Algorithms & Models of Computation

CS/ECE 374, Fall 2020

18.2

Bellman Ford Algorithm

Algorithms & Models of Computation

CS/ECE 374, Fall 2020

18.2.1

Shortest path with negative lengths: The challenge

Shortest Paths with Negative Lengths

Lemma 18.1.

Let **G** be a directed graph with arbitrary edge lengths. If $s = v_0 \rightarrow v_1 \rightarrow v_2 \rightarrow \ldots \rightarrow v_k$ is a shortest path from s to v_k then for 1 < i < k:

- $lackbox{0} \quad s = m{v}_0
 ightarrow m{v}_1
 ightarrow m{v}_2
 ightarrow \ldots
 ightarrow m{v}_i$ is a shortest path from $m{s}$ to $m{v}_i$
- **2** False: $dist(s, v_i) \leq dist(s, v_k)$ for $1 \leq i < k$. Holds true only for non-negative edge lengths.

Cannot explore nodes in increasing order of distance! We need other strategies.

Shortest Paths with Negative Lengths

Lemma 18.1.

Let **G** be a directed graph with arbitrary edge lengths. If $s = v_0 \rightarrow v_1 \rightarrow v_2 \rightarrow \ldots \rightarrow v_k$ is a shortest path from **s** to v_k then for $1 \leq i < k$:

- $lackbox{0} \quad s = m{v}_0
 ightarrow m{v}_1
 ightarrow m{v}_2
 ightarrow \ldots
 ightarrow m{v}_i$ is a shortest path from $m{s}$ to $m{v}_i$
- **2** False: $dist(s, v_i) \leq dist(s, v_k)$ for $1 \leq i < k$. Holds true only for non-negative edge lengths.

Cannot explore nodes in increasing order of distance! We need other strategies.

Shortest Paths with Negative Lengths

Lemma 18.1.

Let **G** be a directed graph with arbitrary edge lengths. If $s = v_0 \rightarrow v_1 \rightarrow v_2 \rightarrow \ldots \rightarrow v_k$ is a shortest path from **s** to v_k then for $1 \leq i < k$:

- $lackbox{0} \quad s = m{v}_0
 ightarrow m{v}_1
 ightarrow m{v}_2
 ightarrow \ldots
 ightarrow m{v}_i$ is a shortest path from $m{s}$ to $m{v}_i$
- **2** False: $dist(s, v_i) \leq dist(s, v_k)$ for $1 \leq i < k$. Holds true only for non-negative edge lengths.

Cannot explore nodes in increasing order of distance! We need other strategies.

THE END

...

(for now)