Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

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

Dynamic Programming and Optimal Control

Fall 2009

Problem Set:
Deterministic Systems and the Shortest Path Problem

Notes:

- Problem marked with BERTSEKAS are taken from the book *Dynamic Programming and Optimal Control by Dimitri P. Bertsekas, Vol. I, 3rd edition, 2005, 558 pages, hardcover.*
- The solutions were derived by the teaching assistants in the previous class. Please report any error that you may find to `strimpe@ethz.ch` or `aschoellig@ethz.ch`.

Problem Set

Problem 1 (BERTSEKAS, p. 98, exercise 2.1)

Find a shortest path from each node to node 6 for the graph of Fig. 1 by using the DP algorithm.

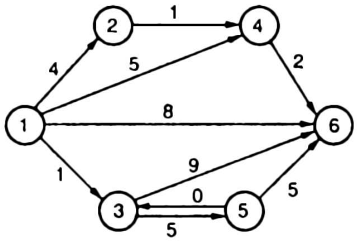

Figure 1: The arc lenghts are shown next to the arcs.

Problem 2 (BERTSEKAS, p. 98, exercise 2.2)

Find a shortest path from node 1 to node 5 for the graph of Fig. 2 by using the label correcting method of Section 2.3.1 (see BERTSEKAS).

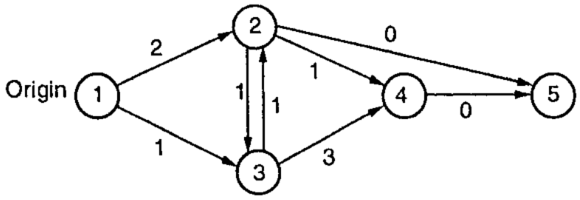

Figure 2: The arc lenghts are shown next to the arcs.

Problem 3 (BERTSEKAS, p. 103, exercise 2.14)

Consider the shortest path problem of Section 2.3 (see BERTSEKAS), except that the number of nodes in the graphs may be countably infinite (although the number of outgoing arcs from each node is finite). We assume that the length of each arc is a positive integer. Furthermore, there is at least one path from the origin node s to the destination node t. Consider the label correcting algorithm as stated and initialized in Section 2.3.1, except that UPPER is initially set to some integer that is an upper bound to the shortest distance from s to t. Show that the algorithm will terminate in a finite number of steps with UPPER equal to the shortest distance from s to t. *Hint*: Show that there is a finite number of nodes whose shortest distance from s does not exceed the initial value of UPPER.

Sample Solutions

Problem 1 (Solution)

We use the definitions and derivation on pages 67-68 (see BERTSEKAS):

- set of nodes $S = \{1, 2, 3, 4, 5\}\,, \quad N = 5$
- destination node t: node 6

Start DP Algorithm

- $J_{N-1}(i) = \alpha_{it}$

  That is, only one move to the end,

$$\begin{aligned} J_4(1) &= 8 \\ J_4(2) &= \infty \\ J_4(3) &= 9 \\ J_4(4) &= 2 \\ J_4(5) &= 5 \end{aligned}$$

- $J_k(i) = \min\limits_{j \in \{1,2,\ldots,5\}} \big(\alpha_{ij} + J_{k+1}(j)\big), \quad k = 0, 1, \ldots, N-2$

  For $k = 3$, i.e. two moves to the end,

$$\begin{aligned} J_3(1) &= \min\big(\alpha_{11} + J_4(1), \alpha_{12} + J_4(2), \alpha_{13} + J_4(3), \alpha_{14} + J_4(4), \alpha_{15} + J_4(5)\big) \\ &= 7 \quad \textit{(path 1-4-6)} \end{aligned}$$

  Analogously,

$$\begin{aligned} J_3(2) &= \min\big(\alpha_{21} + J_4(1), \alpha_{22} + J_4(2), \alpha_{23} + J_4(3), \alpha_{24} + J_4(4), \alpha_{25} + J_4(5)\big) \\ &= 3 \quad \textit{(path 2-4-6)}. \end{aligned}$$

  By directly omitting paths with cost infinity, i.e. only considering paths/ways which exist in Fig. 1, we get

$$\begin{aligned} J_3(3) &= \min\big(\alpha_{33} + J_4(3), \alpha_{35} + J_4(5)\big) \\ &= 9 \quad \textit{(path 3-6)} \end{aligned}$$

$$\begin{aligned} J_3(4) &= \min\big(\alpha_{44} + J_4(4)\big) \\ &= 2 \quad \textit{(path 4-6)} \end{aligned}$$

$$\begin{aligned} J_3(5) &= \min\big(\alpha_{53} + J_4(3), \alpha_{55} + J_4(5)\big) \\ &= 5 \quad \textit{(path 5-6)} \end{aligned}$$

Then, for $k = 2$ and three moves to the end,

$$\begin{aligned} J_2(1) &= \min\big(\alpha_{11} + J_3(1), \alpha_{12} + J_3(2), \alpha_{13} + J_3(3), \alpha_{14} + J_3(4)\big) \\ &= 7 \quad \textit{(path 1-4-6, 1-2-4-6, 1-4-6)} \end{aligned}$$

$$\begin{aligned} J_2(2) &= \min\big(\alpha_{22} + J_3(2), \alpha_{24} + J_3(4)\big) \\ &= 3 \quad \textit{(path 2-4-6, 2-4-6)} \end{aligned}$$

$$\begin{aligned} J_2(3) &= \min\big(\alpha_{33} + J_3(3), \alpha_{35} + J_3(5)\big) \\ &= 9 \quad \textit{(path 3-6)} \end{aligned}$$

$$\begin{aligned} J_2(4) &= \min\big(\alpha_{44} + J_3(4)\big) \\ &= 2 \quad \textit{(path 4-6)} \end{aligned}$$

Since $J_2(i) = J_3(i) \quad \forall i \in S$, the algorithm is terminated.

Problem 2 (Solution)

Note that, considering the Label Correcting Method, there are different methods for selecting the node i to be removed from the open bin at each iteration.

- In general, removing a node i with small (recent) arrival distance d_i is more successful, since this might result in a small $d_j = d_i + \alpha_{ij}$ and, finally, in a low distance/cost for the path $s \to t: \; d_t$
- Having a small d_t, many paths can be neglected because of $d_i > d_t$ or $d_{i,new} > d_{i,old}$

As in the lecture's example, we use depth-first search (see p.85). However, since nodes have more than one incoming arrow, we have to keep track of not only d_i, the cost associated with node i, but also the *parent node* of node i, called P_i.

It #	Remove	Open Bin	$d_t = d_5$
0	-	1	∞
1	1	2 $(d_2 = 2, P_2 = 1)$, 3 $(d_3 = 1, P_3 = 1)$	∞
2	3 $(d_3 = 1, P_3 = 1)$	2 $(d_2 = 2, P_2 = 1)$[1]; 4 $(d_4 = 4, P_4 = 3)$	∞
3	4 $(d_4 = 4, P_4 = 3)$	2 $(d_2 = 2, P_2 = 1)$	4 $(P_5 = 4)$
4	2 $(d_2 = 2, P_2 = 1)$	4 $(d_4 = 3, P_4 = 2)$[2]	2 $(P_5 = 2)$
5	4 $(d_4 = 3, P_4 = 2)$	-	2 $(P_5 = 2)$

[1] Unchanged since $1 \to 2$ and $1 \to 3 \to 2$ have same cost.
[2] Node 3 is not added since cost $d_2 + \alpha_{23} > d_3 = 1$.

$\Rightarrow$ Optimal path: $1 \to 2 \to 5, \quad d_5 = 2$

Problem 3 (Solution)

Given

1. Number of nodes countably infinite

2. $\alpha_{ij} \geq 1,\ \alpha_{ij} \in \mathbb{N},\ \forall i, j \in S$ (arc length)

3. Number of outgoing arcs from each node is *finite*

4. $\exists$ path $s \to t$

5. shortest distance between s and t, d_t^*, is bounded by d_t
   $d_t^* \leq d_{t,max}\ \in \mathbb{N},\ d_{t,max} < \infty$

Note that, with 4. and 5., we know that there exist a path $s \to t$ consisting of a finite number of nodes since $\alpha_{ij} \geq 1$ (assumption 2.). The maximum number of arcs between s and t is $d_{t,max}$!

Define a set $R = U_{i=1}^{d_{t,max}}\ S_i$, where S_i is the set of all nodes k with minimum number of arcs between s and k is i.

Way of Proceeding

A) Show that R is a finite set of nodes
B) Show that nodes $i \notin R$ will never enter the open bin

Only the set R has to be taken into account. R is finite and, with 4., we know that there exist at least one path $s \to t$.
By Proposition 2.3.1 (p. 82), the label correcting algorithm terminates with d_t = shortest distance from origin s to destination t in a finite number of steps. *q.e.d*

To be done: steps A) and B)

A) Proof by Induction

- start:
  $S_1 = \{i | i \text{ is child of } S\}$, finite because of 3.

- hypothesis:
  S_k is finite (*)

- proof:
  $S_{k+1} \leq \{i | i \text{ is child of a node in } S_k\}$, where S_k is finite (*) and each node $j \in S_k$ has only a finite number of childs, see 3.
  $\to\ S_{k+1}$ is finite.

$$\Rightarrow \quad R = \bigcup_{i=1}^{d_{t,max}} S_i \text{ is finite, since } d_{t,max} < \infty, \text{ see 5.}$$

B)

For $i \notin R$, minimum number of arcs between s and i is larger than $d_{t,max}$. With 2. all paths from s to i have length greater than $d_{t,max}$ and, therefore, i will never enter the open bin.