_t)$$

We have

$$-\frac{d}{dt}\underbrace{\frac{\partial}{\partial x} V(t, x_t)}_{\lambda_t} = \frac{\partial}{\partial x} l_t(x_t, u_t) + [\frac{\partial}{\partial x} f(x_t, t_t)]^T \underbrace{\frac{\partial}{\partial x} V(t, x_t)}_{\lambda_t}$$

Pontryagin's Maximum Principle

- Theorem:
  - If $x^*_{0:T}$ and $u^*_{0:T}$ are the optimal state and control trajectories starting at x_0, then there exits a co-state trajectory $\lambda^*_{0:T}$ with $\lambda^*_T = \frac{\partial}{\partial x} l_T(x^*_T)$ satisfying

$$\begin{aligned}
\dot{x}^*_t &= \frac{\partial}{\partial \lambda} H(t, x^*_t, u^*_t, \lambda^*_t) = f(x^*_t, u^*_t) \\
-\dot{\lambda}^*_t &= \frac{\partial}{\partial x} H(t, x^*_t, u^*_t, \lambda^*_t) = \frac{\partial}{\partial x} l_t(x^*_t, u^*_t) + \left[\frac{\partial}{\partial x} f(x^*_t, u^*_t)\right]^T \lambda^*_t \\
u^*_t &= \arg\min_u H(t, x^*_t, u, \lambda^*_t)
\end{aligned}$$

Maximum Principle in Continuous Time vs. Discrete Time

Continuous time

$$\dot{x}_t^* = \frac{\partial}{\partial \lambda} H(t, x_t^*, u_t^*, \lambda_t^*) = f(x_t^*, u_t^*)$$

$$-\dot{\lambda}_t^* = \frac{\partial}{\partial x} H(t, x_t^*, u_t^*, \lambda_t^*) = \frac{\partial}{\partial x} l_t(x_t^*, u_t^*) + \left[\frac{\partial}{\partial x} f(x_t^*, u_t^*)\right]^T \lambda_t^*$$

$$u_t^* = \arg\min_u H(t, x_t^*, u, \lambda_t^*)$$

Discrete time

$$x_{k+1}^* = \frac{\partial}{\partial \lambda} H_k(x_k^*, u_k^*, \lambda_{k+1}^*) = f(x_k^*, u_k^*)$$

$$\lambda_k^* = \frac{\partial}{\partial x} H_k(x_k^*, u_k^*, \lambda_{k+1}^*) = \frac{\partial}{\partial x} l_k(x_k^*, u_k^*) + \left[\frac{\partial}{\partial x} f(x_k^*, u_k^*)\right]^T \lambda_{k+1}^*$$

$$u_k^* = \arg\min_u H_k(x_k^*, u, \lambda_{k+1}^*)$$

Continuous Time Optimal Control

- Problem formulation and examples
- Dynamic programming
- Maximum principle
- Linear quadratic regulator

Linear Quadratic Regulator

$$J = \frac{1}{2}\int_{t=0}^{T}[x_t^T Q_t x_t + u_t^T R_t u_t]dt + \frac{1}{2}x_T^T S_T x_T$$

$$\dot{x} = A_t x + B_t u$$

Q_t, S_T	symmetric positive semi-definite
R_t	symmetric positive definite
A_t, B_t	controllable

Linear Quadratic Regulator

$$J = \frac{1}{2}\int_{t=0}^{T} [x_t^T Q_t x_t + u_t^T R_t u_t] dt + \frac{1}{2} x_T^T S_T x_T$$

$$\dot{x} = A_t x + B_t u$$

Hamiltonian

$$H(t, x_t, u_t, \lambda_t) = \frac{1}{2} x_t^T Q_t x_t + \frac{1}{2} u_t^T R_t u_t + \lambda_t^T (A_t x_t + B_t u_t)$$

Maximum principle

$$-\dot{\lambda}_t = \frac{\partial}{\partial x} H(t, x_t, u_t, \lambda_t) = Q_t x_t + A_t^T \lambda_t$$

$$0 = \frac{\partial}{\partial u} H(t, x_t, u_t, \lambda_t) = R_t u_t + B_t^T \lambda_t \quad \Rightarrow \quad u_t = -R_t^{-1} B_t^T \lambda_t$$

Optimal Trajectory

- According to the maximum principle, the optimal trajectory satisfies the following ODE

$$\dot{x} = A_t x_t - B_t R_t^{-1} B_t^T \lambda_t$$
$$\dot{\lambda} = -Q_t x_t - A_t^T \lambda_t$$

- with boundary condition x_0 and

$$\lambda_T = \frac{\partial}{\partial x} l_T(x_T) = S_T x_T$$

Algebraic Riccati Equation

$$
\begin{aligned}
\dot{x} &= A_t x_t - B_t R_t^{-1} B_t^T \lambda_t \\
\dot{\lambda} &= -Q_t x_t - A_t^T \lambda_t
\end{aligned}
$$

- Assume that $\lambda_t = P_t x_t$ and $P_T = S_T$

$$
\begin{aligned}
0 &= \dot{\lambda} + Q_t x_t + A_t^T \lambda_t \\
&= \dot{P}_t x_t + P_t \dot{x}_t + Q_t x_t + A_t^T \lambda_t \\
&= \dot{P}_t x_t + P_t A_t x_t - P_t B_t R_t^{-1} B_t^T P_t x_t + Q_t x_t + A_t^T P_t x_t
\end{aligned}
$$

$$
\boxed{\dot{P}_t + P_t A_t + A_t^T P_t - P_t B_t R_t^{-1} B_t^T P_t + Q_t = 0}
$$

Implementation of LQR

- Obtain system dynamic model $\dot{x} = A_t x + B_t u$
- Specify the matrices Q_t, S_T, and R_t in the objective function

$$J = \frac{1}{2}\int_{t=0}^{T}[x_t^T Q_t x_t + u_t^T R_t u_t]dt + \frac{1}{2}x_T^T S_T x_T$$

- Solve for P_t in the ARE backward in time (offline)
- Real-time control: at time t, we apply the following optimal control

$$u_t = -R_t^{-1} B_t^T P_t x_t$$

Comparison

	Continuous Time	Discrete Time
System Dynamics	$\dot{x} = A_t x + B_t u$	$x_{k+1} = A_k x_k + B_k u_k$
Cost Function	$J = \frac{1}{2}\int_{t=0}^{T} [x_t^T Q_t x_t + u_t^T R_t u_t] dt + \frac{1}{2} x_T^T S_T x_T$	$J = \frac{1}{2}\sum_{k=0}^{N-1} [x_k^T Q_k x_k + u_k^T R_k u_k] + \frac{1}{2} x_N^T S_N x_N$
Optimal Control Law	$u_t = -R_t^{-1} B_t^T P_t x_t$	$u_k = -\underbrace{[R_k + B_k^T P_{k+1} B_k]^{-1} B_k^T P_{k+1} A_k}_{K_k} x_k$
Riccati Equation	$\dot{P}_t + P_t A_t + A_t^T P_t - P_t B_t R_t^{-1} B_t^T P_t + Q_t = 0$	$P_k = Q_k + A_k^T P_{k+1} A_k - A_k^T P_{k+1} B_k [R_k + B_k^T P_{k+1} B_k]^{-1} B_k^T P_{k+1} A_k$

Continuous Time Optimal Control

- Problem formulation and examples
- Dynamic programming
- Maximum principle
- Linear quadratic regulator