3-7 Relax! We are SSSP Algorithms.

Hengfeng Wei

hfwei@nju.edu.cn

November 12, 2018

Definition (Shortest Path)

$$G = (V, E, w): \text{ weighted digraph}$$

$$\delta(u, v) = \begin{cases} \min\{w(p) : u \rightsquigarrow^p v\} & \text{if } u \rightsquigarrow v \\ \infty & \text{o.w.} \end{cases}$$

Definition (Shortest Path)

$$G = (V, E, w): \text{ weighted digraph}$$

$$\delta(u, v) = \begin{cases} \min\{w(p) : u \rightsquigarrow^p v\} & \text{if } u \rightsquigarrow v \\ \infty & \text{o.w.} \end{cases}$$

Path *vs.* Simple path

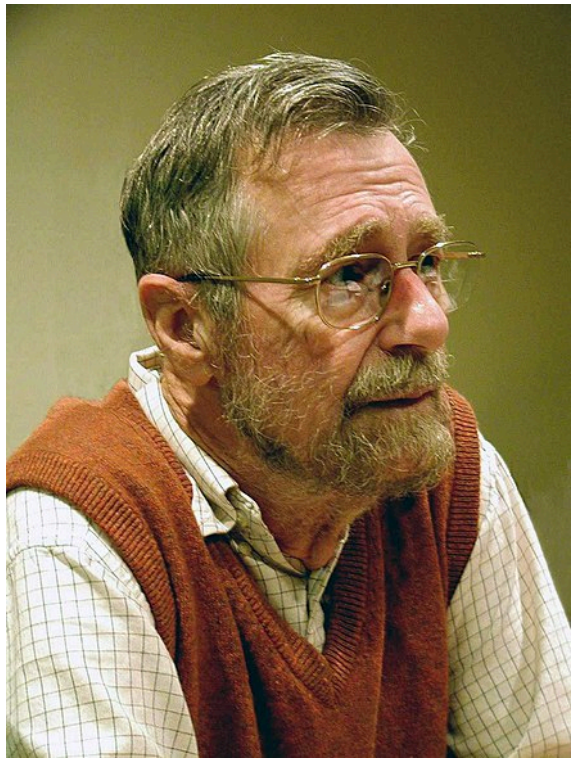
Definition (Shortest Path)

$$G = (V, E, w): \text{ weighted digraph}$$

$$\delta(u, v) = \begin{cases} \min\{w(p) : u \overset{p}{\rightsquigarrow} v\} & \text{if } u \rightsquigarrow v \\ \infty & \text{o.w.} \end{cases}$$

Path *vs.* Simple path

Q : What about undirected graphs?



```
1: procedure DIJKSTRA(G, w, s)
2:     INIT-SINGLE-SOURCE(G, s)
3:     S = ∅
4:     Q = G.V
5:     while Q ≠ ∅ do
6:         u ← EXTRACT-MIN(Q)
7:         S ← S ∪ {u}
8:         for v ∈ G.Adj[u] do
9:             RELAX(u, v, w)
```

```
1: procedure DIJKSTRA(G, w, s)
2:     INIT-SINGLE-SOURCE(G, s)
3:     S = ∅
4:     Q = G.V
5:     while Q ≠ ∅ do
6:         u ← EXTRACT-MIN(Q)
7:         S ← S ∪ {u}
8:         for v ∈ G.Adj[u] do
9:             RELAX(u, v, w)
```

Array: $O(n^2)$

Min-heap: $O(E \log V)$

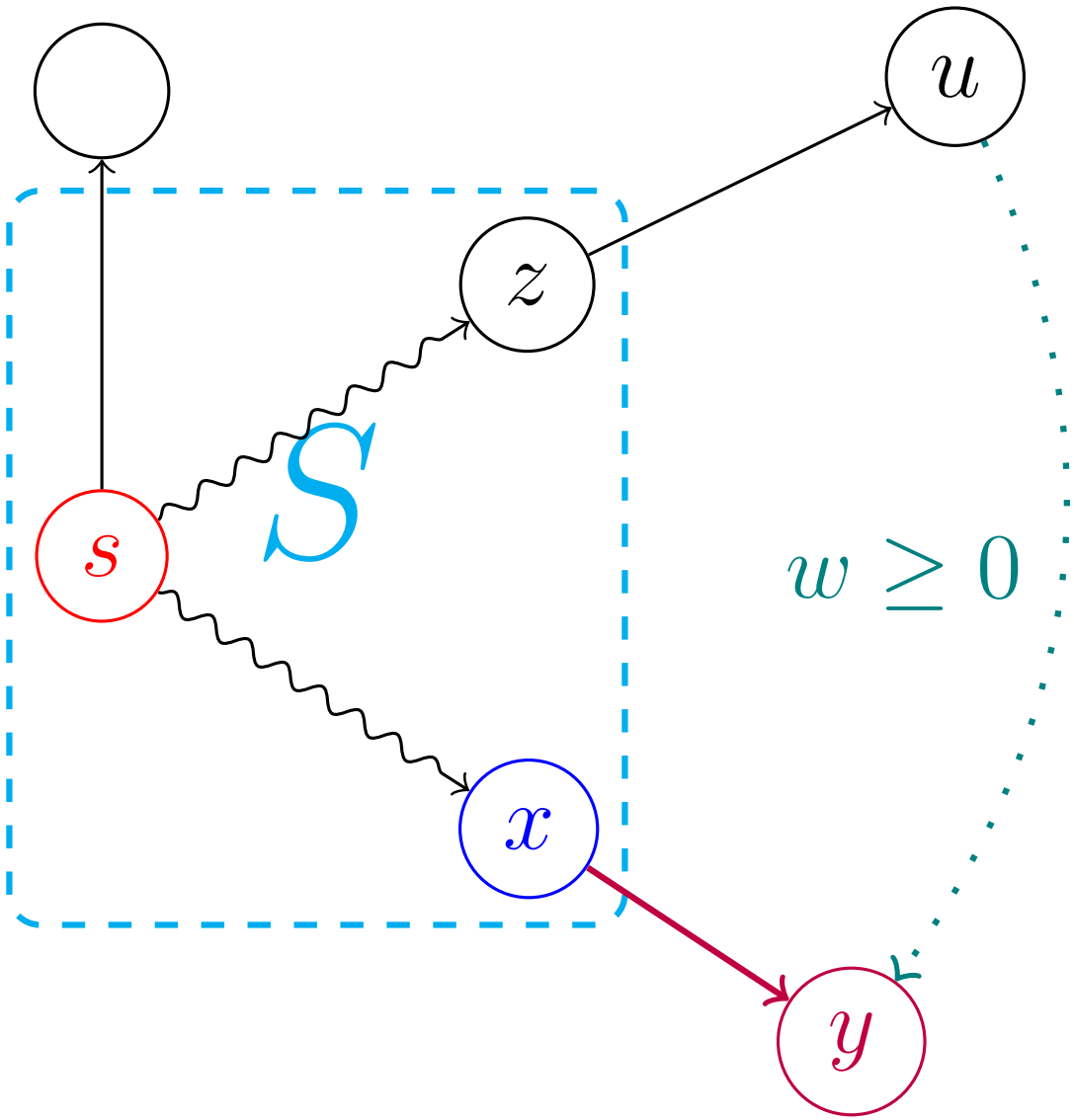
Fib-heap: $O(V \log V + E)$

```
1: procedure DIJKSTRA(G, w, s)
2:     INIT-SINGLE-SOURCE(G, s)
3:     S = ∅
4:     Q = G.V
5:     while Q ≠ ∅ do
6:         u ← EXTRACT-MIN(Q)
7:         S ← S ∪ {u}
8:         for v ∈ G.Adj[u] do
9:             RELAX(u, v, w)
```

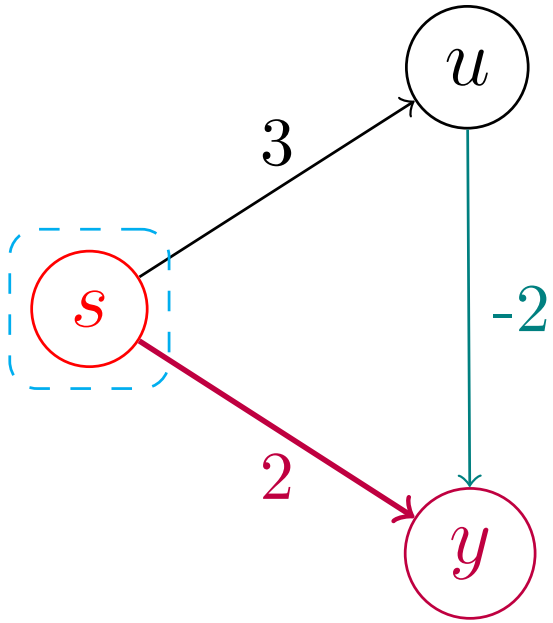
Array: $O(n^2)$

Min-heap: $O(E \log V)$

Fib-heap: $O(V \log V + E)$


u
z
S
s
w ≥ 0
x
y

Negative-weight Edges for Dijkstra's Algorithm (Problem 24.3-2)

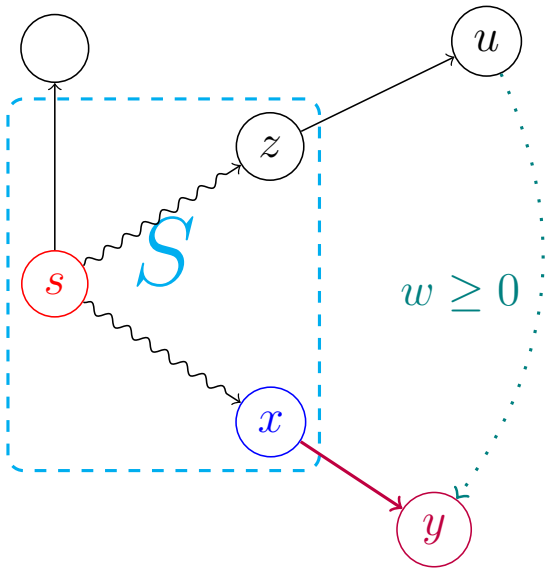


Negative-weight Edges for Dijkstra's Algorithm (Additional Problem 24.3-10)

- All negative-weight egdes are from s
- No negative-weight cycles

Negative-weight Edges for Dijkstra's Algorithm (Additional Problem 24.3-10)

- All negative-weight egdes are from s
- No negative-weight cycles



Checking Output of Dijkstra's Algorithm (Problem 24.3-4)

Dijkstra's Algorithm on Digraph with Nonnegative-weight Edges

Lawler's Algorithm on DAG

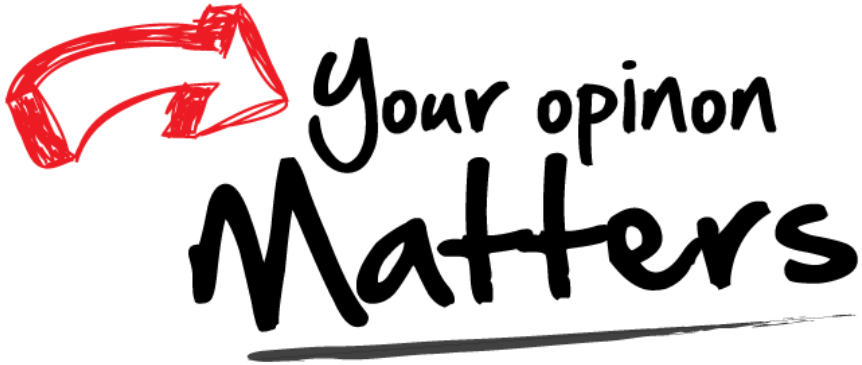
$\Longleftarrow$

Dijkstra's Algorithm on Digraph with Nonnegative-weight Edges

$\Longrightarrow$

Bellman-Ford Algorithm on Digraph with Negative-weight Edges


THANK YOU



Office 302

Mailbox: H016

hfwei@nju.edu.cn