A^* Search

$O(E \log V) \leftarrow$ regular heap

Dijkstra: SSSP in $O(E + V \log V) \leftarrow$ fancy heap

if no negative edges

```
Dijkstra(G, s, t):
    parent = {}, dist = {}
    Q = PriorityQueue([(0, s, None)])
    while Q:
        d, u, p = Q.pop()                        d = d - h(u)
        if u in parent:   | or: u in dist & d ≥ dist[u]:
            continue
        parent[u] = p, dist[u] = d
        if u == t:    break

        for v in u.adj:
            Q.append((dist[u] + w(u->v), v, u))
                              + h(u)
    return dist, parent
```

A^*

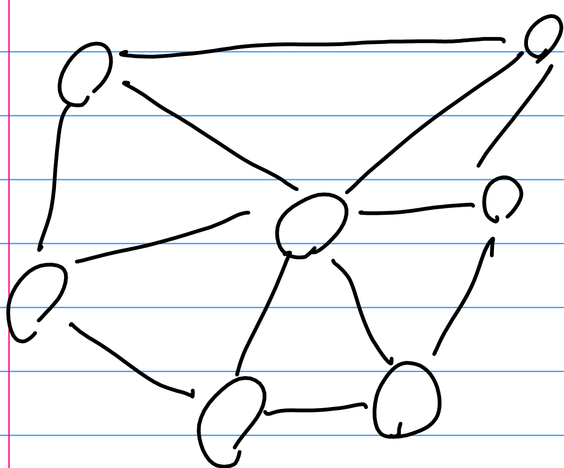
1: Correct but slow w/ negative edges

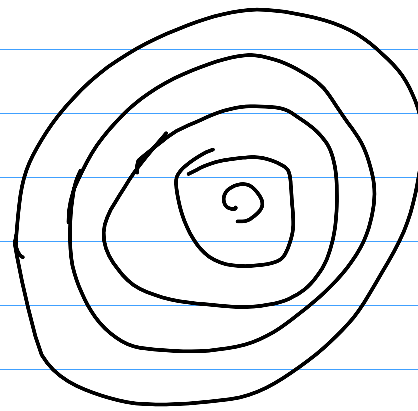
what about shortest $s \leadsto t$ path?

1: stop once visit t

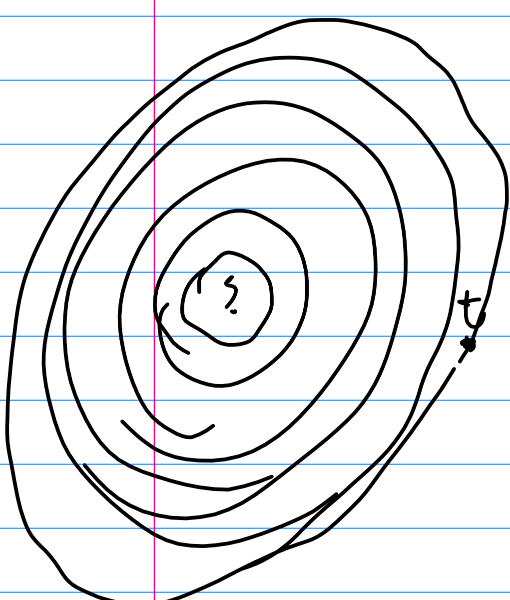
Imagine road network

shortest (or fastest) s–t path





Visits vertices in order of distance from s
ends when reaches t



Austin – SF: 1760 miles

Austin – NYC: 1740 miles

Hasto: maybe ∃ portal NY – SF

<u>Heuristic</u> h, (or "potential")

visit in order of increasing

$\text{dist}(s, u) + h(u)$

Equivalent to adjusting lengths

$$w_h(u \to v) = w(u \to v) + h(v) - h(u)$$

$$\text{length}_h(u_1 \to u_2 \to u_3 \to \cdots \to u_k) = \text{length}(u_1 \to u_2 \to \cdots \to u_k) + h(u_k) - h(u_1)$$

adjustment depends on endpoints (u_1, u_k) but not path

$\Rightarrow$ shortest paths same for w & w_h.

$$\text{dist}_h(s,u) = \text{dist}(s,u) + h(u) - h(s)$$

$\Rightarrow$ Dijkstra w/ w_h visits vertices in order of $\text{dist}(s,u) + h(u)$

What $h()$ are good?

h "admissible": $h(u) \le \text{dist}(u,t)$, $h(t) = 0$

when visit t, all unvisited u have

$\text{dist}(s,u) + h(u) \ge \text{dist}(s,t) + h(t)$

admissible $\Rightarrow$ $\text{dist}(s,u) + \text{dist}(u,t) \ge \text{dist}(s,t)$

$\Rightarrow$ $\text{dist}(s,t)$ we've found is optimal

$\Rightarrow$ A^* is correct,

But can still be slow (exponentially slow!)

h "consistent": $h(t) = 0$ & $w_h(u \to v) \geq 0 \;\; \forall u, v$

$$w(u \to v) + h(v) - h(u) \geq 0$$
$$h(u) - h(v) \leq w(u \to v)$$

consistent $\Rightarrow$ admissible:

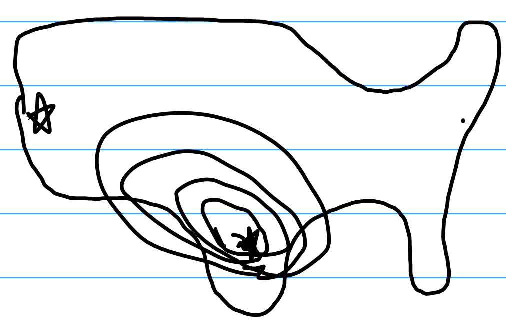
$$0 \leq dist_h(u, t) = dist(u \to t) + h(t) - h(u)$$
$$= dist(u \to t) - h(u)$$
$$\Rightarrow h(u) \leq dist(u \to t)$$

consistent $\to$ A^* correct & $O(E + V \log V)$

<u>example</u> Euclidean, $w(u \to v) = \|u - v\|$

$h(u) = \| u - t \|$

$$\|u - t\| - \|v - t\| \leq \|u - v\| \quad \text{by } \Delta \text{ inequality}$$



Johnson's Algorithm

pick arbitrary v^*

$h(u) = -dist(v^*, u)$

$-dist(v^*, u) + dist(v^*, v) \leq w(u \to v)$

$\Rightarrow w_h(u \to v) \geq 0 \quad \forall u, v$

$\Rightarrow$ Dijkstra on w_h is fast, even if G
has -ve edges

v^* $\Rightarrow$ APSP in $O(VE) + O(V \cdot (E + V \log V))$

1 Bellman Ford   V Dijstras

even w/ negative edges