A High-Level Overview of Path Integral Control

Pierre-Luc Bacon

August 22, 2022

Discrete-Time OCP

$$
\begin{aligned}
&\text{minimize} \quad c_T(x_T) + \sum_{t=1}^{T-1} c_t(x_t, u_t) \\
&\text{subject to} \quad x_{t+1} = f_t(x_t, u_t), \quad \text{for} \quad t = 1, \ldots, T-1 \\
&\text{given} \quad x_1 \quad .
\end{aligned}
$$

The **value function** of a parametric mathematical program is the mapping from program parameters to optimal value. Here the "parameters" (inputs) are x_1 and the value is $c_T(x_T^\star) + \sum_{t=1}^{T-1} c_t(x_t^\star, u_t^\star)$ where $\{x_t^\star\}_{t=2}^{T}$ and $\{u_t^\star\}_{t=1}^{T-1}$.

Bellman Optimality Equations

In the optimal control context, the value function is called the cost-to-go function and satisfies the Bellman optimality equations:

$$
\begin{aligned}
J_T(x_T) &\triangleq c_T(x_T) \\
J_t(x_t) &\triangleq \min_{u_t \in \mathcal{U}(x_t)} \left\{ c_t(x_t, u_t) + J_{t+1}(f_t(x_t, u_t)) \right\} \quad t = 1, \ldots, T-1.
\end{aligned}
$$

The above are DP equations in a discrete-time finite horizon MDP.

Special case: Linear Quadratic Regulation

$$
\begin{aligned}
&\text{minimize} \;\; x_T Q_T x_T + \sum_{t=1}^{T} \left(x_t^\top Q x_t + u_t^\top R u_t \right) \\
&\text{subject to} \;\; x_{t+1} = A x_t + B u_t, \;\; t = 1, \ldots T - 1 \\
&\text{given} \;\; x_1 \quad .
\end{aligned}
$$

Backward Induction for LQR: Ricatti Equation

The local minimization problem in the Bellman optimality equations can be solved in closed-form under the LQR setting.

The cost-to-go function is quadratic and of the form $J_t(x_t) \triangleq x_t^\top P_t x_t$ for all $t = 1, \ldots, T$. and the matrices $\{P_1, \ldots, P_T\}$ can be found by backward induction:

- Set $P_T = Q_T$
- From $t = T - 1, \ldots, 1$:
  - Set $P_t = Q + A^\top P_{t+1} A - A^\top P_{t+1} B (R + B^\top P_{t+1} B)^{-1} B^\top P_{t+1} A$
  - Set $K_t = -(R + B^\top P_{t+1} B)^{-1} B^\top P_{t+1} A$

You can then compute the optimal control at time t in state x_t with $u_t^\star = K_t x_t$ (the optimal controls are linear in the states).

Continuous-Time

Is there also a Dynamic Programming approach to continuous-time OCPs?

$$\begin{aligned}
&\text{minimize} \;\; c(x(t_f)) + \int_{t_0}^{t_f} c(x(t), u(t))dt \\
&\text{subject to} \;\; \dot{x}(t) = f(x(t), u(t)) \\
&\text{given} \;\; x(t_0) = x_0 \quad .
\end{aligned}$$

The Hamilton-Jacobi Equations

We can show that the cost-to-go for the above continuous-time problem function satisfies for all x, t:

$$J(x, T) = c(x) \quad \text{and} \quad 0 = \min_{u \in \mathcal{U}} \left\{ c(x, u) + \underbrace{D_2 J(x, t)}_{\text{time derivative}} + \overbrace{D_1 J(x, t)}^{\text{space derivative}} f(x, u) \right\}$$

Note that while the OCP only involves an ODE, the HJB are partial differential equations (PDEs): a specification of how the partial derivatives of a function ought to behave together.

Theorem (Sufficiency). Let $V(x,t)$ be a solution to the PDE:

$$0 = \min_{u \in \mathcal{U}} \left\{ c(x,u) + D_2 V(x,t) + D_1 V(x,t) f(x,u) \right\} \text{ for all } x, t \quad ,$$

and boundary condition $V(x, t_f) = c(x)$ for all x. Then V is the cost-to-go function J and an optimal policy $\mu(x,t)$ can be obtained by minimizing the expression above given $V(x,t)$.

Potential issue: We assumed that J is differentiable, but this may not be the case, and we may not be able to solve the corresponding HJB equations. But if we happen to do find a solution, analytically, or numerically, then we're in good shape!

Stochastic Optimal Control via and Path Integral

Problem formulation

The problem formulation is still in continuous-time, but we now add noise: ie the dynamics are described by stochastic differential equations (SDEs) rather than ODEs.

We also make two important assumptions:

- Control-affine dynamics
- Quadratic cost

SDE Dynamics

The PI framework assumes that we have control-affine dynamics: the system has a linear relationship on the controls but can be nonlinear elsewhere:

$$dX(t) = \underbrace{f(X(t), t)dt}_{\text{drift}} + \underbrace{g(X(t), t)\left(u(X(t), t)dt + dW(t)\right)}_{\text{diffusion}}$$

- The "drift" term is also sometimes referred to as the "passive dynamics" of the system (eg. gravity).
- $dW(t)$ is Gaussian noise with $\mathbb{E}\left[dW(t)\right] = 0$

Euler-Maruyama

The Euler discretization counterpart for SDEs is the "Euler-Maruyama" method, which for an SDE

$$dX(t) = \underbrace{\mu(X(t), t))dt}_{\text{drift}} + \underbrace{\sigma(X(t), t)dW(t)}_{\text{diffusion}} \quad ,$$

we compute the random sequence:

$$X_{k+1} = X_k + \mu(X_k, kh)h + \sigma(X_k, kh)\Delta W_k \quad \text{where} \quad \Delta W_k \sim \mathcal{N}(0, \sqrt{h}) \quad .$$

for a fixed discretization of the interval $[0, T]$ into n bins of width $h = T/n$.

Time-Discretization

Consider our control-affine system:

$$dX(t) = f(X(t), t)dt + g(X(t), t)\left(u(X(t), t)dt + dW(t)\right) \quad ,$$

Furthermore, let's be a bit more precise and assume that $x \in \mathbb{R}^m$, $u \in \mathbb{R}^n$ so that $g :\in \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{m \times n}$ is a **control matrix** and $f :\in \mathbb{R}^m \to \mathbb{R}^m$. The Euler-Maruyma discretization is:

$$x_{k+1} = x_k + f_k h + G_k\left(u_k h + \Delta w_k\right), \quad \Delta W_k \sim \mathcal{N}\left(0, \sqrt{h}\right) \quad .$$

Gaussian Process

$$x_{k+1} = x_k + f_k h + G_k \left(u_k h + \Delta w_k \right), \quad \Delta W_k \sim \mathcal{N}\left(0, \sqrt{h}\right) \quad .$$

The above is a discrete-time Markov Decision Process whose transition probability function is given by:

$$p(x_{k+1} | x_k, u_k) = \mathcal{N}\left(x_k + f_k h + G_k u_k h, G_k \Sigma G_k^\top\right)$$

where Σ is the covariance matrix of the Wiener process. To see this, remember the "reparameterization" properties of the Normal distribution, ie. if $X \sim \mathcal{N}(\mu, \Sigma)$ with $\mu \in \mathbb{R}^n, \Sigma \in \mathbb{R}^{n \times n}$ and under the affine transformation: $Y = a + BX$ then the transformed variable is distributed as $Y \sim \mathcal{N}\left(a + B\mu, B\Sigma B^\top\right)$.

Discrete-time MDP Counterpart

Having a discrete-time Markov process, we can define an approximate MDP counterpart:

$$
\begin{aligned}
J(x_T, T) &\triangleq c_T(x_T) \\
J(x_t, t) &\triangleq \min_{u_t \in \mathcal{U}(x_t)} \left\{ c_t(x_t, u_t) + \mathbb{E}\left[J(X_{t+1}, t+1) \mid U_t = u_t \right] \right\} .
\end{aligned}
$$

The expectation is now taken under the Gaussian defined above and the cost function is assumed to be quadratic in the controls:

$$
c_t(x_t, u_t) \triangleq l(x, t) + u^\top R_t u \quad .
$$

where l can be a nonlinear function.

HJB Equation for PI Control

If we take the second-order Taylor expansion of J, replace it back into the discrete-time Bellman's equations and take the limit of the step size to 0, we get an HJB equation of the form:

$$D_2 J(x,t) = l(x,t) + \min_{u \in \mathcal{U}} \left(u^\top R_t u + D_1 J(x,t)(f_t + G_t u) + \frac{1}{2} \mathrm{tr} \left(D_1^2 J(x,t) G_t \Sigma G_t^\top \right) \right) ,$$

Because of the control-affine assumption, we can solve for u directly and get that the minimizing action $u_t^\star$ is:

$$u_t^\star = -R^{-1} G_t^\top \left(D_1 J(x,t) \right)^\top$$

But how can we solve this PDE?

Exponential Transformation

We are going to transform the above PDE by setting:

$$J(x,t) = -\lambda \log \Psi(x,t) \quad ,$$

for some function Ψ satisfying:

$$\begin{aligned}
&-D_2\Psi(x,t) = \\
&\qquad -\frac{1}{\lambda}l(x,t)\Psi(x,t) + D_1\Psi(x,t)f_t + \frac{1}{2}\mathrm{tr}\left(D_1^2\Psi(x,t)G_t\Sigma G_t^\top\right)
\end{aligned}$$

This PDE is special: it's called the Kolmogorov backward PDE, a second-order **linear** PDE.

Feynman-Kac

The Feynman-Kac theorem allows us to solve such PDEs (Kolmogorov) numerically, by showing that their solution is that of an expectation (an integral) over paths.

Theorem (informal) Feynman-Kac (informal) If $X(t)$ satisfies an SDE of the form $dX(t) = f(X(t), t)dt + G(X(t))dW(t)$ then:

$$\Psi(x, t) = \mathbb{E}\left[\exp\left(-\frac{1}{\lambda}c_T(X(T)) - \frac{1}{\lambda}\int_t^T l(X(t), t)dt\right)\right]$$

iff Ψ satisfies the Backward-Kolmogorov PDE:

$$-D_2\Psi(x, t) = \\ -\frac{1}{\lambda}l(x, t)\Psi(x, t) + D_1\Psi(x, t)f_t + \frac{1}{2}\mathrm{tr}\left(D_1^2\Psi(x, t)G_t\Sigma G_t^\top\right)$$

Recap

1. We posed the stochastic optimal control problem under the assumptions of control-affine SDE dynamics + quadratic cost on the controls
2. The corresponding SOCP can be solved via the HJB equation, a PDE
3. To solve the PDE, we use the exponetial transform: this gives us a **linear PDE**
4. This linear PDE happens to be the Backward-Kolmogorov PDE
5. The Backward-Kolmogorov PDE can be solved numerically using the Monte-Carlo method using the Feyman-Kac theorem