A Gentle Introduction to

Dynamic Programming and the Viterbi Algorithm

Prof. Hubert Kaeslin
Microelectronics Design Center
ETH Zürich

Background Material for Morgan Kaufmann "Top-Down Digital VLSI Design"
ISBN 978-0-12-800730-3 http://store.elsevier.com/9780128007303

last update: July 18, 2014

Quiz: Bridge traversal I

Problem: Four persons want to traverse a suspension bridge at night.

- The bridge can carry no more than two persons at a time.
- The four persons take 5, 10, 20 and 25 min respectively to traverse.
- Each party must carry a torch while traversing the bridge.
- The torch lasts no longer than 60 minutes.

◦ Can you help these people?

◦ How did you arrive at your plan?

◦ Can you formulate a general policy (i.e. an algorithm)?

Quiz: Bridge traversal II

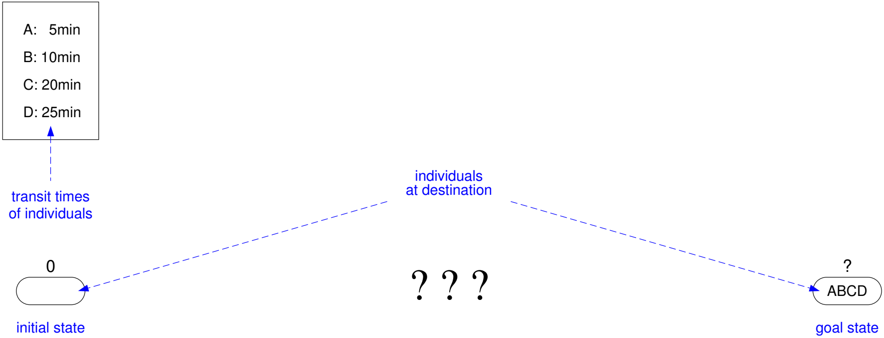


Figure: The problem stated graphically.

Quiz: Bridge traversal III

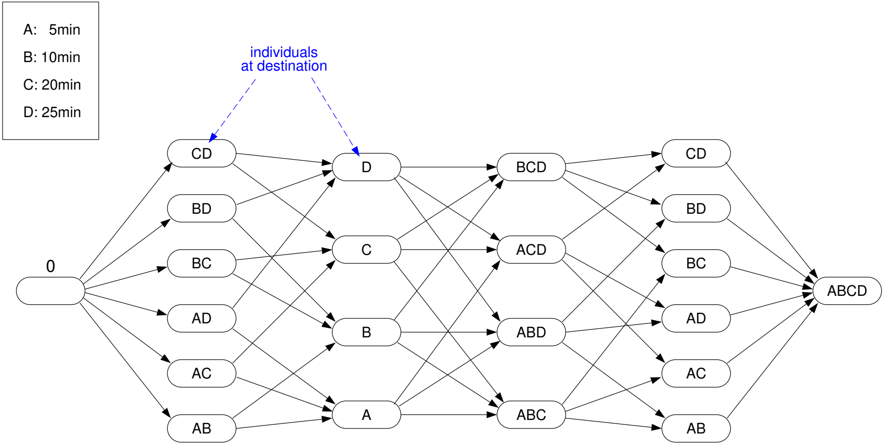


Figure: The solution space drawn as a trellis graph.

Quiz: Bridge traversal IV

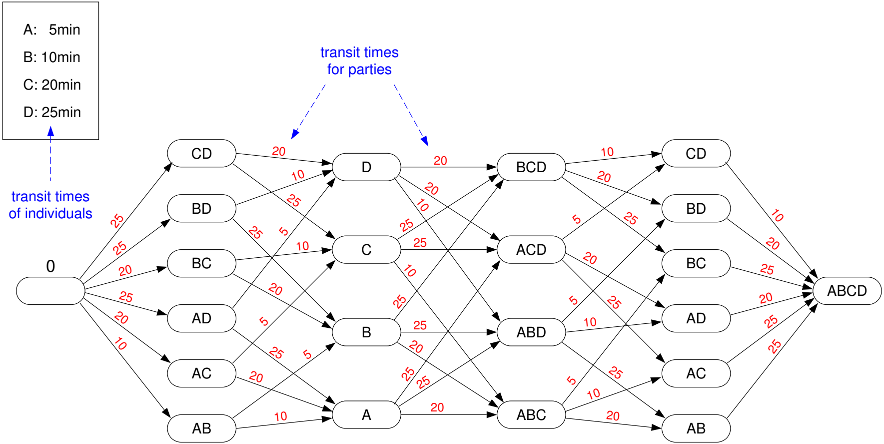


Figure: ... with branch metrics assigned to all state transitions (edges).

Quiz: Bridge traversal V

A: 5min
B: 10min
C: 20min
D: 25min

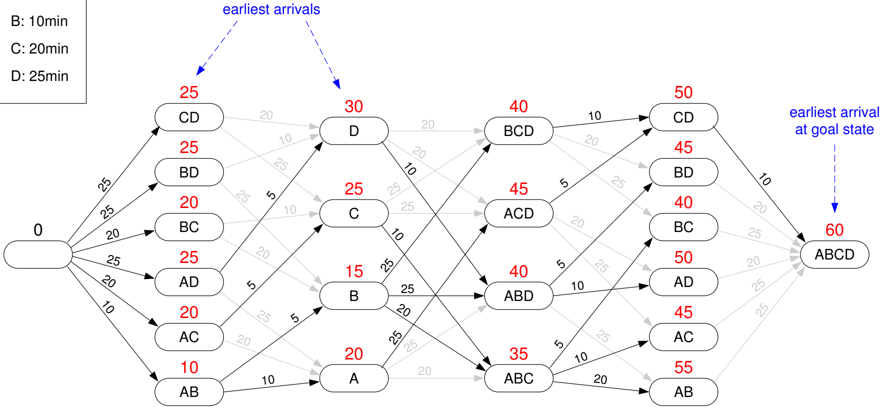


Figure: ... with minimum path lengths updated for all states (nodes).

Quiz: Bridge traversal VI

A: 5min

B: 10min

C: 20min

D: 25min

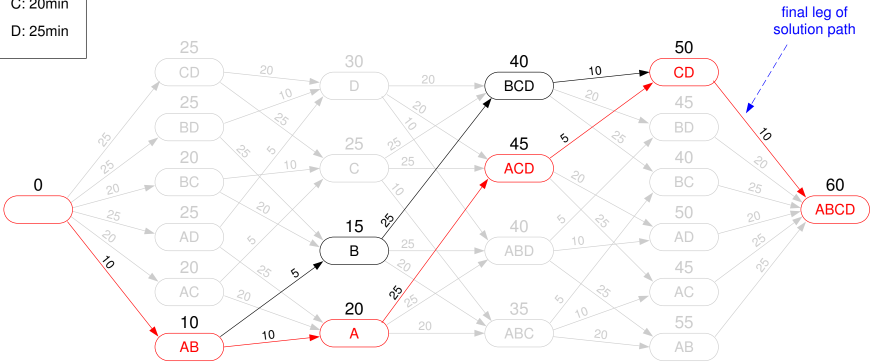
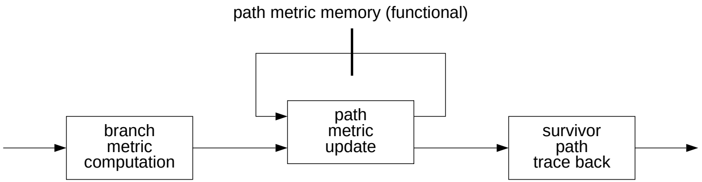


Figure: The surviving trace yields the solution (shortest path through trellis).

General policy

The procedure is known as "Dynamic Programming" and goes

1. Assign each branch a cost metric.
2. Update all path metrics by discarding those edges found to be suboptimal (add-compare-select = ACS).
3. Starting from the final state, trace back the surviving path.



Note:

- "Programming" is used here as a synonym for "finding an optimal plan" (a historical choice that predates computer programming).
- Step 2. is to be carried out recursively stage by stage.

Richard Bellman's principle of optimality

- Rather than tackling a problem by evaluating all possible solutions, Dynamic Programming solves many smaller problems with each subproblem building on the answers to the previous ones.

Richard Bellman's principle of optimality

- Rather than tackling a problem by evaluating all possible solutions, Dynamic Programming solves many smaller problems with each subproblem building on the answers to the previous ones.
- Not all multi-stage decision problems can be solved in this way, though. Those that can satisfy

Bellman's principle of optimality

The globally optimum solution includes no suboptimal local decision.

In Bellman's original words from 1957:

> *An optimal policy has the property that, regardless of the decisions taken to enter a particular state, the remaining decisions made for leaving that stage must constitute an optimal policy.*

Richard Bellman's principle of optimality

- Rather than tackling a problem by evaluating all possible solutions, Dynamic Programming solves many smaller problems with each subproblem building on the answers to the previous ones.
- Not all multi-stage decision problems can be solved in this way, though. Those that can satisfy

Bellman's principle of optimality

The globally optimum solution includes no suboptimal local decision.

In Bellman's original words from 1957:

> *An optimal policy has the property that, regardless of the decisions taken to enter a particular state, the remaining decisions made for leaving that stage must constitute an optimal policy.*

"Life can only be understood backwards, but must be lived forwards."
(Sören Kierkegaard)

Applications

In numerous problems of finding an optimum fit or course of action:

- Error correction coding (Viterbi algorithm).
- Automatic labelling of speech segments (dynamic time warping).
- Genome sequencing
  (Smith-Waterman, Needleman-Wunsch, and Sankoff algorithms).
- Stereo vision (correspondence problem).
- Video coding.
- Digital watermark detection.
- Cell library binding (as part of logic optimization).
- Flight trajectory planning.
- Knapsack problems (for non-negative integer weights).
- Control of hybrid and recuperating engines and vehicles.
- ...