Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute for Dynamic Systems and Control
IDSC
Institut für Dynamische Systeme und Regelungstechnik

151-0563-00 **Dynamic Programming and Optimal Control** (Fall 2009)

Programming Exercise #2 Topic: The Stochastic Shortest Path Problem

Issued: Nov 11, 2009 Due: Nov 25, 2009

Angela Schöllig (aschoellig@ethz.ch), 11. November 2009

Policy Iteration, Value Iteration and Linear Programming

– Find the optimal policy and the associated optimal cost for each starting node of the given stochastic shortest path problem –

The stochastic shortest path problem is to be solved for a finite graph with $(N+1)$ nodes, where the $(N+1)$th node represents the termination state t. The dynamics of the system can be controlled by an input u taking values in the finite set $U = \{1, 2, \ldots, M\}$. At a node i, the use of the input u specifies the probability $p_{ij}(u)$ of transitioning to state j. For each input u, we define a transition probability matrix $P(u)$ of dimension $(N+1)$ by $(N+1)$ with $p_{ij}(u)$ as its (i,j)-th element.

The goal is to minimize the expected number of steps to get to the termination state for any given starting node $i_0 \in \{1, 2, \ldots, N, t\}$, i.e. to find the minimum expected number of transitions for each i_0 and the associated optimal policy $u^* = \mu^*(i)$ at each node i. Three different algorithms are to be implemented to solve this problem:

(a) Use policy iteration first.

(b) Apply value iteration. The result of (a) might be used to provide a meaningful stopping criterion for the value iteration algorithm.

(c) Define a linear programming problem solving the stated stochastic shortest path problem.

Hint: Use the `linprog` function of MATLAB's Optimization Toolbox to solve the linear program.

Compare the three different methods and verify the result by simulation.

(d) Simulate the system for a given starting node.

Hint: Assuming that you are currently at node i and you apply the optimal policy $u^* = \mu^*(i)$, the following state can be obtained as follows: First, calculate the cumulative distribution $F_i(K,u) = \sum_{j=1}^{K} p_{ij}(u)$ for all $K \in \{1, 2, \ldots, N, t\}$. $F_i(K,u)$ represents the probability of transitioning to a node $j \leq K$. Use MATLAB's `rand` function to generate a random number r drawn from the uniform distribution on the interval $(0,1)$. Then, the next state is given by the smallest K satisfying $F_i(K,u) \geq r$.

Provided Files

A set of MATLAB files is provided on the class website. Please use them for solving the above problem.

`script.m`	MATLAB script that can be used to load the problem data, execute the three different algorithms, run the simulation, and display the results.

`ssp_PolicyIt.m`	MATLAB function template to be used for your implementation of the policy iteration algorithm for the stochastic shortest path problem.
`ssp_ValueIt.m`	MATLAB function template to be used for your implementation of the value iteration algorithm for the stochastic shortest path problem.
`ssp_LinearProg.m`	MATLAB function template to be used for your implementation of the linear programming solution for the stochastic shortest path problem.
`ssp_Sim.m`	MATLAB function template to be used for your implementation of the simulation for the stochastic shortest path problem.
`generateProblemData.m`	MATLAB script to generate random problem matrices $P(u)$ for a given number of inputs M and nodes N. Here, states that are "close" to each other have higher transition probabilities than ones that are not. We use "close" to simply be the numerical difference of the states, with wrap-around. That is, it is very likely to go from node 1 to 2 and also likely to transition from node 1 to node 5 in case $(N+1)=5$.
`exampleProblemData.mat`	Problem matrices $P(u)$ specifying an example problem.

Tasks

Implement your solutions for problem (a), (b), and (c) in the files `ssp_PolicyIt.m`, `ssp_ValueIt.m`, and `ssp_LinearProg.m`, respectively. Write an appropriate simulation function as described in (d). Your code has to be able to be run with the MATLAB script `script.m`. For your code development, you may find it helpful to consider the example `exampleProblemData.mat`. Check for consistency between the results obtained from the different solution methods.

For evaluating your solution, we will test it on the given problem `exampleProblemData.mat` as well as on other random examples generated with `generateProblemData.m`.

Deliverables

Please hand in by e-mail

- your implementation of the three algorithms `ssp_PolicyIt.m`, `ssp_ValueIt.m`, and `ssp_LinearProg.m`;
- your implementation of the simulation of the system `ssp_Sim.m`;
- in a `pdf`-file, answers to the following questions
  1. How can the obtained optimal costs for the problem given in `exampleProblemData.mat` be interpreted? Include the resulting figures.
  2. What are the characteristics of the different solution methods: policy iteration, value iteration, and linear programming? Advantages and disadvantages?

Please include all five files into one `zip`-file, which you name `DPOCEx2_Names.zip`, where *Names* is a list of the surnames of all students who have worked on the solution.[1]

Send your file to Angela (`aschoellig@ethz.ch`) until the due date indicated above. We will send a confirmation e-mail upon receiving your e-mail. You are ultimately responsible that we receive your solution in time.

[1] Up to three students are allowed to work together on the problem. They will all receive the same grade.