Lecture 16: Dijkstra’s Algorithm
Dijkstra’s Algorithm

- Defined also: directed graphs
- Designate starting vertex $s$
- $O((m + n) \log n)$ time with binary heap
- Output is a *single source shortest path tree*
- As defined: non-negative edge weights only!
Dijkstra’s Algorithm

- Defined also: directed graphs
- Starting vertex $s = A$ above
- $O((m + n) \log n)$ time with binary heap
- Output is a single source shortest path tree
- As defined: non-negative edge weights only!
Negative Edge Weights?

- Suppose no \textit{negative cost cycles}
- Largest magnitude negative edge cost \(-C\)
- Solve SSSP with negative edge costs?
  - Add \(C\) to every edge cost
  - Run Dijkstra

Correct shorter \(S \rightarrow T\) path: \(S \rightarrow A \rightarrow B \rightarrow T\)
Algorithm: \(S \rightarrow T\) direct \(\times\)
Only one negative edge cost?

- Suppose no *negative cost cycles*
- Only one edge with negative cost
- How can I find shortest path $s$ to $t$?

**nouse:** remove $e$ from $G$. Run Dijkstra’s on $G$ from $s$. Path $s \leadsto t$.

with: $s \leadsto u \rightarrow v \leadsto t$ from first run of Dijkstra’s in input run of Dijkstra’s from $v$ in $G$. 

find in $O(m)$ via $e = (u, v)$
Escape Problem

- On island with velociraptors
- Want to get $S$ to $T$
- Directed graph, $G = (V, E)$
  - Edge has probability $p(e)$ dinosaur eats me
  - Safely arrive $\prod_{e \in P}(1 - p(e))$

- How to use Dijkstra’s without re-programming?

$$\text{want } \max \prod (1 - p_e)$$
$$\equiv \max \sum \log (1 - p_e)$$
$$\equiv \min \sum [-\log(1 - p_e)]$$