Quiz 10

CSCE 580

March 30, 2017

Consider the state-space search problem in the figure below (Source: S. Edelkamp and S. Schroedl. *Heuristic Search: Theory and Applications*. Morgan-Kaufmann, 2012.), where heuristic estimates (the *h* function) are shown in parenthesis for each node, the start node is *a*, and the goal node is *g*.



Does Dynamic Programming use *h*?

**Answer**: no.

Run Dynamic Programming by hand by filling out the table below. Break ties in the order up, left, right, and then down.

|  |  |  |  |
| --- | --- | --- | --- |
| Step | node | Cost-to-goal | child node (backpointer) |
| 1 | *g* | 0 | nil |
| 2 | *d* | 5+0=5 | *g* |
| 3 | *c* | min(3+5=8, 10+5=15)=8 | *d* |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |

**Answer**:

|  |  |  |  |
| --- | --- | --- | --- |
| Step | node | Cost-to-goal | child node (backpointer) |
| 1 | *g* | 0 | nil |
| 2 | *d* | 5+0=5 | *g* |
| 3 | *c* | min(3+5=8, 10+5=15)=8 | *d* |
| 4 | *a* | min(8+6=14 (c), 10+5 (d))=14 | *c* |
| 5 | *b* | 2+14=16 | *a* |
| 6 | *e (tie with f)* | 4+16=20 | *b* |
| 7 | *f* | min(3+20=23, 4+16=20) | *b* |

Note that the backpointer can be used to find the shortest path from any node to the goal. However, as noted in the last paragraph on p.103 of the textbook, the shortest path can be determined without using the backpointer with a local search among the successors of each node. For example, suppose that we are looking for the shortest path from f to g. Since cost-to-goal(f) = 20, cost-to-goal(e) = 20, and coast-to-goal(b) =16, the next node on the shortest path is b, not e, because coast-to-goal(f) = min(cost(<f,e>)+cost-to-goal(e), cost(<f,b>)+cost-to-goal(b)), and this is minimized by choosing <f,b>.

