CMPSCI 611: Advanced Algorithms

Lecture 9: Dijkstra's Algorithm and Seidel's Algorithm

Andrew McGregor

Last Compiled: October 7, 2009

Outline

Dijkstra's Algorithm

Seidel's Algorithm

Shortest Paths

Let $G = (V, E)$ be a directed graph with weights $w : E \longrightarrow \mathbb{R}^+$.

Definition

For path $p = (v_1, \ldots, v_k)$ be a path, define

$$w(p) = \sum_{i=1}^{k-1} w(v_i, v_{i+1}) \ .$$

The *shortest path* between u and v is

$$\delta(u, v) = \min\{w(p) : p \text{ is a path from } u \text{ to } v\}$$

if there is a path from u to v and ∞ otherwise.

Dijkstra's Warm-Up

Single-Source Problem: Given $s \in V$, find $\delta(s, v)$ for all $v \in V$.

Dijkstra's algorithm solves problem if all edges are non-negative:

- Maintains array $(d[v] : v \in V)$ and we'll prove the invariant:

$$d[v] \geq \delta(s, v)$$

- Maintains a set of processed vertices R and unprocessed vertices Q. We'll prove that for all $v \in R$:

$$d[v] = \delta(s, v)$$

Dijkstra's Algorithm

Algorithm

1. $d[s] = 0$ *and for* $s \neq v$*:*

$$d[v] = w(s,v) \text{ if } (s,v) \in E \text{ and } \infty \text{ otherwise}$$

2. $R = \{s\}$, $Q = V - \{s\}$
3. *While* $|Q| \geq 1$*:*
   3.1 *Pick* $u = \operatorname{argmin}_{v \in Q} d[v]$
   3.2 $R = R + u$, $Q = Q - u$
   3.3 *For each* v *with* $(u,v) \in E$*:*

$$d[v] = \min(d[u] + w(u,v), d[v])$$

Running Time: $O(|V|^2)$ for simple implementation but can be improved to $O(|E| + |V| \log |V|)$ using Fibonacci heaps.

Correctness of Algorithm

The correctness of the algorithm follows because $d[u]$ never increases and appealing to the following lemmas:

Lemma

Throughout the algorithm, $d[v] \geq \delta(s, v)$ *for all* v.

Lemma

When u *is added to* R, $d[u] = \delta(s, u)$

The $d[v]$'s are never too small...

Lemma

Throughout the algorithm, $d[v] \geq \delta(s, v)$ for all v.

Proof.

- By contradiction: Let v is the first vertex such that at some point in the algorithm, $d[v] < \delta(s, v)$.
- Consider when v updated:

$$d[v] = d[u] + w(u, v) \text{ for some } u \in R$$

- But then $d[u] + w(u, v) = d[v] < \delta(s, v)$
- $\delta(s, v) \leq \delta(s, u) + w(u, v)$
- Hence $d[u] < \delta(s, u)$ and v wasn't first vertex to go "bad"

□

When u get's add to R, $d[u]$ is correct... (1/2)

Let $d_u[v]$ be value of $d[v]$ at start of iteration when u is chosen as minimum.

Lemma

For all u, $d_u[u] = \delta(s, u)$

- By contradiction: Let u be first vertex placed in R such that $d_u[u] > \delta(s, u)$
- Let P be a shortest path from s to u. Let y be first vertex on P that is not in R and let x be predecessor of y
- Claim: $d_u[y] = \delta(s, y)$
  - $d_x[x] = \delta(s, x)$ by assumption that u is first bad vertex.
  - After iteration where x is added to R: $d[y] \leq \delta(s, x) + w(x, y)$
  - $\delta(s, x) + w(x, y) = \delta(s, y)$ since P included shortest path to y

When u get's add to R, $d[u]$ is correct... (2/2)

- By contradiction: Let u be first vertex placed in R such that $d_u[u] > \delta(s, u)$
- Let P be a shortest path from s to u. Let y be first vertex on P that is not in R and let x be predecessor of y
- Claim: $d_u[y] = \delta(s, y)$
- Since y lies on shortest path to u: $\delta(s, y) \leq \delta(s, u)$
- Putting above two lines together:

$$d_u[y] = \delta(s, y) \leq \delta(s, u) < d_u[u]$$

- But because u was the next minimum: $d_u[u] \leq d_u[y]$
- Contradiction!

Outline

Dijkstra's Algorithm

Seidel's Algorithm

Seidel's Algorithm

Problem: For an undirected, unweighted graph G, compute $\delta_G(i,j)$ for all $i, j \in V$.

Seidel's Algorithm is based on matrix multiplication and runs in time

$$O(\mu(n) \log n)$$

where $\mu(n)$ is the time to multiply two $n \times n$ matrices together. Recall

$$n^2 \leq \mu(n) \leq n^{2.38}$$

The G_2 graph

Definition

Given a undirected, unweighted graph $G = (V, E)$, define $G_2 = (V, E')$ where $(i, j) \in E'$ iff $\delta_G(i, j) \leq 2$.

Lemma

Let $P_G(i, j) = 1$ if $\delta_G(i, j)$ is odd and $P_G(i, j) = 0$ otherwise. Then,

$$\delta_G(i, j) = 2\delta_{G_2}(i, j) - P_G(i, j)$$

For Next Time...

- Finish reading up Section 4 of the notes.
- Homework 2 to be unleashed later today.