Spring 2019

Stanford
AA 203: Introduction to Optimal Control and Dynamic Optimization
Problem set 3, due on April 24

Problem 1: Suppose we have a machine that is either running or is broken down. If it runs throughout one week, it makes a gross profit of \$100. If it fails during the week, gross profit is zero. If it is running at the start of the week and we perform preventive maintenance, the probability that it will fail during the week is 0.4. If we do not perform such maintenance, the probability of failure is 0.7. However, maintenance will cost \$20. When the machine is broken down at the start of the week, it may either be repaired at a cost of \$40, in which case it will fail during the week with a probability of 0.4, or it may be replaced at a cost of \$150 by a new machine that is guaranteed to run through its first week of operation. Find the optimal repair, replacement, and maintenance policy that maximizes total profit over four weeks, assuming a new machine at the start of the first week.

Problem 2: Consider the scalar system

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

with the constraint $|u(t)| \leq 1$ for all $t \in [0, T]$. The cost function is

$$J = \Big(x(T)\Big)^2 + \int_0^T \Big(u(t)\Big)^2 \, dt.$$

Find an optimal control policy for this problem by using the HJB equation.

Hint: Consider the following cost-to-go function candidate:

$$J^*(t,x) = \begin{cases} (x - T + t)^2 + T - t & \text{if } x > 1 + T - t, \\ (x + T - t)^2 + T - t & \text{if } x < -(1 + T - t), \\ x^2/(1 + T - t) & \text{if } |x| \leq 1 + T - t, \end{cases}$$

and then verify that it is indeed a solution to the HJB equation.

Problem 3: Consider the continuous LQR problem:

$$\dot{x}(t) = \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(t),$$

with cost function:

$$J = \frac{1}{2}\left\{ x^T(t_f)\begin{pmatrix} 0 & 0 \\ 0 & h \end{pmatrix} x(t_f) + \int_0^{t_f} \left[x^T(t)\begin{pmatrix} q & 0 \\ 0 & 0 \end{pmatrix} x(t) + r\,u^2(t) \right] dt \right\},$$

where $q = 1$, $r = 3$, $h = 4$, $t_f = 10$. Write a Matlab script to solve the corresponding Riccati differential equation. Plot the time-varying gains and the state solution with initial condition $x(0) = [1, 1]^T$. Repeat the problem with $t_f = 100$ and compare the results with those for the case $t_f = 10$.

Problem 4: Consider a *one-dimensional* dynamical model of a space launcher, inspired by the classic Goddard model widely used in the early days of rocket science,

$$\begin{cases} \dot{h} = v \\ \dot{v} = \dfrac{u}{m} - g + d \\ \dot{m} = -bu \end{cases}$$

where:

- the state (h, v, m) comprises rescaled height h (i.e., height divided by a rescaling, non-dimensional constant C used to ease the discretization procedure), rescaled vertical velocity v (where we use the same rescaling constant C), and propellant mass m;
- the launcher gains height via bounded rescaled thrust u, specifically $0 \leq u \leq \frac{10000}{C}$ kg m/s^2, against a constant rescaled gravity $g = \frac{9.81}{C}$ m/s^2, and a bounded, rescaled disturbance $|d| \leq \frac{1}{C}$ m/s^2 (due to e.g. inaccurate aerodynamic coefficients, wind disturbances, etc.);
- the fuel consumption is at rate bu, where $b = C \cdot 10^{-4}$ s/m;
- the rescaling constant is set to $C = 100000$.

The initial condition of the launcher is $(h, v, m) = (0, 0, 500)$, and the objective is to reach height $h = \frac{150000}{C}$ m, with velocity $v = \frac{28}{C}$ m/s, with an acceptable tolerance of $\frac{500}{C}$ m in distance and $\frac{2.8}{C}$ m/s in velocity. For safety reasons, the launcher should have enough fuel left for further maneuvers, specifically, the final mass of propellant should not be lower than 250 kg.

1. Write down the target set $\mathcal{T}$ representing the acceptable set of states for reaching the desired final condition within the tolerances, and find a suitable function $\ell(h, v, m)$ such that $\ell(h, v, m) \leq 0 \iff (h, v, m) \in \mathcal{T}$.
2. The value function $V(h, v, m, t)$ representing the set of states from which the rocket can reach the target set within t seconds is the solution to the HJI PDE equation

$$\frac{\partial V}{\partial t}(h, v, m, t) + \min_{u \in \mathcal{U}} \max_{d \in \mathcal{D}} \nabla V(h, v, m, t)' f(h, v, m, u, d) = 0$$

where $f(h, v, m, u, d)$ are the system dynamics, and $\mathcal{U}$ and $\mathcal{D}$ represent the bounds on control and disturbance, respectively. Determine the optimal control u and the worst-case disturbance d that optimize the HJI equation. That, is find an analytic expression for u^* and d^*, where

$$u^*(h, v, m, t) = \arg\min_{u \in \mathcal{U}} \max_{d \in \mathcal{D}} \nabla V(h, v, m, t)' f(h, v, m, u, d)$$

$$d^*(h, v, m, t) = \arg\max_{d \in \mathcal{D}} \nabla V(h, v, m, t)' f(h, v, m, u^*, d)$$

(*Hint: as shown in class for a similar problem,* $\nabla V(h, v, m, t)' f(h, v, m, u, d)$ *results in an expression where* u *and* d *are linear terms, which can be maximized/minimized independently.*)

3. Compute $V(h, v, m, t)$ from $t = -6$ to $t = 0$ using the `helperOC` toolbox, which can be found at `https://github.com/HJReachability/helperOC`.

   (*Hints: use as a starting point the example code shown in class* ***`DubinsAA203.m`***. *Place the folder* ***`@Goddard`*** *(containing the system dynamics) in* ***`helperOC-master/dynSys`***. *Use the following recommended grid bounds and resolution:* $h \in \left[0, \frac{150500}{C}\right]$ *m with 20 grid points,* $v \in \left[0, \frac{30}{C}\right]$ *m/s with 20 grid points,* $m \in [250, 500]$ *kg with 50 grid points.)*

4. By using `visSetIm`, plot $V(t = -4.5, h, v, m)$ and $V(t = -5.5, h, v, m)$.

5. (Optional, no credits given for this question) Roughly, starting from which time t one has feasible values for the mass m (i.e., in $[250, 500]$ kg) that allow the rocket to exactly reach the desired final state $(h, v) = \left(\frac{150000}{C}, \frac{28}{C}\right)$?

Learning goals for this problem set:

Problem 1: To gain a basic understanding of dynamic programming for stochastic optimal control, and exercise the "mechanics" of dynamic programming.

Problem 2: To learn how to solve optimal control problems by solving the HJB equation.

Problem 3: To gain insights into the continuous-time LQR problem.

Problem 4: To gain a fundamental understanding of reachability analysis, and learn how to use computational tools to solve the HJI equation.