Spring 2019

Stanford
AA 203: Introduction to Optimal Control and Dynamic Optimization
Problem set 2, due on April 17

Problem 1: Consider the shortest path problem in Figure 1, where it is only possible to travel to the right and the numbers represent the travel times for each leg. The control is the decision to go up-right or down-right at each node.

(a) By using Dynamic Programming (DP), find the shortest path from A to B.

(b) Consider a generalized version of the shortest path problem in Figure 1 where the grid has n segments on each side. Find the number of computations required by an exhaustive search algorithm (i.e., the number of routes that such algorithm would need to evaluate) and the number of computations required by a DP algorithm (i.e., the number of DP evaluations). (For example, for the case where $n = 3$, the number of computations for the exhaustive search algorithm is 20 and the number of computations for the DP algorithm is 15.)

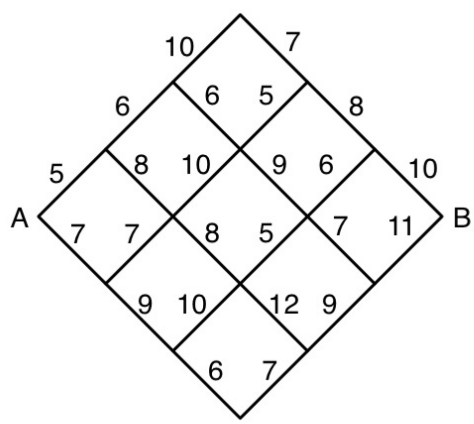

Figure 1: Shortest path problem for Problem 1.

Problem 2: Consider the two-dimensional system

$$\dot{x}_1(t) = u_1(t), \qquad \dot{x}_2(t) = u_2(t),$$

with the control constraint $\|u(t)\| = 1$. We want to find a state trajectory that starts at a given point $x(0)$, ends at another given point $x(T)$, and minimizes

$$\int_0^T r(x(t))\, dt.$$

The function $r(t)$ is nonnegative and continuous, and the final time T is subject to optimization. Suppose we discretize the plane with a mesh of size Δ that passes through $x(0)$ and $x(T)$, and we introduce a shortest path problem of going from $x(0)$ to $x(T)$ using moves of the following type: from each mesh point $\bar{x} = (\bar{x}_1, \bar{x}_2)$ we can go to each of the mesh points $(\bar{x}_1 + \Delta, \bar{x}_2)$, $(\bar{x}_1 - \Delta, \bar{x}_2)$, $(\bar{x}_1, \bar{x}_2 + \Delta)$, and $(\bar{x}_1, \bar{x}_2 - \Delta)$, at a cost $r(\bar{x})\Delta$. Show by example that this is a bad discretization of the original problem in the sense that the shortest distance need not approach the optimal cost of the original problem as $\Delta \to 0$.

Problem 3: Consider the problem of inscribing an N-side polygon in a given circle, so that the polygon has maximal perimeter.

(a) Formulate the problem as a DP problem involving sequential placement of N points in the circle.

(b) Use DP to show that the optimal polygon is regular (all sides are equal).

Problem 4: Currently being finalized, to be released 04/11.

Problem 5: Currently being finalized, to be released 04/11.

Learning goals for this problem set:

Problem 1: To familiarize with the DP algorithm and to appreciate the computational savings of DP versus an exhaustive search algorithm.

Problem 2: To gain insights into the delicate issue of discretization.

Problem 3: To familiarize with the process of casting an optimal control problem into a form amenable to a DP solution.