

AA 203

Optimal and Learning-Based Control

The Hamilton-Jacobi-Bellman equation

Autonomous Systems Laboratory

Stanford University

April 26, 2023
(last updated April 26, 2023)



Stanford
University

Agenda

1. The Bellman equation as a sufficient optimality condition
2. Continuous-time dynamic programming and the HJB equation
3. LQR control in continuous-time
4. Non-smooth value functions

Agenda

1. The Bellman equation as a sufficient optimality condition
2. Continuous-time dynamic programming and the HJB equation
3. LQR control in continuous-time
4. Non-smooth value functions

Review: Principle of optimality and dynamic programming

Consider the discrete-time OCP

$$\begin{aligned} & \underset{\{u_t\}_{t=0}^{T-1}}{\text{minimize}} \quad \ell_T(x_T) + \sum_{t=0}^{T-1} \ell(t, x_t, u_t) \\ & \text{subject to} \quad x_{t+1} = f(t, x_t, u_t), \quad \forall t \in \{0, 1, \dots, T-1\} \\ & \quad \quad \quad u_t \in \mathcal{U}, \quad \forall t \in \{0, 1, \dots, T-1\} \end{aligned}$$

Define the tail sub-problem *cost-to-go*

$$J(t, x, \{u_k\}_{k=t}^{T-1}) := \ell_T(x_T) + \sum_{k=t}^{T-1} \ell(k, x_k, u_k)$$

where $x_{k+1} = f(k, x_k, u_k)$ with initial condition $x_t = x$ is assumed implicitly.

Define the *value function* at $t \in \{0, 1, \dots, T\}$ and $x_t \in \mathbb{R}^n$ by

$$V(t, x_t) := \inf_{\{u_k\}_{k=t}^{T-1} \subseteq \mathcal{U}} J(t, x_t, \{u_k\}_{k=t}^{T-1}), \quad V(T, x_T) = \ell_T(x_T).$$

Previously we used “ $J_t^*(x_t)$ ”, but this notation will translate better to continuous-time later.

Review: Principle of optimality and dynamic programming

Suppose $\{u_t^*\}_{t=0}^{T-1} \subseteq \mathcal{U}$ is *globally optimal* for this OCP, i.e.,

$$J(0, x_0, \{u_t^*\}_{t=0}^{T-1}) = V(0, x_0).$$

Then the truncation $\{u_k^*\}_{k=t}^{T-1}$ is *globally optimal* for the corresponding tail sub-problem, i.e.,

$$J(t, x_t, \{u_k^*\}_{k=t}^{T-1}) = V(t, x_t).$$

From this, we must have the Bellman equation

$$V(t, x) = \inf_{u \in \mathcal{U}} \left(\ell(t, x, u) + V(t+1, f(t, x, u)) \right)$$

with the boundary condition $V(T, x) = \ell_T(x)$, for all $t \in \{0, 1, \dots, T-1\}$ and $x \in \mathbb{R}^n$.

So the Bellman equation above is a *necessary* condition for *global* optimality of $\{u_t^*\}_{t=0}^{T-1}$.

The Bellman equation as a sufficient optimality condition

Suppose $\hat{V} : \{0, 1, \dots, T\} \times \mathbb{R}^n \rightarrow \mathbb{R}$ is a function that satisfies the Bellman equation

$$\hat{V}(t, x) = \inf_{u \in \mathcal{U}} \left(\ell(t, x, u) + \hat{V}(t+1, f(t, x, u)) \right), \quad \hat{V}(T, x) = \ell_T(x),$$

for all $t \in \{0, 1, \dots, T-1\}$ and $x \in \mathbb{R}^n$.

Suppose (\hat{x}, \hat{u}) satisfy $\hat{x}_{t+1} = f(t, \hat{x}_t, \hat{u}_t)$ with initial condition \hat{x}_0 and

$$\ell(t, \hat{x}_t, \hat{u}_t) + \hat{V}(t+1, f(t, \hat{x}_t, \hat{u}_t)) = \inf_{u \in \mathcal{U}} \left(\ell(t, \hat{x}_t, u) + \hat{V}(t+1, f(t, \hat{x}_t, u)) \right),$$

for all $t \in \{0, 1, \dots, T-1\}$.

Then we can write

$$\hat{V}(t, \hat{x}_t) - \hat{V}(t+1, f(t, \hat{x}_t, \hat{u}_t)) = \ell(t, \hat{x}_t, \hat{u}_t).$$

The Bellman equation as a sufficient optimality condition

Then we can write

$$\hat{V}(t, \hat{x}_t) - \hat{V}(t+1, f(t, \hat{x}_t, \hat{u}_t)) = \ell(t, \hat{x}_t, \hat{u}_t)$$

$$\hat{V}(t, \hat{x}_t) - \hat{V}(t+1, \hat{x}_{t+1}) = \ell(t, \hat{x}_t, \hat{u}_t)$$

$$\sum_{t=0}^{T-1} \left(\hat{V}(t, \hat{x}_t) - \hat{V}(t+1, \hat{x}_{t+1}) \right) = \sum_{t=0}^{T-1} \ell(t, \hat{x}_t, \hat{u}_t)$$

$$\hat{V}(0, \hat{x}_0) - \hat{V}(T, \hat{x}_T) = \sum_{t=0}^{T-1} \ell(t, \hat{x}_t, \hat{u}_t)$$

$$\begin{aligned} \implies \hat{V}(0, \hat{x}_0) &= \ell_T(\hat{x}_T) + \sum_{t=0}^{T-1} \ell(t, \hat{x}_t, \hat{u}_t) \\ &= J(0, \hat{x}_0, \{\hat{u}_t\}_{t=0}^{T-1}) \end{aligned}$$

So $\hat{V}(0, \hat{x}_0)$ is the cost-to-go for $\{\hat{u}_t\}_{t=0}^{T-1}$.

The Bellman equation as a sufficient optimality condition

Now consider any other (x, u) satisfying $x_{t+1} = f(t, x_t, u_t)$ and $x_0 = \hat{x}_0$. Then

$$\begin{aligned}\hat{V}(t, x_t) &= \inf_{u \in \mathcal{U}} \left(\ell(t, x_t, u) + \hat{V}(t+1, f(t, x_t, u)) \right) \\ &\leq \ell(t, x_t, u_t) + \hat{V}(t+1, f(t, x_t, u_t))\end{aligned}$$

A similar summation argument gives us

$$\hat{V}(0, x_0) \leq J(0, x_0, \{u_t\}_{t=0}^{T-1}).$$

Since $x_0 = \hat{x}_0$, we have

$$\hat{V}(0, \hat{x}_0) \leq J(0, \hat{x}_0, \{u_t\}_{t=0}^{T-1}).$$

Overall, \hat{u} yields the cost $\hat{V}(0, \hat{x}_0)$ and no other admissible u can produce a smaller cost.

While we chose $t = 0$ as the initial time, this was arbitrary since the Bellman equation is assumed to hold for all $t \in \{0, 1, \dots, T\}$ and $x \in \mathbb{R}^n$. So $\hat{V}(t, x)$ is the optimal cost-to-go, i.e., $\hat{V}(t, x) = V(t, x)$ for all $t \in \{0, 1, \dots, T\}$ and $x \in \mathbb{R}^n$, and \hat{u} is a *globally optimal* control.

Agenda

1. The Bellman equation as a sufficient optimality condition
2. Continuous-time dynamic programming and the HJB equation
3. LQR control in continuous-time
4. Non-smooth value functions

Continuous-time dynamic programming

Consider the continuous-time OCP

$$\begin{aligned} & \underset{u}{\text{minimize}} \quad \ell_T(x(T)) + \int_0^T \ell(t, x(t), u(t)) dt \\ & \text{subject to} \quad \dot{x}(t) = f(t, x(t), u(t)), \quad \forall t \in [0, T] \\ & \quad \quad \quad u(t) \in \mathcal{U}, \quad \forall t \in [0, T] \end{aligned}$$

Define the tail sub-problem *cost-to-go*

$$J(t, x, u_{[t, T]}) := \ell_T(x(T)) + \int_t^T \ell(s, x(s), u(s)) ds$$

where $\dot{x}(s) = f(s, x(s), u(s))$ with initial condition $x(t) = x$ is assumed implicitly.

Define the *value function* at $t \in [0, T]$ and $x(t) \in \mathbb{R}^n$ by

$$V(t, x(t)) := \inf_{u_{[0, T]} \subseteq \mathcal{U}} J(t, x(t), u_{[0, T]}), \quad V(T, x(T)) = \ell_T(x(T)).$$

Continuous-time dynamic programming

Suppose $u^* : [0, T] \rightarrow \mathcal{U}$ is *globally optimal* for this OCP, i.e.,

$$J(0, x(0), u_{[0, T]}^*) = V(0, x(0)).$$

Then the truncation $u_{[t, T]}^* : [0, T] \rightarrow \mathcal{U}$ is *globally optimal* for the corresponding tail sub-problem, i.e.,

$$J(t, x(t), u_{[t, T]}^*) = V(t, x(t)).$$

From this, we must have the Bellman equation

$$V(t, x) = \inf_{u_{[t, t+\varepsilon]} \in \mathcal{U}} \left(\int_t^{t+\varepsilon} \ell(s, x(s), u(s)) ds + V(t + \varepsilon, x(t + \varepsilon)) \right)$$

with the boundary condition $V(T, x) = \ell_T(x)$, for all $t \in [0, T)$, $\varepsilon \in (0, T - t]$, and $x \in \mathbb{R}^n$, where $x(s)$ for $s \in [t, t + \varepsilon]$ is the state trajectory corresponding to $u_{[t, t+\varepsilon]}$ with initial condition $x(t) = x$.

That is, the Bellman equation above is a *necessary* condition for *global* optimality of u^* .

Continuous-time dynamic programming

From this, we must have the Bellman equation

$$V(t, x) = \inf_{u_{[t, t+\varepsilon]} \in \mathcal{U}} \left(\int_t^{t+\varepsilon} \ell(s, x(s), u(s)) ds + V(t + \varepsilon, x(t + \varepsilon)) \right)$$

with the boundary condition $V(T, x) = \ell_T(x)$, for all $t \in [0, T)$, $\varepsilon \in (0, T - t]$, and $x \in \mathbb{R}^n$, where $x(s)$ for $s \in [t, t + \varepsilon]$ is the state trajectory corresponding to $u_{[t, t+\varepsilon]}$ with initial condition $x(t) = x$.

Assume V is \mathcal{C}^1 -smooth with respect to t and x . Then

$$V(t + \varepsilon, x(t + \varepsilon)) = V(t, x) + \frac{\partial V}{\partial t}(t, x)\varepsilon + \nabla_x V(t, x)^\top f(t, x, u(t))\varepsilon + o(\varepsilon)$$

and

$$\int_t^{t+\varepsilon} \ell(s, x(s), u(s)) ds = \ell(t, x, u(t))\varepsilon + o(\varepsilon),$$

where $\lim_{\varepsilon \rightarrow 0} \frac{o(\varepsilon)}{\varepsilon} = 0$.

The Hamilton-Jacobi-Bellman equation

Assume V is \mathcal{C}^1 -smooth with respect to t and x . Then

$$V(t + \varepsilon, x(t + \varepsilon)) = V(t, x) + \frac{\partial V}{\partial t}(t, x)\varepsilon + \nabla_x V(t, x)^\top f(t, x, u(t))\varepsilon + o(\varepsilon)$$

and

$$\int_t^{t+\varepsilon} \ell(s, x(s), u(s)) ds = \ell(t, x, u(t))\varepsilon + o(\varepsilon),$$

where “ $o(\varepsilon)$ ” is little-o notation encapsulating terms that satisfy $\lim_{\varepsilon \rightarrow 0} \frac{o(\varepsilon)}{\varepsilon} = 0$.

Substitute these into the Bellman equation to get

$$-\frac{\partial V}{\partial t}(t, x)\varepsilon = \inf_{u_{[t, t+\varepsilon]} \in \mathcal{U}} \left(\ell(t, x, u(t))\varepsilon + \nabla_x V(t, x)^\top f(t, x, u(t))\varepsilon + o(\varepsilon) \right)$$

Divide by ε and take the limit as $\varepsilon \rightarrow 0$ to get the *Hamilton-Jacobi-Bellman (HJB) equation*

$$-\frac{\partial V}{\partial t}(t, x) = \inf_{u \in \mathcal{U}} \left(\ell(t, x, u) + \nabla_x V(t, x)^\top f(t, x, u) \right)$$

with boundary condition $V(T, x) = \ell_T(x)$, for all $t \in [0, T)$ and $x \in \mathbb{R}^n$.

The Hamilton-Jacobi-Bellman equation

The *Hamilton-Jacobi-Bellman (HJB) equation* is

$$-\frac{\partial V}{\partial t}(t, x) = \inf_{u \in \mathcal{U}} \left(\ell(t, x, u) + \nabla_x V(t, x)^\top f(t, x, u) \right)$$

with boundary condition $V(T, x) = \ell_T(x)$, for all $t \in [0, T)$ and $x \in \mathbb{R}^n$.

If we assume a globally optimal control exists, then we can replace “inf” with “min” above, and the HJB equation is a necessary condition for global optimality of $u^* : [0, T] \rightarrow \mathcal{U}$. A similar derivation to the discrete-time case can show that the HJB equation can be used to form *sufficient* optimality conditions.

Define the *Hamiltonian*

$$H(t, x, u, p) := p^\top f(t, x, u) - \ell(t, x, u).$$

Then we can rewrite the HJB equation as

$$\frac{\partial V}{\partial t}(t, x) = \sup_{u \in \mathcal{U}} H(t, x, u, -\nabla_x V(t, x)).$$

The *Hamilton-Jacobi-Bellman (HJB) equation* is

$$\frac{\partial V}{\partial t}(t, x) = \sup_{u \in \mathcal{U}} H(t, x, u, -\nabla_x V(t, x))$$

with boundary condition $V(T, x) = \ell_T(x)$, for all $t \in [0, T)$ and $x \in \mathbb{R}^n$.

In the PMP, we saw that an optimal control u^* must satisfy

$$u^*(t) = \arg \max_{u \in \mathcal{U}} H(t, x^*(t), u, p^*(t)), \quad \forall t \in [0, T].$$

This is an *open-loop* specification, since u^* depends on the state and co-state trajectories, which come from solving a BVP over the entire interval $[0, T]$.

With the HJB, we have that

$$u^*(t) = \arg \max_{u \in \mathcal{U}} H(t, x^*(t), u, -\nabla_x V(t, x^*(t))), \quad \forall t \in [0, T].$$

This is a *closed-loop* specification, since if we know $V(t, x)$ everywhere, then $u^*(t)$ is completely determined by $x^*(t)$.

In the PMP, we saw that an optimal control u^* must satisfy

$$u^*(t) = \arg \max_{u \in \mathcal{U}} H(t, x^*(t), u, p^*(t)), \quad \forall t \in [0, T].$$

This is an *open-loop* specification, since u^* depends on the state and co-state trajectories, which come from solving a BVP over the entire interval $[0, T]$.

With the HJB, we have that

$$u^*(t) = \arg \max_{u \in \mathcal{U}} H(t, x^*(t), u, -\nabla_x V(t, x^*(t))), \quad \forall t \in [0, T].$$

This is a *closed-loop* specification, since if we know $V(t, x)$ everywhere, then $u^*(t)$ is completely determined by $x^*(t)$. However, computing $V(t, x)$ everywhere is much harder to do; it is the solution of a PDE, while the PMP required us to solve a system of ODEs.

Comparing the PMP and the HJB also gives us a new interpretation of the adjoint state as a *sensitivity*

$$p^*(t) = -\nabla_x V(t, x^*(t)).$$

1. The Bellman equation as a sufficient optimality condition
2. Continuous-time dynamic programming and the HJB equation
3. LQR control in continuous-time
4. Non-smooth value functions

Consider the continuous-time OCP

$$\begin{aligned} & \underset{u}{\text{minimize}} \quad \frac{1}{2}x(t)^\top Q_T x(t) + \frac{1}{2} \int_0^T (x(t)^\top Q(t)x(t) + u(t)^\top R(t)u(t)) dt \\ & \text{subject to} \quad \dot{x}(t) = A(t)x(t) + B(t)u(t), \quad \forall t \in [0, T] \end{aligned}$$

where $Q_T \succeq 0$, $Q(t) \succeq 0$, and $R(t) \succ 0$ for all $t \in [0, T]$.

The HJB equation for this problem is

$$-\frac{\partial V}{\partial t}(t, x) = \inf_{u \in \mathbb{R}^m} \left(\frac{1}{2}x^\top Q(t)x + \frac{1}{2}u^\top R(t)u + \nabla_x V(t, x)^\top (A(t)x + B(t)u) \right)$$

with boundary condition $V(T, x) = \frac{1}{2}x^\top Q_T x$.

Take the derivative with respect to u and set it equal to zero to get

$$u = -R(t)^{-1}B(t)^\top \nabla_x V(t, x)$$

The HJB equation for this problem is

$$-\frac{\partial V}{\partial t}(t, x) = \inf_{u \in \mathbb{R}^m} \left(\frac{1}{2} x^\top Q(t) x + \frac{1}{2} u^\top R(t) u + \nabla_x V(t, x)^\top (A(t)x + B(t)u) \right)$$

with boundary condition $V(T, x) = \frac{1}{2} x^\top Q_T x$.

Take the derivative with respect to u and set it equal to zero to get

$$u = -R(t)^{-1} B(t)^\top \nabla_x V(t, x)$$

Substitute this back into the HJB equation and rearrange to get

$$\frac{\partial V}{\partial t}(t, x) = \frac{1}{2} \nabla_x V(t, x)^\top B(t) R(t)^{-1} B(t) \nabla_x V(t, x) - x^\top A(t)^\top \nabla_x V(t, x) - \frac{1}{2} x^\top Q(t) x,$$

which must hold for all (t, x) with boundary condition $V(T, x) = \frac{1}{2} x^\top Q_T x$.

Substitute this back into the HJB equation and rearrange to get

$$\frac{\partial V}{\partial t}(t, x) = \frac{1}{2} \nabla_x V(t, x)^\top B(t) R(t)^{-1} B(t) \nabla_x V(t, x) - x^\top A(t)^\top \nabla_x V(t, x) - \frac{1}{2} x^\top Q(t) x,$$

which must hold for all (t, x) with boundary condition $V(T, x) = \frac{1}{2} x^\top Q_T x$.

Based on the boundary condition, let us make the *ansatz* $V(t, x) = \frac{1}{2} x^\top P(t) x$, where $P(t)$ is symmetric positive-definite. Then the HJB equation becomes

$$\begin{aligned} \frac{1}{2} x \dot{P}(t) x &= \frac{1}{2} x^\top P(t) B(t) R(t)^{-1} B(t) P(t) x - x^\top A(t)^\top P(t) x - \frac{1}{2} x^\top Q(t) x \\ &= \frac{1}{2} x^\top \left(P(t) B(t) R(t)^{-1} B(t) P(t) - P(t) A(t) - A(t)^\top P(t) - Q(t) \right) x \end{aligned}$$

This must hold for all (t, x) , so $P(t)$ must satisfy the *continuous-time Riccati equation*

$$\dot{P}(t) = P(t) B(t) R(t)^{-1} B(t) P(t) - P(t) A(t) - A(t)^\top P(t) - Q(t),$$

which is an ODE that can be solved backwards in time from $P(T) = Q_T$.

LQR control in continuous-time

The value function is $V(t, x) = \frac{1}{2}x^\top P(t)x$, where $P(t) \succ 0$ must satisfy the *continuous-time Riccati equation*

$$\dot{P}(t) = P(t)B(t)R(t)^{-1}B(t)P(t) - P(t)A(t) - A(t)^\top P(t) - Q(t),$$

which is an ODE that can be solved backwards in time from $P(T) = Q_T$.

The optimal control is

$$u^* = -R(t)^{-1}B(t)^\top \nabla_x V(t, x) = \underbrace{-R(t)^{-1}B(t)^\top P(t)}_{=:K(t)} x,$$

which is a *linear feedback policy*.

Recall that in the discrete-time case we had to solve the *discrete-time Riccati equation*

$$P_t = Q_t + A_t^\top P_{t+1}A_t - A_t^\top P_{t+1}B_t(R_t + B_t^\top P_{t+1}B_t)^{-1}B_t^\top P_{t+1}A_t$$

recursively from the boundary condition $P_T = Q_T$. The optimal control input in this case was $u^* = K_t x$ with $K_t := -(R_t + B_t^\top P_{t+1}B_t)^{-1}B_t^\top P_{t+1}A_t$.

Agenda

1. The Bellman equation as a sufficient optimality condition
2. Continuous-time dynamic programming and the HJB equation
3. LQR control in continuous-time
4. Non-smooth value functions

Non-smooth value functions

Recall that in deriving the HJB equation we assumed that $V(t, x)$ is \mathcal{C}^1 -smooth with respect to t and x . However, this is often not true, particularly for problems with bounded control and a terminal cost.

As an example, consider the scalar problem

$$\begin{aligned} & \underset{u}{\text{minimize}} \quad x(T) \\ & \text{subject to} \quad \dot{x}(t) = x(t)u(t), \quad \forall t \in [0, T] \\ & \quad \quad \quad u(t) \in [-1, 1], \quad \forall t \in [0, T] \end{aligned}$$

By inspection,

$$u^* = \begin{cases} 1, & x < 0 \\ -1, & x > 0 \\ ?, & x = 0 \end{cases} \implies \dot{x}^* = \begin{cases} x, & x < 0 \\ -x, & x > 0 \\ 0, & x = 0 \end{cases} \implies x^*(t) = \begin{cases} e^{-(t-t_0)}x_0, & x_0 < 0 \\ e^{t-t_0}x_0, & x_0 > 0 \\ 0, & x_0 = 0 \end{cases}$$

Non-smooth value functions

As an example, consider the scalar problem

$$\begin{aligned} & \underset{u}{\text{minimize}} && x(T) \\ & \text{subject to} && \dot{x}(t) = x(t)u(t), \quad \forall t \in [0, T] \\ & && u(t) \in [-1, 1], \quad \forall t \in [0, T] \end{aligned}$$

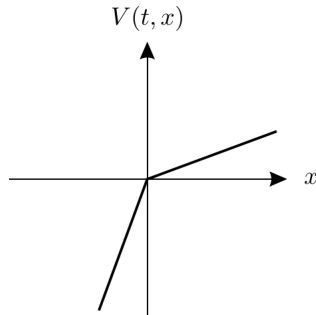
The value function is

$$V(t, x) = \begin{cases} e^{-(T-t)}x, & x < 0 \\ e^{T-t}x, & x > 0, \\ 0, & x = 0 \end{cases}$$

which is not \mathcal{C}^1 -smooth, but it does satisfy the HJB equation

$$-\frac{\partial V}{\partial t}(t, x) = \inf_{u \in [-1, 1]} \nabla_x V(t, x) x u = -|\nabla_x V(t, x) x|$$

away from $x = 0$, with boundary condition $V(T, x) = x$.



$V(t, x)$ for fixed t .

Non-smooth value functions

It turns out that the HJB equation

$$-\frac{\partial V}{\partial t}(t, x) = \inf_{u \in \mathcal{U}} \left(\ell(t, x, u) + \nabla_x V(t, x)^\top f(t, x, u) \right)$$

with boundary condition $V(T, x) = \ell_T(x)$, can also have *locally Lipschitz* solutions that can be non-smooth at points (t, x) .

In such cases, the HJB equation must be reinterpreted as

$$\begin{aligned} -\frac{\partial \bar{\varphi}}{\partial t}(t, x) - \inf_{u \in \mathcal{U}} \left(\ell(t, x, u) + \nabla_x \bar{\varphi}(t, x)^\top f(t, x, u) \right) &\leq 0 \\ -\frac{\partial \underline{\varphi}}{\partial t}(t, x) - \inf_{u \in \mathcal{U}} \left(\ell(t, x, u) + \nabla_x \underline{\varphi}(t, x)^\top f(t, x, u) \right) &\geq 0 \end{aligned}$$

for all \mathcal{C}^1 -smooth test functions $\bar{\varphi}$ and $\underline{\varphi}$ such that $\bar{\varphi} - V$ has a local minimum at (t, x) and $\underline{\varphi} - V$ has a local maximum at (t, x) , for all (t, x) .

In this case, V is known as a *viscosity solution* to the HJB. More details can be found in (Liberzon, 2012, §5.3).

System identification and adaptive control

D. Liberzon. *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton University Press, 2012.