ICCAS 2016 Tutorial: Optimal Control

Jun Moon

School of Electrical and Computer Engineering
Ulsan National Institute of Science and Technology (UNIST)

October 16, 2016

Outline

- Introduction: Optimal Control Problem (MDP)
- Dynamic Programming
- Linear Quadratic Gaussian Control
  - Perfect State Information
  - Partially Observed Problem
  - Infinite-Horizon Problem
- Continuous-Time Optimal Control

Outline

Introduction: Optimal Control

The discrete-time dynamical system

$$x_{k+1} = f_k(x_k, u_k) + w_k$$

- $x_0 \sim \mathcal{N}(0, X)$, $w_k \sim \mathcal{N}(0, W)$, $v_k \sim \mathcal{N}(0, V)$
- $\{w_k\}$, $\{v_k\}$, x_0: uncorrelated
- u_k: control or decision variable that has access to $\{x_0, \ldots, x_k\}$
  $\Rightarrow$ state feedback problem
- The system is a Markov process

Note that

$$\mathbb{P}(x_{k+1} \in A | x_k, \ldots, x_0, u_k, \ldots, u_0) = \mathbb{P}(x_{k+1} \in A | x_k, u_k)$$

That is, to determine the state at $k+1$, we only need x_k and u_k.

Introduction: Optimal Control

The discrete-time dynamical system: $x_{k+1} = f(x_k, u_k) + w_k$

The expected cost function:

$$J(u) = \mathbb{E}\Big[g(x_K) + \sum_{k=0}^{K-1} l_k(x_k, u_k)\Big]$$

Optimal Control Problem

The optimal control problem is to find $u^* = \{u_0^* \ldots, u_{K-1}\}$ that minimizes the expected cost function, i.e., for any admissible control u

$$J(u^*) \leq J(u)$$

- This is the finite-horizon optimal control problem
- The optimal decision u^* cannot be made at each stage separately, since the dynamical system affects the future cost
- The problem is also called the Markov Decision Process (MDP)

Introduction: Optimal Control

Types of the expected cost functions (finite-horizon):

- Discounted cost function with $\alpha \in (0, 1)$

$$J(u) = \mathbb{E}\Big[g(x_K) + \sum_{k=0}^{K-1} \alpha^k l_k(x_k, u_k)\Big]$$

- Risk-sensitive cost function with $\delta > 0$

$$J(u) = \delta \log \mathbb{E}\Big[\exp\Big(\frac{1}{\delta} g(x_K) + \frac{1}{\delta} \sum_{k=0}^{K-1} l_k(x_k, u_k)\Big)\Big]$$

Let $\Phi = g(X_K) + \sum_{k=0}^{K-1} l_k(x_k, u_k)$. Then

$$J(u) = \mathbb{E}[\Phi] + \frac{1}{2\delta} var(\Phi) + o(\frac{1}{\delta})$$

Introduction: Optimal Control

Types of the expected cost functions (infinite-horizon):

- Discounted cost function with $\alpha \in (0, 1)$

$$J(u) = \mathbb{E}\Big[\sum_{k=0}^{\infty} \alpha^k l_k(x_k, u_k)\Big]$$

- Average cost function

$$J(u) = \limsup_{K \to \infty} \frac{1}{K} \mathbb{E}\Big[\sum_{k=0}^{K-1} l_k(x_k, u_k)\Big]$$

- Risk-sensitive cost function

$$J(u) = \limsup_{K \to \infty} \frac{\delta}{K} \log \mathbb{E}\Big[\exp\Big(\frac{1}{\delta} \sum_{k=0}^{K-1} l_k(x_k, u_k)\Big)\Big]$$

Introduction: Optimal Control

In my personal opinion, there are two approaches to solve the optimal control problem

- Dynamic programming (R. Bellman, 1953)
- Maximum principle (Pontryagin, 1956)

Applications

- control
- signal processing
- communications
- machine learning
- radar (target detection and estimation)
- autonomous driving systems
- economics
- many applications

Outline

Dynamic Programming: Principle of Optimality

The discrete-time dynamical system: $x_{k+1} = f_k(x_k, u_k) + w_k$

The expected cost function:

$$J(u) = \mathbb{E}\Big[g(x_K) + \sum_{k=0}^{K-1} l_k(x_k, u_k)\Big]$$

Principle of Optimality (R. Bellman)

Let $u^* = \{u_0^*, \ldots, u_{K-1}^*\}$ be an optimal control that minimizes the expected cost, and $\{x_0, \ldots, x_{K-1}\}$ be the corresponding optimal state trajectory. Consider the subproblem at x_i from $s = i$ to K

$$\mathbb{E}\Big[g(x_K) + \sum_{s=i}^{K-1} l_s(x_s, u_s)\Big]$$

Then the truncated optimal control $\{u_i^*, \ldots, u_{K-1}^*\}$ is optimal for this subproblem.

Dynamic Programming: Principle of Optimality

Verification:

- If the truncated control $\{u_i^*, \ldots, u_{K-1}^*\}$ were not optimal
- at time $s = i$, we are able to reduce the cost further by switching to an optimal control for the subproblem
- then $u^* = \{u_0^*, \ldots, u_{K-1}^*\}$ is not an optimal control

Principle of optimality: an optimal control can be constructed in backward in time

- At $i = K-1$, u_{K-1}^* must solve the corresponding subproblem
- At $i = K-2$, $\{u_{K-2}^*, u_{K-1}^*\}$ must solve the corresponding subproblem
- $K-3$,$K-4$, $K-5$, $\ldots$
- at $i = 0$, $\{u_0^*, \ldots, u_{K-2}^*, u_{K-1}^*\}$ must solve the corresponding problem
- The optimal control can be constructed sequentially

Dynamic Programming: Principle of Optimality

The Original Statement of Principle of Optimality by R. Bellman

An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision

- Why called "dynamic programming"?
- "The 1950s were not good years for mathematical research., ..., Try thinking of some combination that will possibly give it a pejorative meaning. It's impossible. Thus, I thought dynamic programming was a good name. It was something not even a Congressman could object to. So I used it as an umbrella for my activities."

Dynamic Programming

Dynamic programming algorithm: Define the value function

$$V_i(x_i) = \min_{\{u_i,\ldots,u_{K-1}\}} \mathbb{E}\Big[g(x_K) + \sum_{s=i}^{K-1} l_s(x_s, u_s)\Big]$$

$$V_K(x_K) = g(x_K)$$

Note that $V_0(x_0)$ is the optimal cost. Then (informally)

$$\begin{aligned}
V_i(x_i) &= \min_{\{u_i,\ldots,u_{K-1}\}} \mathbb{E}\Big[g(x_K) + \sum_{s=i}^{K-1} l_s(x_s, u_s)\Big] \\
&= \min_{\{u_i,\ldots,u_{K-1}\}} \mathbb{E}\Big[l_i(x_i, u_i) + g(x_K) + \sum_{s=i+1}^{K-1} l_s(x_s, u_s)\Big] \\
&= \min_{u_i} \mathbb{E}\Big[l_i(x_i, u_i) + \underbrace{\min_{\{u_{i+1},\ldots,u_{K-1}\}} \Big(\mathbb{E}\Big[g(x_K) + \sum_{s=i+1}^{K-1} l_s(x_s, u_s)\Big]\Big)}_{\text{principle of optimality}}\Big]
\end{aligned}$$

Dynamic Programming

$$V_i(x_i) = \min_{u_i} \mathbb{E}\Big[l_i(x_i, u_i) + \underbrace{\min_{\{u_{i+1},\ldots,u_{K-1}\}} \Big(\mathbb{E}\Big[g(x_K) + \sum_{s=i+1}^{K-1} l_s(x_s, u_s) \Big] \Big)}_{\text{principle of optimality}} \Big]$$

$$= \min_{u_i} \mathbb{E}\Big[l_i(x_i, u_i) + V_{i+1}(x_{i+1}) \Big] \quad \text{definition of the value function}$$

$$= \min_{u_i} \mathbb{E}\Big[l_i(x_i, u_i) + V_{i+1}(f_i(x_i, u_i) + w_i) \Big] \quad \text{dynamics of } x_k$$

- $V_k(x_k)$: value function, cost-to-go at state x_k, cost-to-go function at time k

Dynamic Programming

Dynamic Programming

- The optimal control problem can be solved backward in time from $K-1$ to 0 by proceeding the following sequential algorithm:

$$V_K(x_K) = g(x_K)$$
$$V_k(x_k) = \min_{u_k} \mathbb{E}\Big[l_k(x_k, u_k) + V_{k+1}(f_k(x_k, u_k) + w_k)\Big]$$

- The optimal cost is $V_0(x_0)$ for every initial condition x_0
- If $u^* = \{u_0^*, \ldots, u_{K-1}^*\}$ solves the above sequential optimization problem from $K-1$ to 0, then it is a state feedback optimal control

- At each stage k, we have to solve an unconstrained static optimization problem
- Dynamic programming converts the constrained dynamic optimization problem into the sequence of unconstrained optimization problems

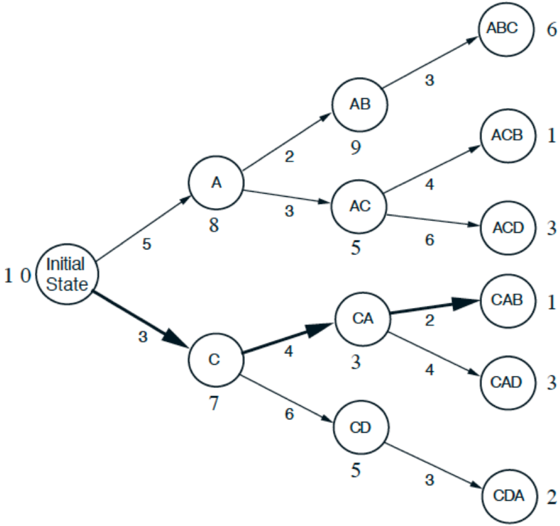
Dynamic Programming

The dynamic programming (DP) algorithm

$$V_K(x_K) = g(x_K)$$
$$V_k(x_k) = \min_{u_k} \mathbb{E}\Big[l_k(x_k, u_k) + V_{k+1}(f_k(x_k, u_k) + w_k)\Big], \; k = 0, 1, \ldots, K-1$$

- In many cases, the analytic solution of each optimization problem is hard to obtain (need some numerical approaches)
- For the LQ case, we can solve it analytically
- Sufficient condition for optimality ⇒ the sequence of optimal controls that solves the DP is the solution of the optimal control problem
- The maximum principle is a necessary condition for optimality

Dynamic Programming



Outline

3. Linear Quadratic Gaussian Control

Outline

- LQG Control: State Feedback

LQG Control: State Feedback

Linear dynamical system

$$x_{k+1} = Ax_k + Bu_k + w_k$$

Quadratic cost function

$$J(u) = \mathbb{E}\Big[x_K^T Z x_K + \sum_{k=0}^{K-1} x_k^T Q x_k + u_k^T R u_k\Big]$$

- $Z, Q \geq 0$ and $R > 0$: to make the (strictly) convex optimization problem (a unique linear optimal solution)
- $J(u) \geq 0$ for any admissible controls: minimization problem

LQG Control: State Feedback

The dynamic programming

$$V_K(x_K) = x_K^T Z x_K$$
$$V_k(x_k) = \min_{u_k} \mathbb{E}\Big[x_k^T Q x_k + u_k R u_k + V_{k+1}(Ax_k + Bu_k + w_k)\Big]$$

- We have to know the value function $V_k(x_k)$
- It is hard to know in general
- But for the LQ case, we can find an explicit value function $V_k(x_k)$ and the associated optimal control

LQG Control: State Feedback

- We will show that the value function is

$$
\begin{aligned}
V_k(x_k) &= x_k^T P_k x_k + q_k,\ V_K(x_K) = x_K^T Z x_K \\
P_k &= A^T P_{k+1} A + Q - A^T P_{k+1} B (B^T P_{k+1} B + R)^{-1} B^T P_{k+1} A,\ P_K \\
q_k &= \mathbb{E}[w_k^T P_{k+1} w_k] + q_{k+1} = \mathrm{Tr}(P_{k+1} W) + q_{k+1},\ q_K = 0
\end{aligned}
$$

- The unique linear state feedback optimal control

$$u_k^* = -(B^T P_{k+1} B + R)^{-1} B^T P_{k+1} A x_k$$

- Uniqueness: each subproblem admits a unique solution
- P_k backward: due to dynamic programming (principle of optimality)
- Optimal cost: $J(u^*) = x_0^T P_0 x_0 + \sum_{k=0}^{K-1} \mathrm{Tr}(P_{k+1} W)$

LQG Control: State Feedback

We use the (backward) induction. For $k = K$, by our assumption

$$V_K(x_K) = x_K^T P_K x_K + q_K = x_K^T Z x_K$$

Assume that it holds for $k+1$. Then

$$\begin{aligned}
V_k(x_k) &= \min_{u_k} \mathbb{E}\Big[x_k^T Q x_k + u_k R u_k + V_{k+1}(Ax_k + Bu_k + w_k)\Big] \\
&= \min_{u_k} \mathbb{E}\Big[x_k^T Q x_k + u_k R u_k + \underbrace{(Ax_k + Bu_k + w_k) P_{k+1} (Ax_k + Bu_k + w_k}_{\text{induction argument}} \\
&= x_k^T Q x_k + x_k^T A^T P_{k+1} A x_k + \mathbb{E}[w_k^T P_{k+1} w_k] + q_{k+1} \\
&\quad + \min_{u_k} \Big[\underbrace{u_k^T (B^T P_{k+1} B + R) u_k}_{\text{strictly convex}} + 2u_k^T B^T P_{k+1} A x_k\Big] \\
&= x_k^T \Big(A^T P_{k+1} A + Q - A^T P_{k+1} B (B^T P_{k+1} B + R)^{-1} B^T P_{k+1} A\Big) x_k \\
&\quad + \mathbb{E}[w_k^T P_{k+1} w_k] + q_{k+1} \quad \text{minimizer: } u_k^* = -(B^T P_{k+1} B + R)^{-1} B^
\end{aligned}$$

Outline

- LQG Control: Partially Observed Problem

LQG Control: Partially Observed Problem

Dynamical system

$$x_{k+1} = Ax_k + Bu_k + w_k$$
$$y_k = Cx_k + v_k$$

- $x_0 \sim \mathcal{N}(0, X)$, $w_k \sim \mathcal{N}(0, W)$, $v_k \sim \mathcal{N}(0, V)$
- $\{w_k\}$, $\{v_k\}$, x_0: uncorrelated

Partially observed optimal control problem

- The controller, u_k, has access to $\{y_0, \ldots, y_k\}$ for each k
- We minimize the same quadratic cost function as in the state feedback problem
- In general, this class of problems is extremely challenging, since we have to construct an estimator to estimate the state
- But for the LQG case, it is tractable (why?)

LQG Control: Partially Observed Problem

The associated Kalman filter is given by

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + A\Sigma_k C^T (C\Sigma_k C^T + V)^{-1}(y_k - C\hat{x}_k),\ \hat{x}_0 = 0$$
$$\Sigma_{k+1} = A\Sigma_k A^T + W - A\Sigma_k C^T (C\Sigma_k C^T + V)^{-1} C\Sigma_k A^T,\ \Sigma_0 = X$$

Note that

$$\hat{x}_k = \mathbb{E}[x_k | (y_0, \ldots, y_{k-1}), (u_0, \ldots, u_{k-1})]$$

The orthogonality principle (the projection theorem) implies for any $S \geq 0$

$$\mathbb{E}[(x_k - \hat{x}_k)^T S \hat{x}_k] = 0$$

Note also that $(x_k - \hat{x}_k)$ is independent with u_k

LQG Control: Partially Observed Problem

Then (let $e_k = x_k - \hat{x}_k$)

$$
\begin{aligned}
J(u) &= \mathbb{E}\Big[x_K^T Z x_K + \sum_{k=0}^{K-1} x_k^T Q x_k + u_k^T R u_k\Big] \\
&= \mathbb{E}\Big[(x_K - \hat{x}_K + \hat{x}_K)^T Z (x_K - \hat{x}_K + \hat{x}_K) \\
&\quad + \sum_{k=0}^{K-1} (x_k - \hat{x}_k + \hat{x}_k)^T Q (x_k - \hat{x}_k + \hat{x}_k) + u_k^T R u_k\Big] \\
&= \mathbb{E}\Big[\hat{x}_K Z \hat{x}_K + \sum_{k=0}^{K-1} \hat{x}_k^T Q \hat{x}_k + u_k^T R u_k\Big] + \mathbb{E}\Big[e_K^T Z e_k + \sum_{k=0}^{K-1} e_k Q e_k\Big] \\
&\quad + \underbrace{\mathbb{E}\Big[2(x_K - \hat{x}_K)^T Z \hat{x}_K + \sum_{k=0}^{K-1} 2(x_k - \hat{x}_k)^T Q \hat{x}_k\Big]}_{=0 \quad \text{orthogonality principle}}
\end{aligned}
$$

LQG Control: Partially Observed Problem

Then since e_k is independent with control, the partially observed optimal control problem is equivalent to minimizing

$$\bar{J}(u) = \mathbb{E}\Big[\hat{x}_K Z \hat{x}_K + \sum_{k=0}^{K-1} \hat{x}_k^T Q \hat{x}_k + u_k^T R u_k\Big]$$

$$\text{subject to } \ \hat{x}_{k+1} = A\hat{x}_k + Bu_k + \underbrace{A\Sigma_k C^T (C\Sigma_k C^T + V)^{-1} C(y_k - C\hat{x}_k)}_{\text{innovation term}}$$

- Note that $\hat{x}_k$ is completely available to the controller
- The innovation term can be considered as the additive stochastic noise as in the state feedback problem
- Hence, this problem is identical with the state feedback LQG problem

LQG Control: Partially Observed Problem

The optimal control is:

$$u_k^* = -(B^T P_{k+1} B + R)^{-1} B^T P_{k+1} A \hat{x}_k$$

- This is known as the separation principle (or the certainty equivalence principle)
- Optimal control and Kalman filter can be designed independently
- This is possible due to the orthogonality principle and the fact that the Kalman filter is the conditional expectation (the MMSE problem)
- The partially observed LQG problem is still a finite-dimensional optimal control problem

LQG Control: Partially Observed Problem

We can also solve the partially observed problem directly by dynamic programming:

Due to the partially observed information, we have to use the conditional expectation: Let $\mathcal{I}_k = \{y_0, \ldots, y_{k-1}, u_0, \ldots, u_{k-1}\}$. Then

$$V_K(x_K) = \mathbb{E}[x_K^T Z x_K | \mathcal{I}_K]$$

$$V_k(x_k) = \min_{u_k} \mathbb{E}\Big[x_k^T Q x_k + u_k R u_k + V_{k+1}(A x_k + B u_k + w_k) | \mathcal{I}_k\Big]$$

- The quadratic value function with the Riccati equation can also be used as in the state feedback case
- This is because of the orthogonality principle

Outline

- LQG Control: Infinite-Horizon Problem

LQG Control: Infinite-Horizon Problem

The infinite-horizon problem:

$$J(u) = \limsup_{K\to\infty} \frac{1}{K}\mathbb{E}\Big[\sum_{k=0}^{K-1} x_k^T Q x_k + u_k^T R u_k\Big]$$

Consider two Riccati difference equations:

$$P_k = A^T P_{k+1} A + Q - A^T P_{k+1} B (B^T P_{k+1} B + R)^{-1} B^T P_{k+1} A, \ P_K = Z$$
$$\Sigma_{k+1} = A\Sigma_k A^T + W - A\Sigma_k C^T (C\Sigma_k C^T + V)^{-1} C\Sigma_k A^T, \ \Sigma_0 = X$$

- P_k is backward: dynamic programming
- Σ_k is forward: estimation is causal

Note that if $(A, W^{1/2})$ controllable and (A, C) observable:

$$\lim_{k\to\infty} \Sigma_k = \Sigma, \quad \Sigma = A\Sigma A^T + W - A\Sigma C^T (C\Sigma C^T + V)^{-1} C\Sigma A^T$$

LQG Control: Infinite-Horizon Problem

Let $s = K - k$. Then

$$P_{s+1} = A^T P_s A + Q - A^T P_s B(B^T P_s B + R)^{-1} B^T P_s A, \;\; P_0 = Z$$

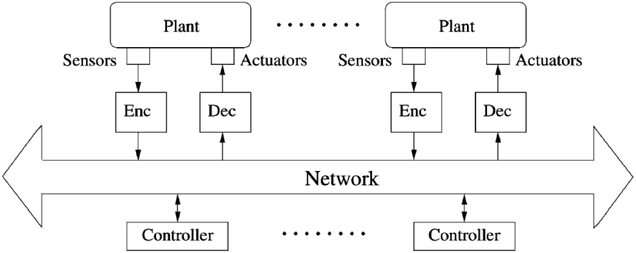
Hence, if (A, B) is controllable and $(A, Q^{1/2})$ is observable

$$\lim_{s \to \infty} P_s = P, \quad P = A^T PA + Q - A^T PB(B^T PB + R)^{-1} B^T PA$$

- Algebraic Riccati equation: LMI and semi-definite programming
- We can use P in the optimal control and Σ in Kalman filter for the infinite-horizon optimal control system

Recent Work on LQG: Networked Control Systems

Control over unreliable networks



- Limitations: finite capacity, delay, and data losses in communication networks
- Stability and performance depend on both the dynamical system and the underlying networks
- Applications: CPS, UAVs, tracking system, sensor networks, etc

Outline

Dynamic Programming: HJB Equation

Continuous-time SDE

$$dx(t) = f(t, x, u)dt + DdW(t)$$

The objective function

$$J(u) = \mathbb{E}\left[g(x(T)) + \int_0^T l(s, x, u)ds\right]$$

- The optimal control is one that minimizes the objective function
- As in the discrete-time case, we may use other types of the objective functions (e.g. discounted, risk-sensitive, etc.)
- Two approaches: dynamic programming and the stochastic maximum principle

Dynamic Programming: HJB Equation

The value function

$$V(t,x) = \inf_{u(t:T)} \mathbb{E}\Big[g(x(T)) + \int_t^T l(s,x,u)ds\Big]$$

Note that $V(0,x)$ is the optimal cost. Then (informally)

$$V(t,x) = \inf_{u(t:T)} \mathbb{E}\Big[\int_t^{t+\Delta t} l(s,x,u)ds + g(x(T) + \int_{t+\Delta t}^T l(s,x,u)ds\Big]$$

$$= \inf_{u(t:t+\Delta t)} \mathbb{E}\Big[\int_t^{t+\Delta t} l(s,x,u)ds + \underbrace{\inf_{u(t+\Delta t:T)} \mathbb{E}\Big[g(x(T) + \int_{t+\Delta t}^T l(s,x}_{\text{principle of optimality}}$$

$$= \inf_{u(t:t+\Delta t)} \mathbb{E}\Big[\int_t^{t+\Delta t} l(s,x,u)ds + \underbrace{V(t+\Delta t, x+\Delta x)}_{\text{value function}}\Big]$$

Dynamic Programming: HJB Equation

For sufficiently small Δt, by the chain rule of the stochastic calculus (note that $V_{xx} = [\frac{\partial^2 V}{\partial x_i x_j}]_{ij}$)

$$-\frac{\partial V}{\partial t} = \min_u \left[l(t,x,u) + \frac{\partial V}{\partial x}^T f(t,x,u) + \frac{1}{2}\mathrm{Tr}(V_{xx}DD^T) \right]$$

- This is the Hamilton-Jacobi-Bellman (HJB) equation, and is the 2nd order parabolic PDE
- Sufficient condition for optimality: If u^* minimizes the HJB equation and the solution of the HJB, V, then u^* is the optimal control
- In general, we cannot find the classical solution of the HJB $\Rightarrow$ viscosity solution (Lions, 1983)
- But for the LQ case, we have an explicit solution

Continuous-Time LQG: State Feedback

The linear SDE and the quadratic cost function

$$
\begin{aligned}
dx(t) &= [Ax(t) + Bu(t)]dt + DdW(t) \\
J(u) &= \mathbb{E}\left[x^T(T)Zx(T) + \int_0^T x^T(t)Qx(t) + u^T(t)Ru(t)dt\right]
\end{aligned}
$$

The value function is quadratic (similar to the Lyapunov function)

$$
\begin{aligned}
V(x) &= x^T M(t)x + q(t) \\
-\frac{dM}{dt} &= A^T M + MA + Q - MBR^{-1}B^T M, \quad M(T) = Z \\
q(t) &= \int_t^T \mathrm{Tr}(M(s)DD^T)ds, \; q(T) = 0
\end{aligned}
$$

The unique linear state feedback optimal control:

$$
u^*(t) = -R^{-1}B^T M(t)x(t)
$$

Continuous-Time LQG: State Feedback

We can show that

$$-x^T \frac{dM}{dt} x = \min_u \Big[x^T Q x + u^T R u + 2x^T M(t)(Ax + Bu) + \mathrm{Tr}(M(t)DD^T) \Big]$$

The corresponding minimizer: $u^*(t) = -R^{-1} B^T M(t) x$,

$$-x^T \frac{dM}{dt} x = x^T \Big(A^T M + MA + Q - MBR^{-1} B^T M \Big) x + \mathrm{Tr}(M(t)DD^T)$$

- Note that for the LQG case, the HJB PDE becomes ODE
- This is due to the linear-quadratic structure of the problem
- In general, the HJB equation is very hard to solve

Continuous-Time LQG: Remarks

- Partially observed problem: as in the discrete-time case, the separation principle holds
- The optimal control is $u^*(t) = -R^{-1}B^T M(t)\hat{x}(t)$ ($\hat{x}$: Kalman-Bucy filter)
- Infinite-horizon problem: $M(t)$ is the backward RDE, and its convergence condition coincides with that of the discrete-time case
- In this case, we have to the solve the algebraic Riccati equation

$$0 = A^T M + MA + Q - MBR^{-1}B^T M$$

Recent Work on LQG: Large-Scale LQG (MFGs)

SDE for agent i, $1 \leq i \leq N$

$$dx_i(t) = [A_i x_i(t) + B_i u_i(t)]dt + D_i dW_i(t)$$

Objective function for agent i

$$J_i(u_i) = \mathbb{E} \int_0^T \left[\left\| x_i(t) - \frac{1}{N} \sum_{i=1}^{N} x_i(t) \right\|_{Q_i}^2 + \|u_i(t)\|_{R_i}^2 \right] dt$$

- $\frac{1}{N}\sum_{i=1}^{N} x_i(t)$: mean field (empirical measure)
- We cannot solve the HJB when N is large (curse of dimensionality)
- Need to control the individual SDEs in a distributed manner
- Applications: smart grid, traffic networks, and economics, etc

Outline

5 Conclusions

Conclusions

Other optimal control problems

- Model predictive control (MPC)
- Robust (H^∞) control
- Risk-sensitive control
- Optimal control of Markov jump systems
- Differential and dynamic games (multi-agent problem)
- others...

Dynamic programming is hard to solve in general due to its great computational expense; therefore there are various efficient computation algorithms

References

Books on stochastic optimal control

- D. P. Bertsekas, Dynamic Programming and Optimal control
- D. P. Bertsekas and J. Tsitsiklis, Neuro-Dynamic Programming
- R. Bellman, Dynamic Programming
- R. Sutton and A. Barto, Reinforcement Learning
- D. Liberzon, Calculus of Variations and Optimal Control Theory
- W. Fleming and H. Soner, Controlled Markov Processes and Viscosity Solutions
- W. Fleming and W. Rishel, Deterministic and Stochastic Optimal Control
- J. Yong and X. Zhou, Stochastic Controls
- T. Başar and G. Olsder, Dynamic Noncooperative Game Theory