Reinforcement Learning in Continuous Time

$\dot{x} = f(x, u)$ $\quad x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ — system

$J_\infty(x, u(\cdot)) := \int_0^\infty e^{-\rho t} q(x(t), u(t))\, dt$ — cost

($\rho > 0$: discount rate)

Bellman (value fcn):

$V(x) := \min_{u(\cdot)} J_\infty(x, u(\cdot))$

HJB: $\rho V(x) = \min_{u \in \mathbb{R}^m} \left\{ q(x,u) + \frac{\partial V}{\partial x}(x) f(x,u) \right\}$

optimal control: $k(x) = \operatorname{argmin}_{u \in \mathbb{R}^m} \left\{ q(x,u) + \frac{\partial V}{\partial x}(x) f(x,u) \right\}$

(assuming existence/uniqueness, etc.)

RL: a set of techniques for approximately and asymptotically approaching optimal performance (as encoded in $V(\cdot)$ and $k(\cdot)$) w/o precise knowledge of system dynamics and/or cost structure.

Two components (primitives) in RL:

1) Given a current candidate optimal control generate a candidate value fcn. [value update]

2) Given a current candidate value fcn, generate a candidate optimal control. [policy update]

Value update

Modified cost-to-go:

$\tilde{J}_\infty(t, x; u(\cdot)) := \int_t^\infty e^{\rho(t-s)} q(x(s), u(s))\, ds$

s.t. $\dot{x}(s) = f(x(s), u(s))$, $s \geq t$

$x(t) = x$

Fix $u(\cdot)$, consider $(x(t), u(t))_{t \geq 0}$ as the corresp. trajectory of state/action pairs.

$$\frac{d}{dt} \tilde{J}_\infty(t, x(t); u(\cdot)) = \rho \tilde{J}_\infty(t, x(t); u(\cdot)) + e^{\rho t} \frac{d}{dt} \int_t^\infty e^{-\rho s} q(x(s), u(s))\, ds$$

$$= \rho \tilde{J}_\infty(t, x(t); u(\cdot)) + e^{\rho t}\left(-e^{-\rho t} q(x(t), u(t))\right)$$

Self-consistency condition: for any admissible control $u(\cdot)$,

$$\frac{d}{dt} \tilde{J}_\infty(t, x(t); u(\cdot)) = \rho \tilde{J}_\infty(t, x(t); u(\cdot)) - q(x(t), u(t))$$

In particular, suppose $\bar{u}(\cdot)$ is optimal, so

$$\tilde{J}_\infty(t, x(t); \bar{u}(\cdot)) = V(x(t)), \qquad x(0) = x$$

where $\dot{x}(t) = f(x(t), \bar{u}(t))$, $x(0) = x$

$$\Rightarrow \boxed{\frac{d}{dt} V(x(t)) = \rho V(x(t)) - q(x(t), \bar{u}(t))}$$

Let $\bar{u}(\cdot)$ be given; then a candidate value fcn $\hat{V}(\cdot)$ has to satisfy

$$\frac{d}{dt} \hat{V}(x(t)) = \rho \hat{V}(x(t)) - q(x(t), \bar{u}(t)),$$

o/w $\hat{V}(\cdot)$ is falsified.

Parametric function class: $V(x; \theta)$ where $x \in \mathbb{R}^n$ (state) and $\theta \in \mathbb{R}^k$ (parameters that can be tuned)

- choose $\{V(\cdot;\theta) : \theta \in \mathbb{R}^k\}$ w/ hope that V can be approximated by an element of this class to any desired accuracy.

Value update: choose some $\theta(0) \in \mathbb{R}^k$, come up w/ update dynamics $\dot\theta(t)$ to decrease the temporal difference (TD)

$$e_t(\theta) := q(x(t), u(t)) - \rho V(x(t);\theta) + \frac{d}{dt} V(x(t);\theta)$$

where $(x(t), u(t))_{t\geq 0}$ is the trajectory induced by a given candidate control $u(\cdot)$.

$$E_t(\theta) := \frac{1}{2} |e_t(\theta)|^2$$

TD-learning: choose $\dot\theta(t)$ to decrease $E_t(\theta)$

$$\dot\theta(t) := -\gamma \nabla_\theta E_t(\theta(t)) \qquad (\gamma > 0: \text{learning rate})$$

i.e., for each $i \in \{1, \dots, k\}$,

$$\dot\theta_i(t) = -\gamma \frac{\partial}{\partial \theta_i} E_t(\theta(t))$$

$$\frac{\partial}{\partial \theta_i} E_t(\theta) = \frac{1}{2} \frac{\partial}{\partial \theta_i} |e_t(\theta)|^2$$

$$= e_t(\theta) \frac{\partial}{\partial \theta_i} e_t(\theta)$$

$$= e_t(\theta) \cdot \frac{\partial}{\partial \theta_i} \left\{ q(x(t), u(t)) - \rho V(x(t);\theta) + \frac{d}{dt} V(x(t);\theta) \right\}$$

where

$$\frac{d}{dt} V(x(t);\theta) = \frac{\partial}{\partial x} V(x(t);\theta)\, \dot x(t)$$

$$= \frac{\partial}{\partial x} V(x(t);\theta) f(x(t), u(t))$$

"Ideal" TD-learning:

$$\dot\theta_i(t) = \gamma e_t(\theta(t)) \left\{ \rho \frac{\partial}{\partial \theta_i} V(x(t);\theta(t)) - \frac{\partial}{\partial \theta_i} \left(\frac{\partial}{\partial x} V(x(t);\theta(t)) \right) \dot x(t) \right\}$$

- this requires knowledge of $f(\cdot,\cdot)$ and $q(\cdot,\cdot)$.

Approximation: $\frac{d}{dt} V(x(t);\theta)$

- forward Euler: $\frac{d}{dt} V(x(t);\theta) \approx \frac{V(x(t+h);\theta) - V(x(t);\theta)}{h}$

- backward Euler: $\frac{d}{dt} V(x(t);\theta) \approx \frac{V(x(t);\theta) - V(x(t-h);\theta)}{h}$

($h > 0$ small and fixed)

Forward Euler is not implementable online, so we will use backward Euler:

$$e_t(\theta) \approx e_t^h(\theta) := q(x(t),u(t)) - \rho V(x(t);\theta) + \frac{1}{h}\{V(x(t);\theta) - V(x(t-h);\theta)\}$$

$$= q(x(t),u(t)) + \frac{1}{h}\{(1-\rho h)V(x(t);\theta) - V(x(t-h);\theta)\}$$

Backward Euler TD-learning:

$$\dot{\theta}(t) = -\gamma e_t^h(\theta(t)) \nabla_\theta e_t^h(\theta(t)) \qquad [TD(0)]$$

TD(λ) modification: for $i \in \{1,\dots,k\}$

$$\dot{\theta}_i(t) = \gamma e_t(\theta(t)) \xi_i(t)$$

$$\dot{\xi}_i(t) = -\lambda \xi_i(t) + \frac{\partial}{\partial \theta_i} V(x(t);\theta(t))$$

($\lambda > 0$: clamping rate)

- Jacobians of $V(x;\theta)$ w.r.t. x and θ can be computed using backpropagation techniques (next lecture?)

(2) Policy update

Let $V(x)$ be the "true" value function. Then $k: \mathbb{R}^n \to \mathbb{R}^m$ is optimal if

$$\rho V(x) = q(x, k(x)) + \frac{\partial}{\partial x} V(x) f(x, k(x))$$
$$= \min_{u \in \mathbb{R}^m} \left\{ q(x,u) + \frac{\partial}{\partial x} V(x) f(x,u) \right\}$$

- if $\hat{V}$ is a candidate value fcn, then we can construct state feedback law $\hat{k}(\cdot)$:

$$\hat{k}(x) = \operatorname*{argmin}_{u \in \mathbb{R}^m} \left\{ q(x,u) + \frac{\partial}{\partial x} \hat{V}(x) f(x,u) \right\}.$$

[Aside: the value function is a solution of the fixed-point problem

$$\rho V = TV,$$

where, for any C^1 $W: \mathbb{R}^n \to \mathbb{R}$,

$$(TW)(x) := \min_{u \in \mathbb{R}^m} \left\{ q(x,u) + \frac{\partial}{\partial x} W(x) f(x,u) \right\}]$$

- again, needs knowledge of q and f.

Example (only partial knowledge of q, f)

$$\dot{x} = f(x) + \sum_{i=1}^{m} u_i g_i(x) \qquad \text{(system)}$$

$$q(x,u) = u^T R(x) u + Q(x) \qquad \text{(cost)}$$

Assume: $g_1, \ldots, g_m$, $R(x)$ are known
$R(x) = R(x)^T > 0$ for each $x \in \mathbb{R}^n$

Then we can compute $\hat{k}(x)$ explicitly for any C^1 $\hat{V}: \mathbb{R}^n \to \mathbb{R}$ (candidate value fcn)

$$T\hat{V}(x) = \min_{u \in \mathbb{R}^m} \left\{ q(x,u) + \frac{\partial}{\partial x} \hat{V}(x) f(x,u) \right\}$$

$$= \min_{u \in \mathbb{R}^m} \left\{ Q(x) + u^T R(x) u + \frac{\partial}{\partial x} \hat{V}(x) f(x) + \sum_{i=1}^m u_i \frac{\partial}{\partial x} \hat{V}(x) g_i(x) \right\}$$

$$= Q(x) + L_f \hat{V}(x) + \min_{u \in \mathbb{R}^m} \left\{ u^T R(x) u + L_G \hat{V}(x)^T u \right\}$$

where $L_f \hat{V}(x) := \frac{\partial}{\partial x} \hat{V}(x) f(x)$

$L_G \hat{V}(x) := \left(\frac{\partial}{\partial x} \hat{V}(x) g_1(x), \ldots, \frac{\partial}{\partial x} \hat{V}(x) g_m(x) \right)^T$

are the Lie derivatives of $\hat{V}(\cdot)$ along $f, g_1, \ldots, g_m$

$u \mapsto u^T R(x) u + L_G \hat{V}(x)^T u$ is strongly cvx, so the minimizing u is unique:

$$\hat{k}(x) = -\frac{1}{2} R(x)^{-1} L_G \hat{V}(x),$$

and thus

$$T\hat{V}(x) = Q(x) + L_f \hat{V}(x) - \frac{1}{4} L_G \hat{V}(x)^T R(x)^{-1} L_G \hat{V}(x).$$

Again, Jacobians can be computed efficiently using backpropagation; in fact,

$$L_G \hat{V}(x) = \left(\frac{\partial}{\partial x} \hat{V}(x) g_1(x), \ldots, \frac{\partial}{\partial x} V(x) g_m(x) \right)^T,$$

each coordinate is a Jacobian-vector product, can be computed efficiently.