A*: Proof of Optimality

George Konidaris
gdk@cs.duke.edu

January 16, 2015

We would like to prove that A*, equipped with an admissible heuristic, always finds an optimal solution. Recall that an admissible heuristic h is one such that:

$$h(s) \leq h^*(s),$$

for all states s, where h^* measures the true cost from state s to the goal. In other words, h is optimistic—it always *underestimates* the cost from s to the goal.

We proceed using a **proof by contradiction**: we *assume that what we are trying to prove does not hold, and then show that this leads to an impossible situation.* Here, we assume that we run A* on a problem and it returns a solution, s_a, with cost $g(s_a)$, and there is another solution that we didn't find, s_{opt}, with cost $g(s_{opt})$ such that:

$$g(s_{opt}) \leq g(s_a).$$

(Recall that $g(s)$ is the true *cost-to-get* from the start state to node s.) Now, consider the *final node expansion*, where s_a was taken off the frontier and evaluated. This is depicted in Figure 1.

Note that when selecting a node from the frontier, we select the node s such that $g(s) + h(s) \leq g(s') + h(s')$, for all other nodes s' in the frontier. Thus, when we were selecting s_a from the frontier, it had the lowest total $g(s) + h(s)$—but since s_a is a goal node, $h(s_a) = 0$ and $g(s_a)$ is the exact cost from the start node to s_a.[1]

Now we note that s_{opt} must have had some ancestor node in the frontier; lets call it s_b. (Note that s_b could be the start node.) Since h is admissible and underestimates the cost the goal:

$$h(s_b) + g(s_b) \leq g(s_{opt}).$$

But since s_a was selected *instead of* s_b, we see that:

$$h(s_a) + g(s_a) = g(s_a) \leq h(s_b) + g(s_b) \leq g(s_{opt}),$$

and hence we see that $g(s_a) \leq g(s_{opt})$. But we started out by assuming that $g(s_{opt}) < g(s_a)$, and so we have reached a contradiction. □

[1] This is the fact I was missing.

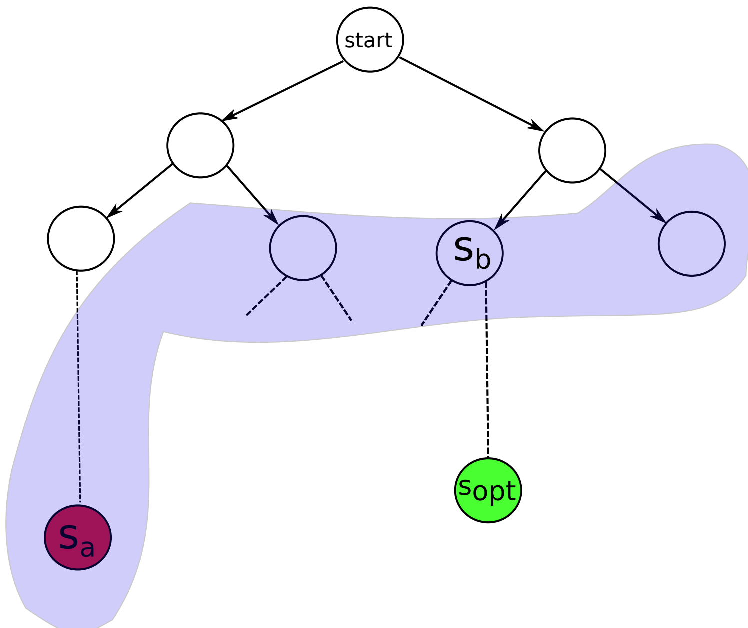

Figure 1: The final node expansion—s_a is selected for expansion from the frontier, but s_b, the ancestor of s_{opt}, is not.