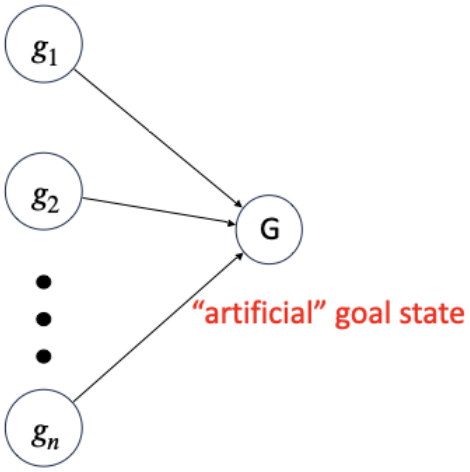
CAS CS 440
Lec 3

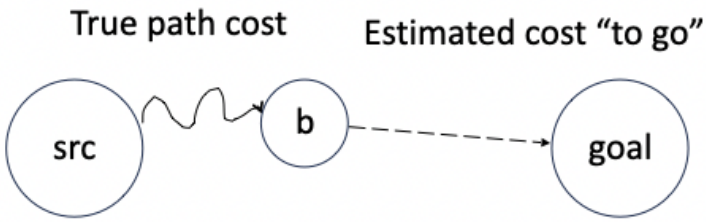
Search III

1. Dijkstra's Algorithm
    a. Pros:
        i. Works with arbitrary (positive) edge weights
    b. Cons:
        i. Still blind
        ii. Not feasible for large graphs O(|E| + |V|log|V|) runtime, O(|V| + |E|) memory
    c. Can we make this less blind?
2. Making Dijkstra Less Blind
    a. Only a single goal state
        i. If more than one: add an artificial goal state
    b. Design goal:
        i. Make Dijkstra's algorithm aware of the goal state
        ii. Choose to expand paths that lead "towards" the goal
        iii. Rather than sort by true path cost (i.e. known path cost)
            1. Sort by estimated path cost from src → goal



        iv.
    c. How to estimate the path cost from src → goal?
        i. Utility function

3. Utilities and Estimating Cost to Goal
    a. Utility function estimates path cost from a vertex to the goal
    b. Total estimated cost from src → goal = true path cost + estimated cost



    c. If we're here
    d. Warning: when we get shortest path so far:
        i. Dijkstra claims that's the correct answer (to wherever that path leads)
        ii. Are we breaking that by sorting by src → goal cost?
4. Utility Constraints
    a. Cost function = path cost + utility cost (heuristic)
    b. As long as utility is admissible & consistent: A* is optimal
        i. Admissible = never overestimates the true cost (i.e. utility is optimistic)
        ii. Consistent = triangle inequality but for action spaces:

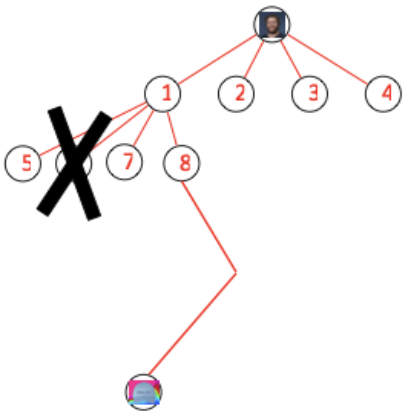
$$f(u) \leq c(u,\ a,\ v) + f(v)$$

Cost of vertex u (to goal)
Cost of vertex v (to goal)
Cost of using action a to transition from u to v

            1.
        iii. Every consistent utility (heuristic_ is admissible

5. BFS Example

    a.

b. 

6. The A* Algorithm
    a. Dijkstra's algorithm (mostly)
        i. Gets shortest path so far based on total estimated cost src → goal
        ii. Everything else is the same as Dijkstra
    b. Dijkstra's algorithm where path comparison is done via src → goal
        i. Rather than true path cost (Dijkstra)
    c. Warning: utility (heuristic) needs to be engineered for every world
        i. Consistency is determined on a world-by-world basis