Graduate AI

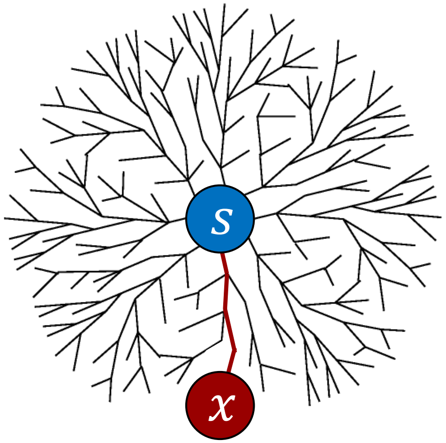
Lecture 3:
Search II

Teachers:
Zico Kolter
Ariel Procaccia (this time)

A* IS OPTIMALLY EFFICIENT

- We say that node x is surely expanded by A* (tree search) if $f(x) < f(t^*)$, where t^* is the optimal goal
- Theorem [Dechter and Pearl 1985]: Any algorithm that is optimal given a consistent heuristic will expand, whenever the heuristic is consistent, all nodes surely expanded by A*

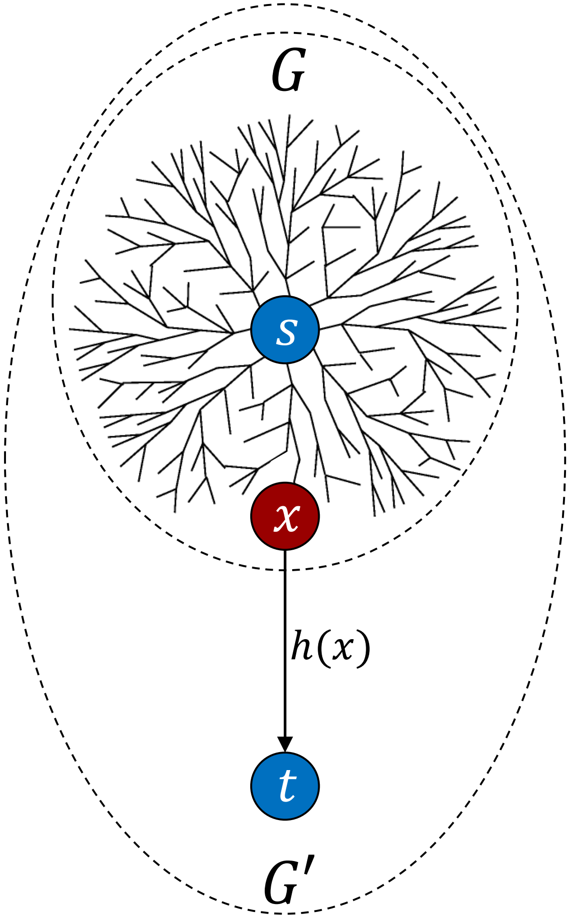
PROOF OF THEOREM



- Let I be an instance with graph G and consistent heuristic h
- Assume node x is surely expanded by A*
- Denote $f(x) < f(t^*) = C^*$
- Let B be an optimal algorithm that does not expand x

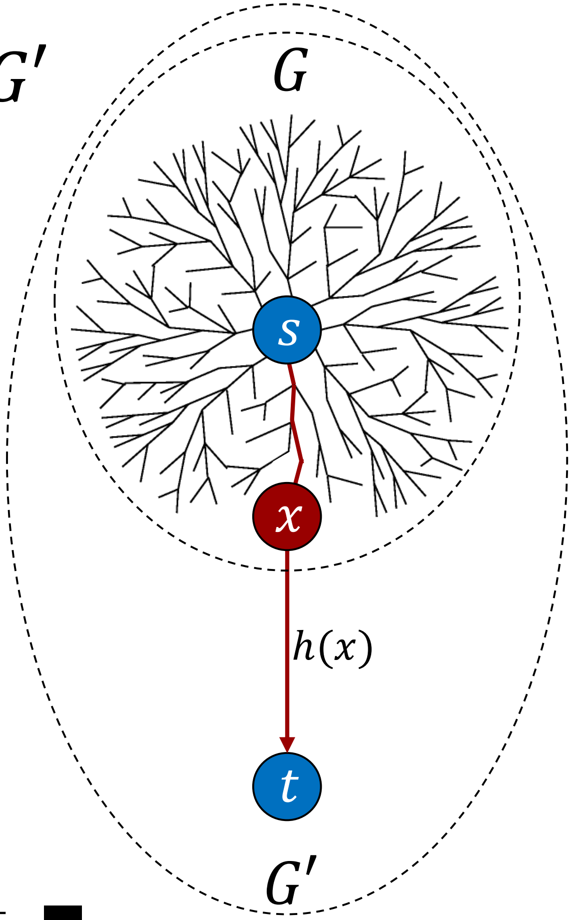
PROOF OF THEOREM

- Create G' by adding a new goal t, and an edge (x, t) with cost $h(x)$
- h' is the same as h, and $h'(t) = 0$
- Lemma: h' is consistent
    - Clearly true on pairs that do not include t
    - For pairs (y, t), $h'(y) = h(y) \leq c(y, x) + h(x)$ $= c(y, t) = c(y, t) + h'(t)$

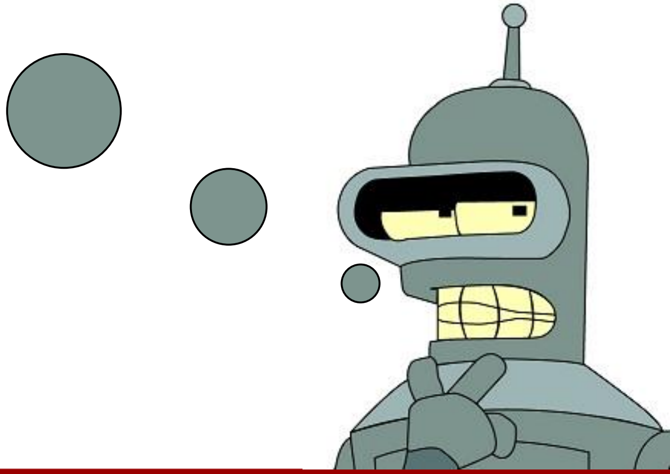


PROOF OF THEOREM

- On the new instance I' defined by G' and h', A* will find the goal t with cost
  $$g(t) + h'(t) = g(x) + h(x) < C^*$$
- Because B does not expand x, I' looks identical to I, and B will find a solution of cost C^*
- This is a contradiction to the assumption that B is optimal whenever the heuristic is consistent ■

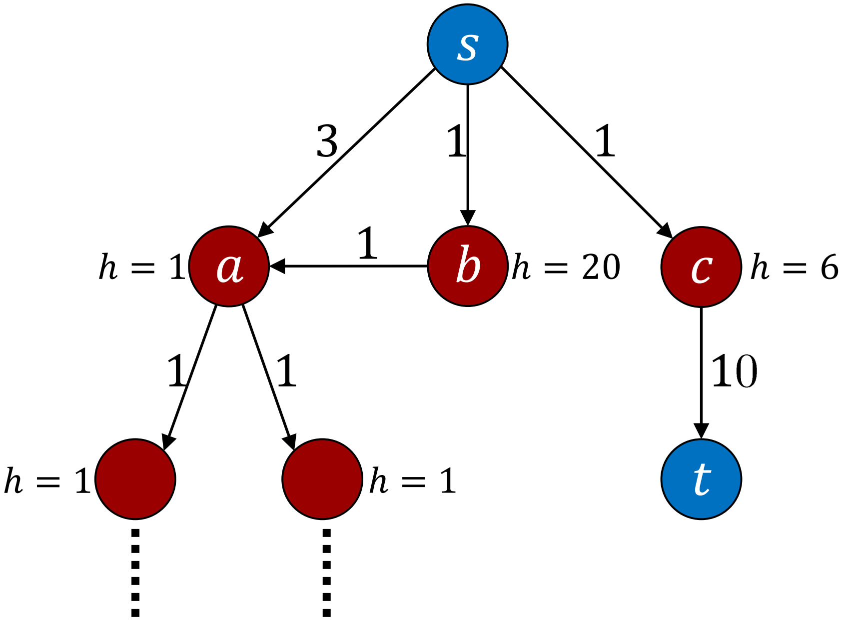


Why does the proof fail if we replace "consistent" with "admissible"?

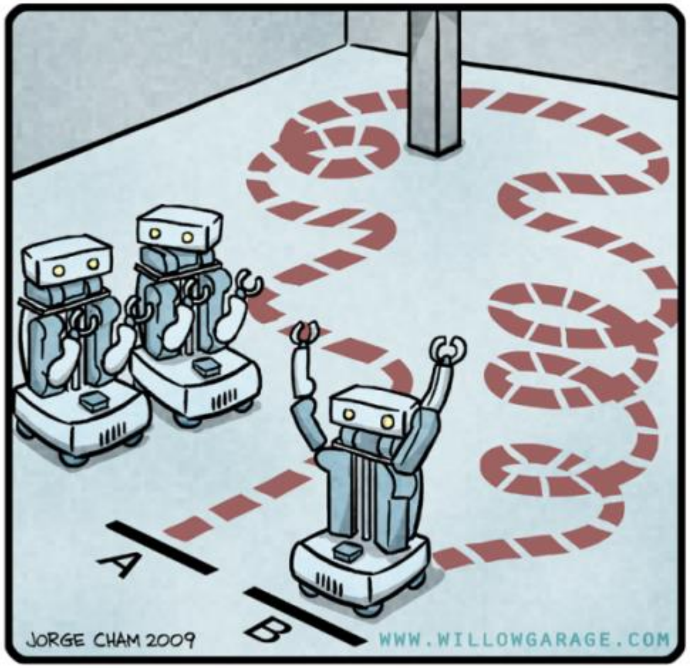


A* IS OPTIMALLY EFFICIENT

- In fact, the theorem is false when the heuristic is only admissible
- In the example on the right, algorithm B will find the optimal solution while expanding fewer nodes than A*



Alg B: Conduct exhaustive search except for expanding a; then expand a only if it has the potential to sprout cheaper solution

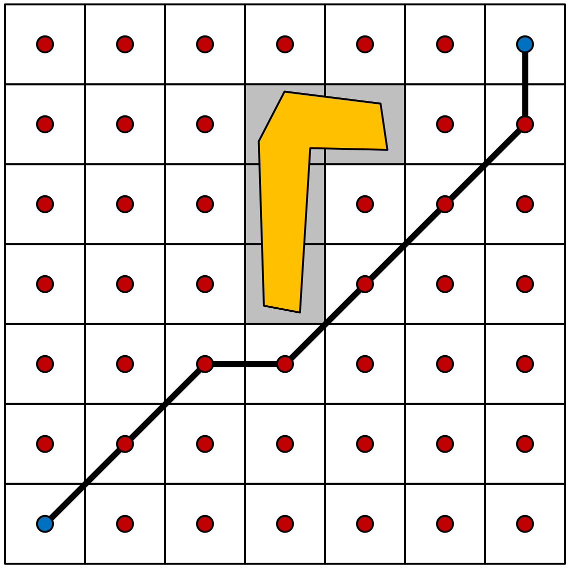


"HIS PATH-PLANNING MAY BE SUB-OPTIMAL, BUT IT'S GOT FLAIR."

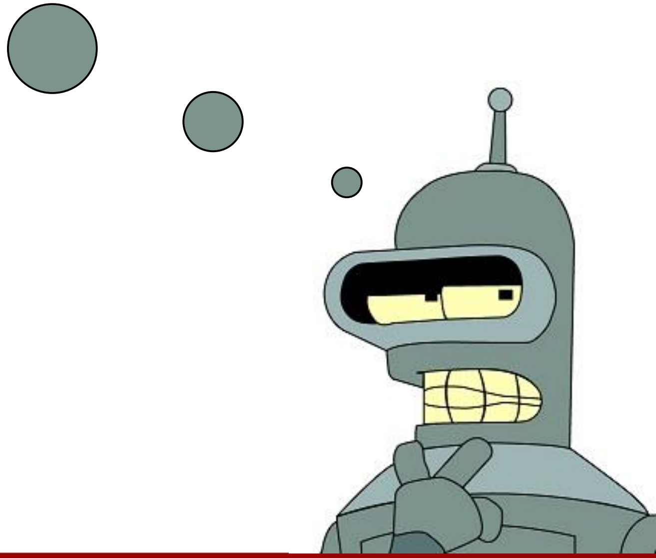
APPLICATION: MOTION PLANNING

Motion planning

- Navigating between two points while avoiding obstacles
- A first approach: define a discrete grid
- Mark cells that intersect obstacles as blocked
- Find path through centers of remaining cells

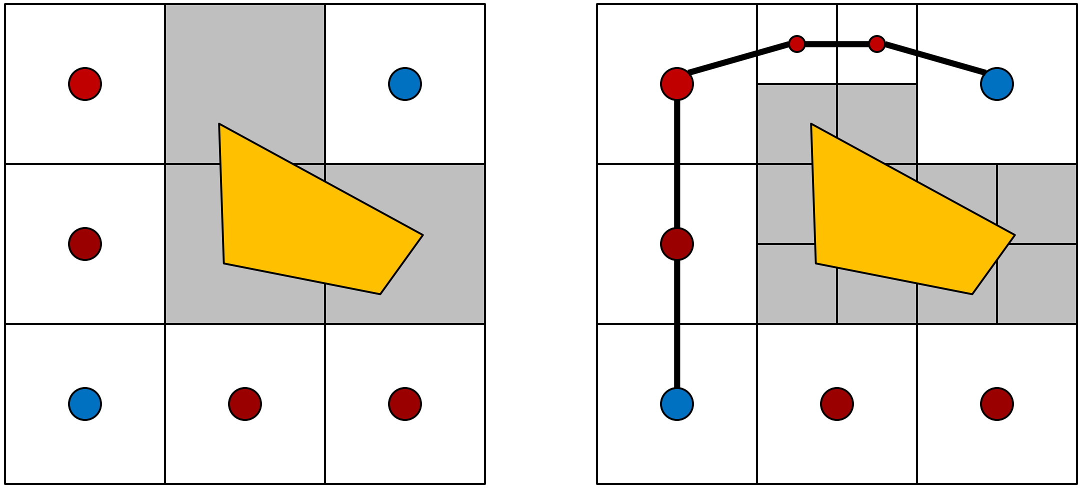


Is this approach optimal? Complete?



CELL DECOMPOSITION

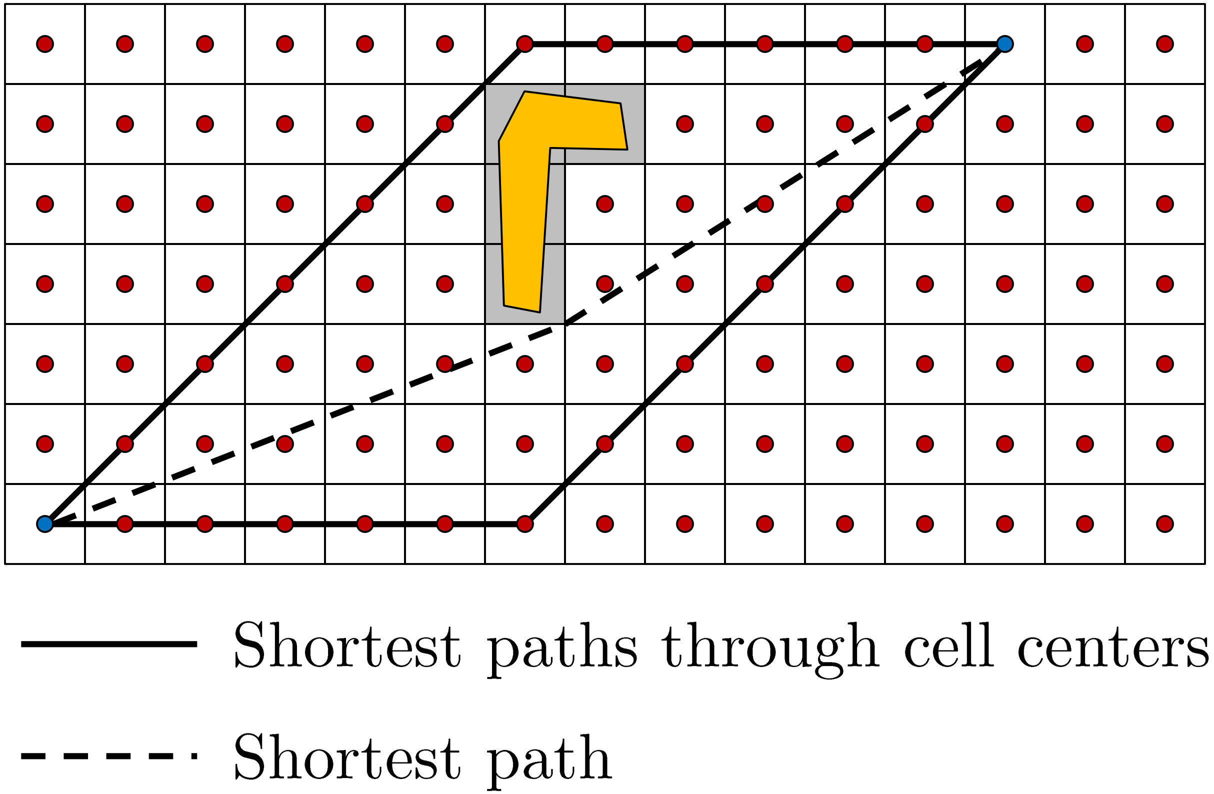
- Distinguish between
  - Cells that are contained in obstacles
  - Cells that intersect obstacles
- If no path found, subdivide the mixed cells



IS IT COMPLETE NOW?

- An algorithm is resolution complete when:
  a. If a path exists, it finds it in finite time
  b. If a path does not exist, it returns in finite time
- Assume obstacles are closed sets, so free space is an open set
- Poll 1: Cell decomposition satisfies:
  [✓ 1.] a but not b
  2. b but not a
  3. Both a and b
  4. Neither a nor b

Cell decomposition

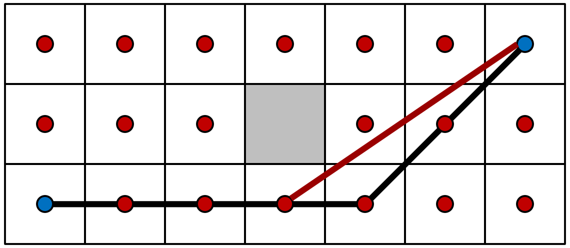


—— Shortest paths through cell centers

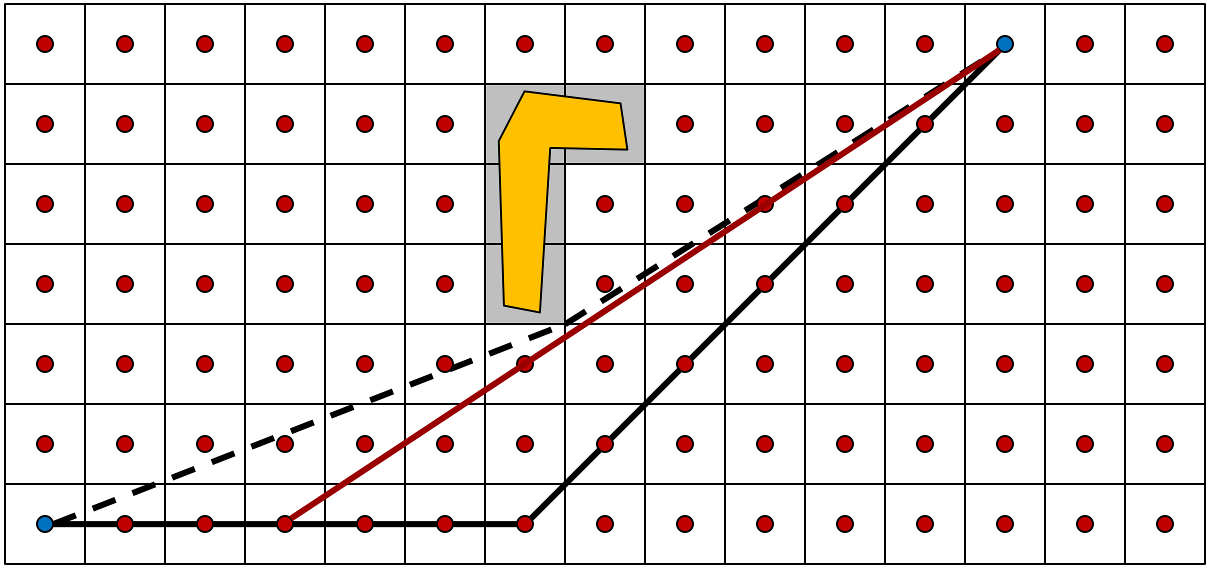
- - - - Shortest path

SOLUTION 1: A* SMOOTHING

- Allows connection to further states than neighbors on the grid
- Key observation:
  - If $x_1, \ldots, x_n$ is valid path
  - And x_k is visible from x_j
  - Then $x_1, \ldots, x_j, x_k, \ldots, x_n$ is a valid path



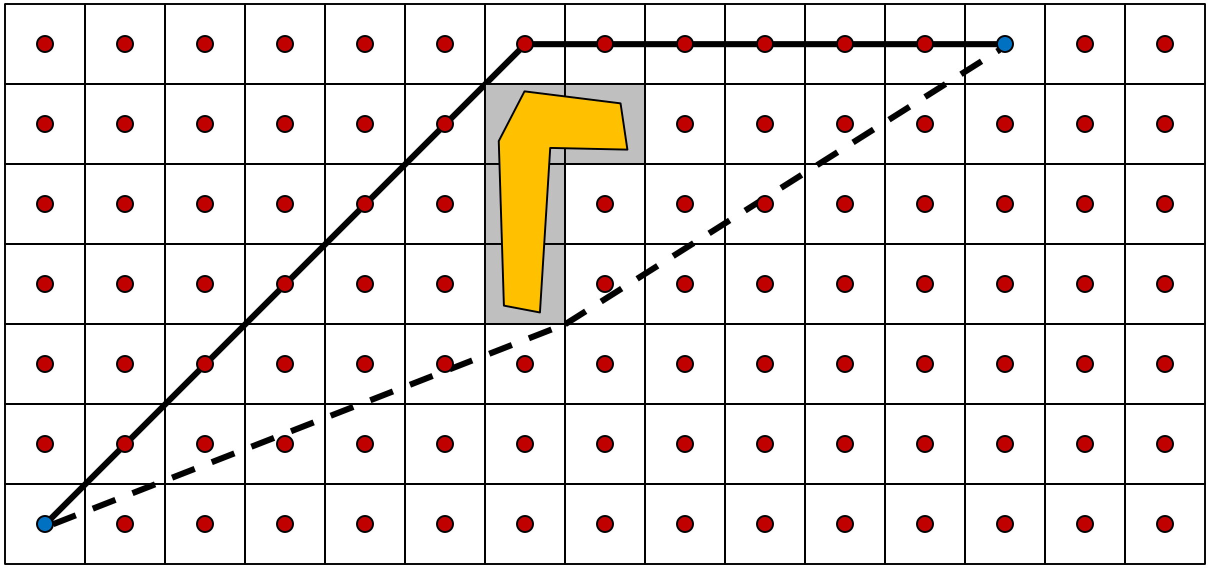
Smoothing works!



—— A shortest path through cell centers

- - - - Shortest path

SMOOTHING DOESN’T WORK!



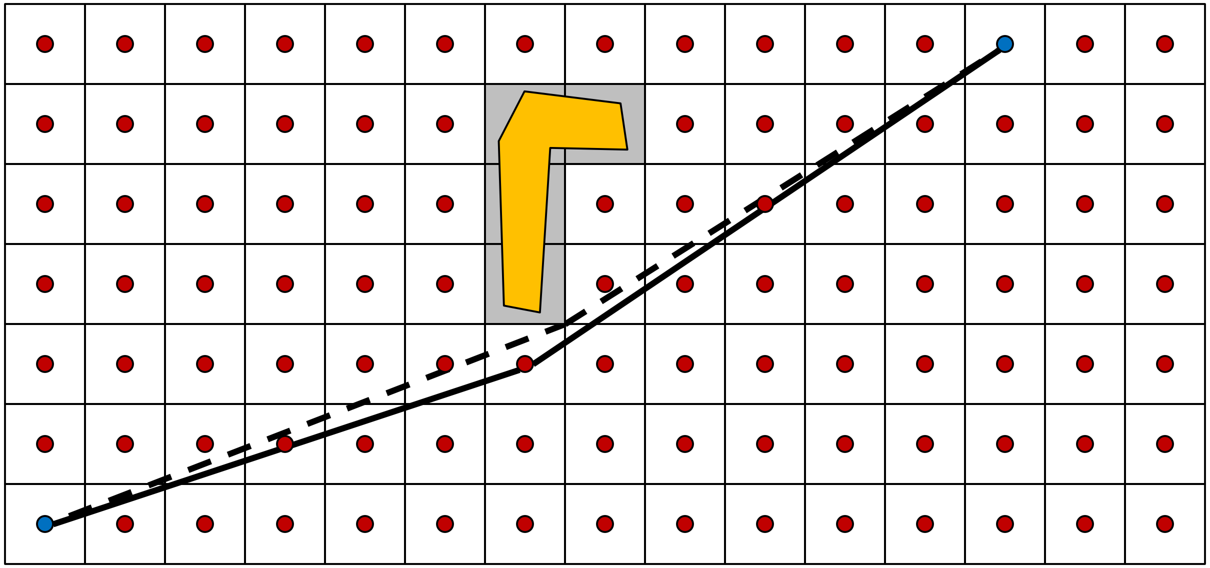
— A shortest path through cell centers

- - - Shortest path

SOLUTION 2: THETA*

- Allow parents that are non-neighbors in the grid to be used during search
- Standard A*
  - $g(y) = g(x) + c(x, y)$
  - Insert y with estimate
$$f(y) = g(x) + c(x, y) + h(y)$$
- Theta*
  - If parent(x) is visible from y, insert y with estimate $f(y) = g(\text{parent}(x)) + c(\text{parent}(x), y) + h(y)$

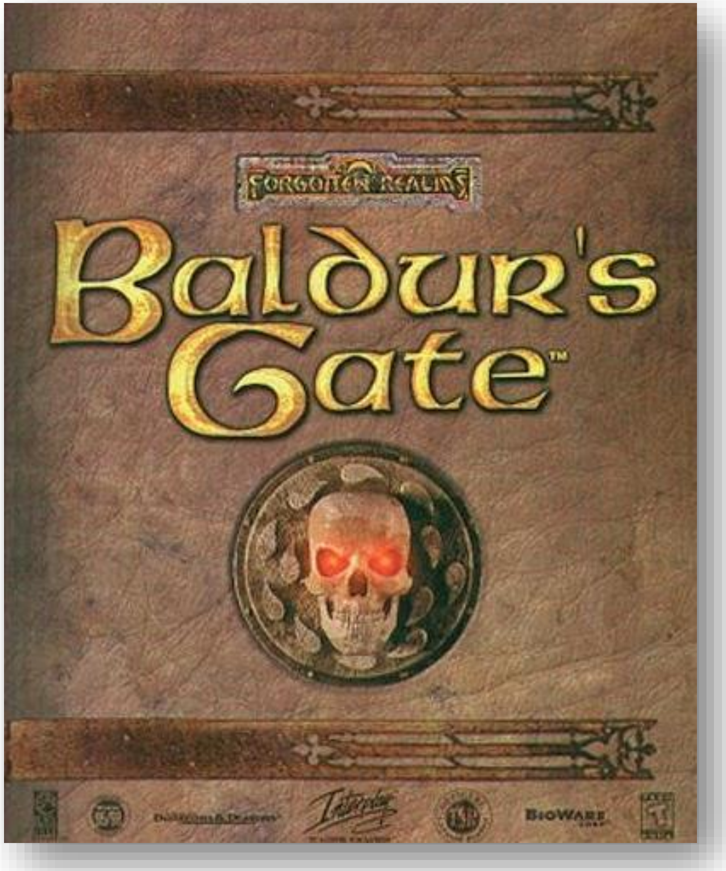
Theta* works!

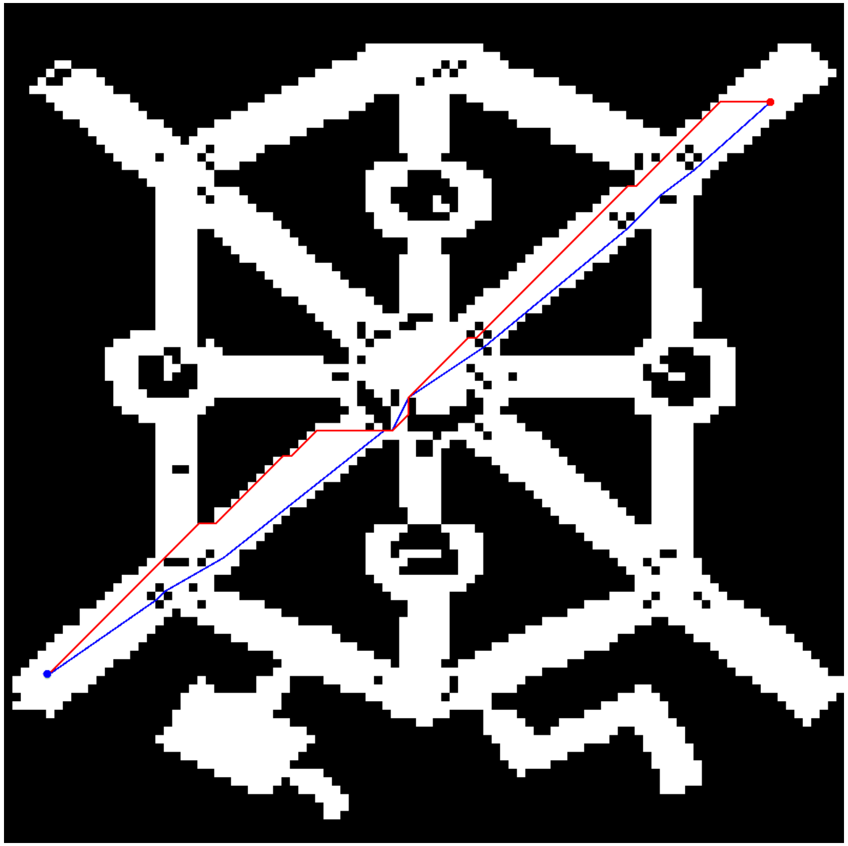


—— Theta* path, I think ☺

- - - - Shortest path

THETA* WORKS!

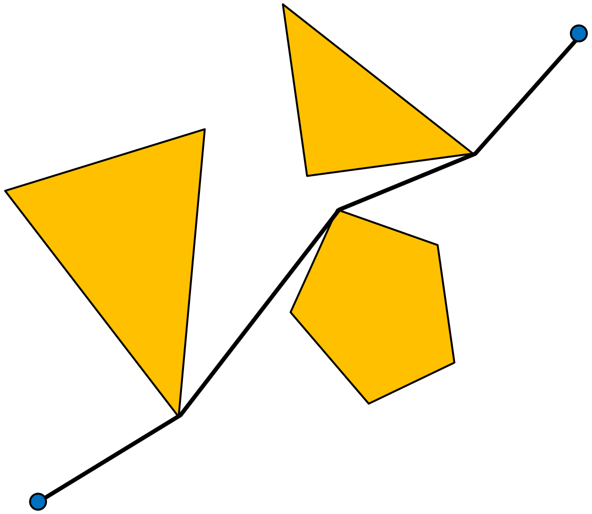




[Nash, AIGameDev 2010]

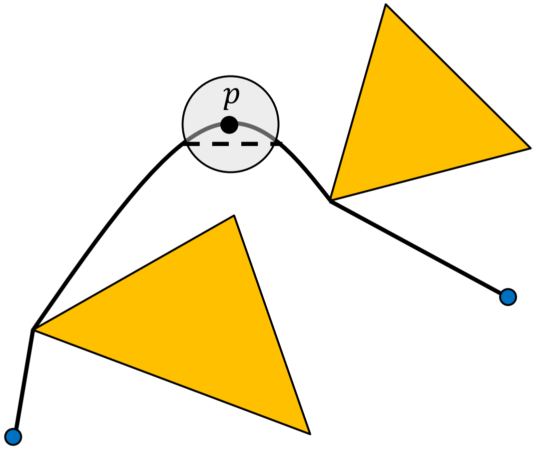
THE OPTIMAL PATH

- **Polygonal path:** sequence of connected straight lines
- **Inner vertex of polygonal path:** vertex that is not beginning or end
- **Theorem:** assuming (closed) polygonal obstacles, shortest path is a polygonal path whose inner vertices are vertices of obstacles



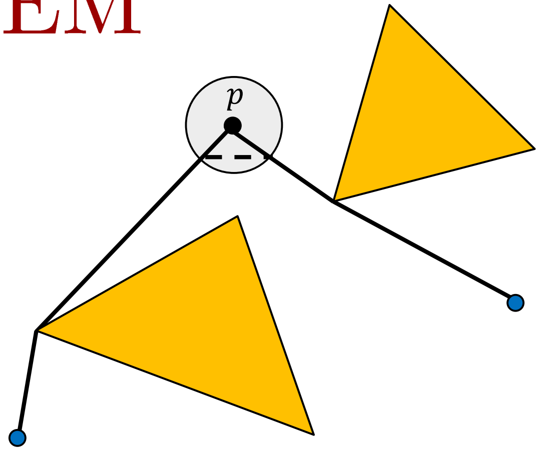
PROOF OF THEOREM

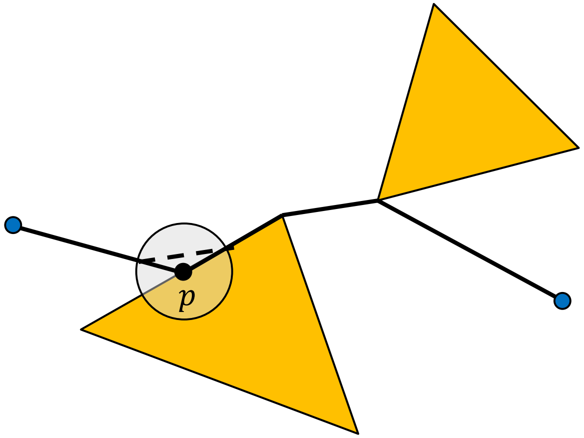
- Suppose for contradiction that shortest path is not polygonal
- Obstacles are polygonal ⇒ $\exists$point p in interior of free space such that "path through p is curved"
- $\exists$disc of free space around p
- Path through disc can be shortened by connecting points of entry and exit



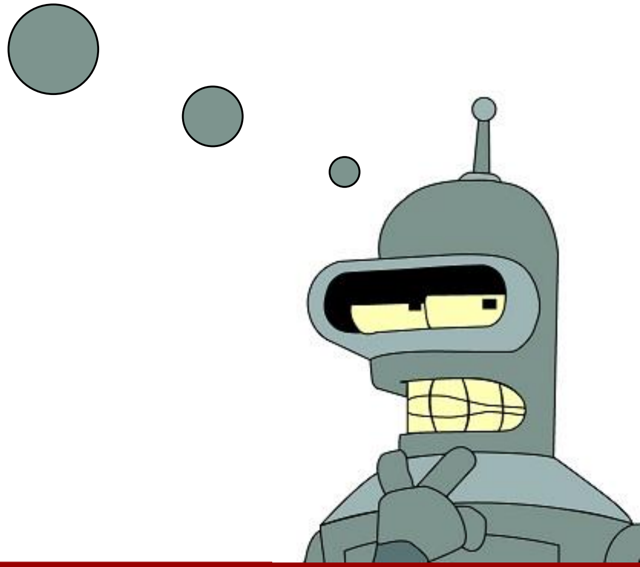
PROOF OF THEOREM

- Path is polygonal!
- Vertex cannot lie in interior of free space, otherwise we can do the same trick
- Vertex cannot lie on a the interior of an edge, otherwise we can do the same trick ■

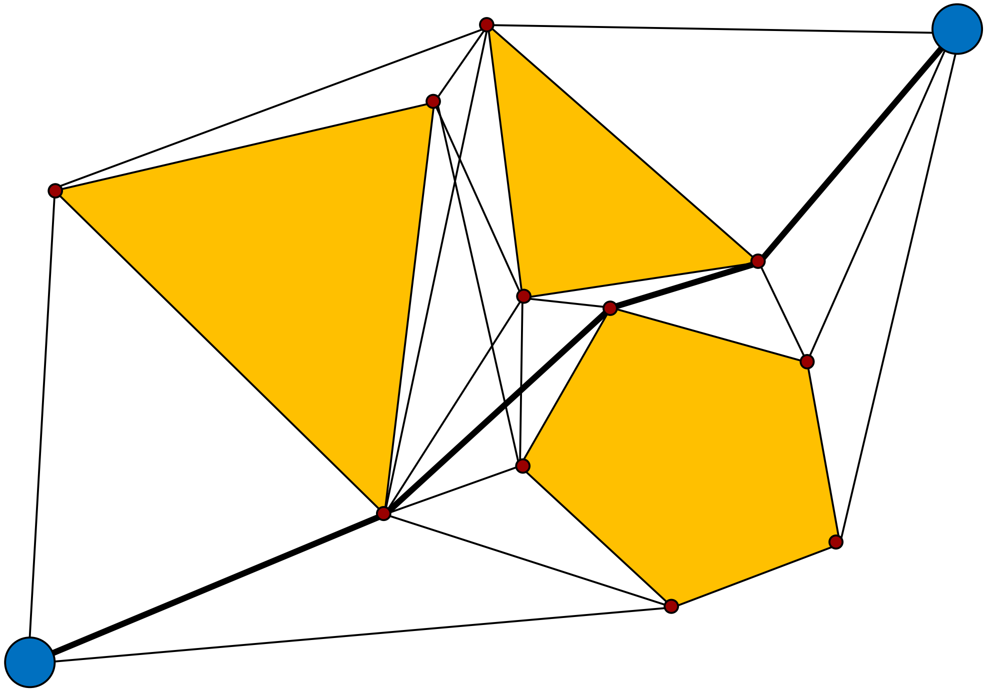




How would we define a graph on which A* would be optimal?



VISIBILITY GRAPH



Vertices = vertices of polygons and s, t

Edges = all (x, y) such that y is visible from x

VISIBILITY GRAPH

- Poll 2: Let n be the total number of vertices of all polygons. How many edges will the optimal path in the visibility graph traverse in the worst case?

1. $\Theta(\sqrt{n})$

[✓ 2.] $\Theta(n)$

3. $\Theta(n^2)$

4. $\Theta(n^3)$

SUMMARY

- Terminology and algorithms:
    - Cell decomposition
    - Resolution completeness
    - Theta*
- Theorems:
    - A* is optimally efficient
    - Geometry of shortest path with polygonal obstacles
- Big ideas:
    - A* can be modified to work well in continuous spaces