Paths in Graphs

Hengfeng Wei

hfwei@nju.edu.cn

June 03 – June 05, 2017

Paths in Graphs

1. Dijkstra's Algorithm for SSSP
2. Dijkstra's Algorithm as Framework
3. Bellman-Ford and Floyd-Warshall Algorithms
4. Miscellaneous

Dijkstra's algorithm for SSSP

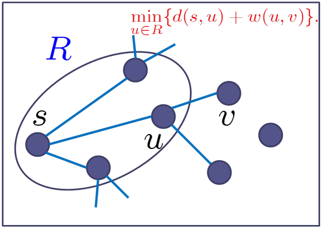
$$R \triangleq \{v \mid s \rightsquigarrow v \text{ is known}\}$$

Finding shortest paths from s to other nodes t
in increasing order of $\text{dist}(s, t)$.

Theorem (Invariant)

$$\exists d : \begin{cases} \textit{dist}(s, v) \leq l, & \forall v \in R, \\ \textit{dist}(s, v) > l, & \forall v \notin R \end{cases}$$

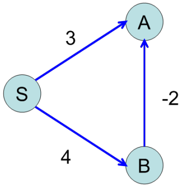
Dijkstra's algorithm for SSSP



Negative edges

Negative edges (Problem 6.16)

Dijkstra's algorithm may fail if $w(e) < 0$.



Negative edges

Negative edges from s (Problem 6.21)

All negative edges are from s.

$$\arg\min_{(s,v)} w(s,v)$$

Generalized shortest path problem

Generalized shortest path problem (Problem 6.20)

- digraph $G = (V, E), l_e > 0, c_v > 0, s \in V$
- shortest paths from s

$$l'(u, v) = l(u, v) + c_v$$

Shortest paths among nodes

Shortest paths among nodes (Problem 6.27)

- $G = (V, E), w(e) \geq 0$
- $S, T \subseteq V$
- compute $\min\{\mathsf{d}(s,t) : s \in S, t \in T\}$

$$V' = V + \{s_0, t_0\}$$

Shortest path through v_0

Shortest paths through v_0 (Problem 6.28)

- strongly connected digraph $G = (V, E), w(e) > 0$
- $v_0 \in V$
- find shortest paths $s \leadsto^{\text{SP}} t$ through v_0

$$s \leadsto^{\text{SP}} v_0 \leadsto^{\text{SP}} t$$

$$\# v_0' s = 1$$

Shortest path in maze

Shortest paths in maze (Problem 6.24)

1. $w(e) = c > 0$ without obstacles
2. $w(e) = c > 0$ with obstacles
3. $w(e) > 0$
   - 3.1 $\rightarrow, \downarrow$; in $O(n+m)$
   - 3.2 $\uparrow, \downarrow, \leftarrow, \uparrow$
4. $\exists e : w(e) < 0$ without negative cycles

$$(3.1)\ \mathsf{d}[v] = \min_{u \to v} \mathsf{d}[u] + w(u \to v)$$

Compute dist$[v]$ in topo. order.

1 Dijkstra's Algorithm for SSSP

2 Dijkstra's Algorithm as Framework

3 Bellman-Ford and Floyd-Warshall Algorithms

4 Miscellaneous

Dijkstra's algorithm

$$\text{dist}(v) = \min_{u \in N(v)} \{\text{dist}(u) + l(u, v)\}$$

for all $v \in V$ **do**
  $\text{dist}[v] \leftarrow \infty$
$\text{dist}[s] \leftarrow 0$

$Q \leftarrow \text{MinPQ}(V)$

while $Q \neq \emptyset$ **do**
  $u \leftarrow \text{deleteMin}(Q)$
  for all $(u, v) \in E \land v \in Q$ **do**
    if $\text{dist}[v] > \text{dist}[u] + l(u, v)$ **then**
      $\text{dist}[v] \leftarrow \text{dist}[u] + l(u, v)$
      decreaseKey(Q, v)

$$O(n + (n + m)\log n) \implies O((n + m)\log n) \implies O(m \log n)$$

Dijkstra's algorithm

Prim's algorithm for MST:

if $\text{cost}[u] > w(u, v)$ **then**
$\quad \text{cost}[u] \leftarrow w(u, v)$

BFS:

$Q \leftarrow \text{FIFO-Q}(s)$

if $\text{dist}[v] = \infty$ **then**
$\quad \text{dist}[v] \leftarrow \text{dist}[u] + 1$

Unique shortest paths

Unique shortest paths (Problem 6.18)

Undirected graph $G = (V, E), w(e) > 0, s \in V$:

$$\mathsf{usp}[v] = T \iff \exists! s \leadsto^{\mathsf{SP}} v$$

for all $v \in V$ **do**
  $\mathsf{usp}[v] \leftarrow F$
$\mathsf{usp}[s] \leftarrow T$

if $\mathsf{dist}[v] > \mathsf{dist}[u] + l(u, v)$ **then**
  $\mathsf{dist}[v] \leftarrow \mathsf{dist}[u] + l(u, v)$
  $\mathsf{usp}[v] \leftarrow \mathsf{usp}[u]$
else if $\mathsf{dist}[v] = \mathsf{dist}[u] + l(u, v)$ **then**
  $\mathsf{usp}[v] \leftarrow F$

Number of shortest paths

Number of shortest paths (Problem 6.31, 5.26)

$$\# s \rightsquigarrow^{\mathsf{SP}} v$$

$$O(n+m) \textit{ if } w(e) = 1$$

Min-max path problem

Min-max path problem (Problem 6.23)

- $G = (V, E)$: network of highways
- l_e: road length; L: tank capacity
- (1) Given L, $\exists ? s \leadsto t$.
- (2) Given G, compute $\min L$.

$$L[v] = \min_{u \in N(v)} \max\{L[u], l(u, v)\}$$

for all $v \in V$ **do**
  $L[v] \leftarrow \infty$
$L[s] \leftarrow 0$

if $L[v] > \max(L[u], l(u, v))$ **then**
  $L[v] \leftarrow \max(L[u], l(u, v))$

Min-max path problem

Min-max path problem (Problem 6.23)

- $G = (V, E)$: network of highways
- l_e: road length; L: tank capacity
- (1) Given L, $\exists ? s \leadsto t$.
- (2) Given G, compute $\min L$.

$O(\log m)$ binary searches for $\min L$

Max-min path problem

Max-min path problem (Problem 6.26)

- $G = (V, E)$: network of oil pipelines
- $c(u, v)$: capacity of (u, v)
- (1) Given s, compute cap(s, v).
- (2) Compute all-pair cap(u, v).

$$\text{cap}[v] = \max_{u \in N(v)} \min(\text{cap}[u], c(u, v))$$

$$Q \leftarrow \text{MaxPQ}(V)$$

Paths in Graphs

1 Dijkstra's Algorithm for SSSP

2 Dijkstra's Algorithm as Framework

3 Bellman-Ford and Floyd-Warshall Algorithms

4 Miscellaneous

Bellman-Ford algorithm

Bellman-Ford algorithm (Problem 6.30)

- digraph $G = (V, E), \exists e : w(e) < 0$
- $\forall s, t : |s \rightsquigarrow^{\mathsf{SP}} t| \leq k$
- Given s, t, find $s \rightsquigarrow^{\mathsf{SP}} t$.

$\mathsf{d}(v, k)$: shortest path distance from s to v using $\leq k$ edges

$$s \rightsquigarrow^{\mathsf{SP}; \leq k-1} u \to^{=1} v$$

$$\mathsf{d}(v, k) = \min_{u \in N(v)} \mathsf{d}(u, k-1) + w(u, v)$$

for all $i = 1 \to n - 1$ **do**
  for all $e \in E$ **do**
    update e

Floyd-Warshall algorithm

$$\text{dist}(i, j, k) = \min(\text{dist}(i, j, k-1), \text{dist}(i, k, k-1) + \text{dist}(k, j, k-1))$$

$$\#k's = 1 \implies \text{dist}(i, k, k-1)$$

Routing table

Routing table (Problem 6.25)

Contruct routing table and extract shortest paths from it.

Init: $\text{Go}(i,j) \leftarrow \text{Null}$

$\forall (i,j) \in E : \text{Go}(i,j) \leftarrow j$

if $\ldots$ **then**
 $\text{Go}(i,j) \leftarrow \text{Go}(i,k)$

if $Go(i,j) = \text{Null}$ **then**
 $\ldots$

while $i \neq j$ **do**
 $i \leftarrow \text{Go}(i,j)$

$$\text{Prev}(i,j) \leftarrow \text{Prev}(k,j)$$

$$\text{Intermediate}(i,j) \leftarrow k$$

Shortest cycle in digraph

Shortest cycle in digraph

Find shortest cycle in digraph $G = (V, E), w(e) > 0$.

Initialize $\text{dist}[v][v] \leftarrow \infty$ in Floyd-Warshall algorithm

$\exists v : \text{dist}[v][v] < 0$ *vs.* $\forall v : \text{dist}[v][v] = \infty$

Max-min path problem

Max-min path problem (Problem 6.26)

- $G = (V, E)$: network of oil pipelines
- $c(u, v)$: capacity of (u, v)
- (2) Compute all-pair $\mathsf{cap}(u, v)$.

$$\mathsf{cap}(u, v, k) = \max(\mathsf{cap}(u, v, k-1), \min(\mathsf{cap}(u, k, k-1), \mathsf{cap}(k, v, k-1)))$$

Paths in Graphs

1 Dijkstra's Algorithm for SSSP

2 Dijkstra's Algorithm as Framework

3 Bellman-Ford and Floyd-Warshall Algorithms

4 Miscellaneous

Hamiltonian path in Tournament graph

Hamiltonian path in Tournament graph (Problem 6.22)

$$\begin{aligned}\forall u, v : (u \to v \lor v \to u) \\ \land \neg(u \to v \land v \to u)\end{aligned}$$

By mathematical induction on n.


THANK YOU