# The Optimality of A*

## A proof

Courtesy of Nate Rojas 2019

# Introduction
A* is a wonderful algorithm. It has the great property that, when coupled with an admissible heuristic, it finds optimal solutions. Lets prove that optimality!

DISCLAIMER: I pulled this proof directly from an edX forum post that wrote (so you may have seen it before).

# Proof
TLDR;
A* ensures optimality because all sub optimal paths will reach a cost that will be greater than some alternative. There is only one alternative which will get us to the goal node - an optimal path.

First some definitions:
- $s_k$ - a sub-path of length k of the sub-optimal goal path $G_{sub}$
- $o_k$ - a sub-path of length k of the optimal goal path $G_O$
- $C(P)$ - the *total* cost of path $P$
- $h(P)$ - an admissible heuristic which estimates the cost from the last node in path $P$ to the goal.
- $f(P) = C(P) + h(P)$ - is the cost estimate on path $P$

Two lemmas:
1. $h -$ admissible $\Rightarrow \forall j : f(o_j) \leq C(G_O)$ . That is, the admissibility of $h$ implies that the estimated cost of any $o_j$ is always less than or equal to the total cost of the optimal path.
2. $C(G_{sub}) > C(G_o) \Rightarrow \exists m : C(s_m) > C(G_O)$ . That is, the fact that the sub-optimal path has higher cost than the optimal path (by definition) implies that there exists some sub-path $s_m$ of $G_{sub}$ such that the total cost of $s_m$ is greater than the cost of the optimal path. In other words, somewhere along the way the sub-optimal path will accrue "too much cost."

### Putting it all together
Lemmas 1 & 2 imply the following set up:
for some subpath $s_m$ of any sub-optimal path $G_{sub}$ we can form:

$$[C(s_m) > C(G_O) \geq f(o_j)] \Rightarrow [C(s_m) > f(o_j)] \Rightarrow [f(s_m) > f(o_j)]$$

The algorithm always chooses least cost ■

That is to say, the optimality of A* is ensured because sub-optimal paths will always "saturate" before the goal node is reached and the algorithm will look elsewhere. This leaves us with the worst possible case: All sub-optimal paths are explored up to their "saturation" point before finally discovering the optimal goal path.