Chapter 16

Network Flow V - Min-cost flow

CS 573: Algorithms, Fall 2013
October 22, 2013

16.1 Minimum Average Cost Cycle

16.1.0.1 Minimum Average Cost Cycle

(A) $\mathsf{G} = (\mathsf{V}, \mathsf{E})$: a ***digraph***, n vertices, m edges.
(B) $\omega : \mathsf{E} \to \mathbb{R}$ weight on the edges.
(C) ***directed cycle***: closed walk $\mathsf{C} = (v_0, v_1, \ldots, v_t)$, where $v_t = v_0$ and $(v_i \to v_{i+1}) \in \mathsf{E}$, for $i = 0, \ldots, t-1$.
(D) ***average cost of a directed cycle*** is $\mathrm{AvgCost}(\mathsf{C}) = \omega(C)/t = \left(\sum_{e \in C} \omega(e)\right)/t$.
(E) $d_k(v)$: min length of walk with exactly k edges, ending at v
(F) $d_0(v) = 0$ and $d_{k+1}(v) = \min_{e=(u \to v) \in E} \Big(d_k(u) + \omega(e) \Big)$.
(G) Compute $d_i(v)$, for $\forall i, \forall v \in \mathsf{V}$.
In $O(nm)$ time using dynamic programming.

16.1.0.2 Computing the Min-Average Cost cycle

Cost of ***minimum average cost cycle*** is

$$\mathrm{MinAvgCostCycle}(\mathsf{G}) = \min_{\mathsf{C} \text{ is a cycle in } \mathsf{G}} \mathrm{AvgCost}(\mathsf{C})$$

Theorem 16.1.1. *The minimum average cost of a directed cycle in* G *is equal to*

$$\alpha = \min_{v \in V} \max_{k=0}^{n-1} \frac{d_n(v) - d_k(v)}{n-k}.$$

Namely, $\alpha = \mathrm{MinAvgCostCycle}(\mathsf{G})$.

16.1.0.3 Proof

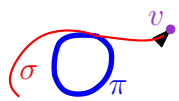
Proof
(A) Adding r to weight of every edge increases the average cost of a cycle $\mathrm{AvgCost}(\mathsf{C})$ by r.
(B) α also increases by r.
(C) Assume price of min. average cost cycle $= 0$.
(D) ... all cycles have non-negative (average) cost.
(E) Prove: $\mathrm{MinAvgCostCycle}(\mathsf{G}) = 0 \implies \alpha = 0$.
(Implies theorem by shifting prices by r).

16.1.1 Proof continued

16.1.1.1 MinAvgCostCycle(G) $= 0 \implies \alpha \geq 0$

Proof continued

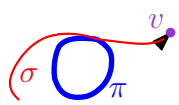
(A) $\alpha = \min_{u \in V} \beta(u)$, where $\beta(u) = \max_{k=0}^{n-1} \dfrac{d_n(u) - d_k(u)}{n-k}$.
(B) Assume α realized by vertex v; $\alpha = \beta(v)$.
(C) P_n: n edges walk ending at v, of length $d_n(v)$.
(D) P_n must contain a cycle.
(E) Break P_n: a cycle π (length $n-k$) and path σ (length k).



(F) $d_n(v) = \omega(P_n) = \omega(\pi) + \omega(\sigma) \geq \omega(\sigma) \geq d_k(v)$,

16.1.2 Proof continued

16.1.2.1 Continue proving: MinAvgCostCycle(G) $= 0 \implies \alpha \geq 0$



(A) $\omega(\pi) \geq 0$: since π is cycle + by assumption $\forall$ cycle cost ≥ 0.
(B) $\implies d_n(v) - d_k(v) \geq 0$. As such, $\frac{d_n(v) - d_k(v)}{n-k} \geq 0$. Let

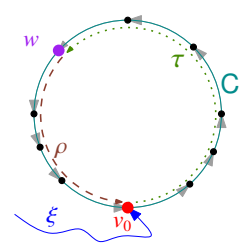
$$\beta(v) = \max_{j=0}^{n-1} \frac{d_n(v) - d_j(v)}{n-j} \geq \frac{d_n(v) - d_k(v)}{n-k} \geq 0.$$

Now, $\alpha = \beta(v) \geq 0$, by the choice of v.
(C) QED for this direction.

16.1.3 Proof for other direction

16.1.3.1 MinAvgCostCycle(G) $= 0 \implies \alpha \leq 0$:

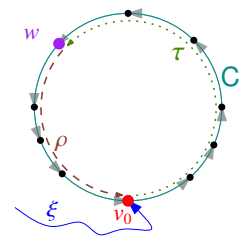
(A) $\mathsf{C} = (v_0, v_1, \ldots, v_t)$: directed cycle of weight 0.
(B) $\min_{j=0}^{\infty} d_j(v_0)$ realized by index $r < n$.
(Otherwise remove non-negative cycles.)
(C) ξ = walk of length r ending at v_0.
(D) $w \in \mathsf{C}$ = walk $n-r$ edges on C from v_0.
(E) τ is this walk (i.e., $|\tau| = n - r$).
(F) $d_n(w) \leq \omega\Big(\xi \mathbin{||} \tau\Big) = d_r(v_0) + \omega(\tau)$,
(G) ρ: walk on C from w back to v_0.
(H) $\tau \mathbin{||} \rho$ goes around C several times.
(I) $\omega(\tau \mathbin{||} \rho) = 0$, as $\omega(\mathsf{C}) = 0$.



16.1.4 Proof for other direction

16.1.4.1 MinAvgCostCycle(G) $= 0 \implies \alpha \leq 0$: continued

(A) For any k: extend k edges shortest path ending at w to a path to v_0 (concatenating ρ)
(B) $d_k(w) + \omega(\rho) \geq d_{k+|\rho|}(v_0) \geq d_r(v_0) \geq d_n(w) - \omega(\tau)$,
(C) $\omega(\rho) \geq d_n(w) - \omega(\tau) - d_k(w)$.
(D) $0 = \omega(\tau \mid\mid \rho) = \omega(\rho) + \omega(\tau) \geq \Big(d_n(w) - \omega(\tau) - d_k(w)\Big) + \omega(\tau)$
$= d_n(w) - d_k(w)$
(E) $\implies \beta(w) = \max_{k=0}^{n-1} \frac{d_n(w) - d_k(w)}{n-k} \leq 0$.
(F) $\alpha = \min_{v \in V(\mathsf{G})} \beta(v) \leq \beta(w) \leq 0$
(G) $\implies \alpha = 0$.



QED

16.1.4.2 Computing α:

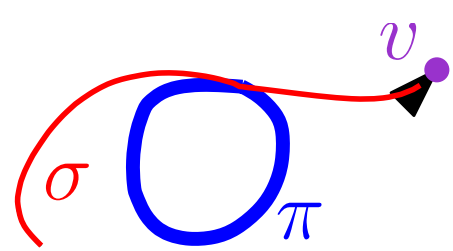
(A) $\forall k, \forall v$ $d_k(v)$: longest path with k edges ending at v. Computed in $O(nm)$ time.
(B) $\alpha = \min_{v \in V} \max_{k=0}^{n-1} \frac{d_n(v) - d_k(v)}{n-k}$.
(C) Compute α in $O(n^2)$ after $d_i(\cdot)$ computed.

16.1.4.3 Finding min average cost cycle...

(A) Proved: Minimum avg cost of cycle in G is $= \alpha = \min_{v \in V} \max_{k=0}^{n-1} \frac{d_n(v) - d_k(v)}{n-k}$.
(B) Compute v that realizes α.
(C) Add $-\alpha$ to all the edges in the graph.
(D) Looking for cycle of weight 0.
(E) Recompute $d_i(\cdot)$ to agree with the new weights of the edges.
(F) For v above: $0 = \alpha = \max_{k=0}^{n-1} \frac{d_n(v) - d_k(v)}{n-k}$
(G) $\implies \forall k \in \{0, \ldots, n-1\}$ $\frac{d_n(v) - d_k(v)}{n-k} \leq 0$
(H) $\implies \forall k \in \{0, \ldots, n-1\}$ $d_n(v) - d_k(v) \leq 0$.
(I) $\implies \forall i$ $d_n(v) \leq d_i(v)$, for all i.

16.1.4.4 Finding min average cost cycle...

(A) Repeat proof of theorem...
(B) P_n: path with n edges realizing $d_n(v)$.
(C) $P_n = \sigma || \pi$
σ : a path of length k, π is a cycle.
(D) $\omega(\pi) \geq 0$
(E) $\omega(\sigma) \geq d_k(v)$
(F) $\omega(\pi) = d_n(v) - \omega(\sigma) \leq d_n(v) - d_k(v) \leq 0$
(G) π is a cycle and $\omega(\pi) = 0$. Done!
(H) Note - the reweighting is not really necessary.



16.1.4.5 Finding min average cost cycle...

Corollary 16.1.2. *A direct graph G with n vertices and m edges, and a weight function $\omega(\cdot)$ on the edges, one can compute the cycle with minimum average cost in $O(nm)$ time.*

16.2 Potentials

16.2.0.6 Shortest path with negative weights...

(A) Dijkstra algorithm works only for graphs with non-negative weights.
(B) If negative weights, then one can use the Bellman-Ford algorithm.
(C) Bellman-Ford is slow... $O(mn)$.
(D) Show how to use Dijkstra algorithm for some cases.
(E) $\mathsf{G} = (\mathsf{V}, \mathsf{E})$ with weight $w(\cdot)$ on edges.
(F) $\mathbf{d}_\omega(s,t)$: length of shortest path.
(G) Weights might be negative!

16.2.0.7 Potential

A ***potential*** $p(\cdot)$ is a function that assigns a real value to each vertex of G, such that if $e = (u \to v) \in \mathsf{G}$ then $w(e) \geq p(v) - p(u)$.

16.2.0.8 Lemma (i)

Lemma 16.2.1. *$\exists p(\cdot)$ potential for $\mathsf{G} \iff \mathsf{G}$ has no negative cycles (for $w(\cdot)$).*

Proof:
$\Rightarrow$: Assume $\exists p(\cdot)$ potential. For any cycle C:

$$w(\mathsf{C}) = \sum_{(u \to v) \in \mathsf{E}(\mathsf{C})} w(e) \geq \sum_{(u \to v) \in \mathsf{E}(\mathsf{C})} (p(v) - p(u)) = 0.$$

$\Leftarrow$: Assume no negative cycle. $p(v)$: shortest walk that ends at v.
Claim: $p(v)$ is a potential.
(A) No negative cycles: $p(v)$ is well defined.
(B) $\forall (u \to v) \in \mathsf{E}(\mathsf{G})$: $p(v) \leq p(u) + w(u \to v)$
(C) $p(v) - p(u) \leq w(u \to v)$, as required. ■

16.2.0.9 Lemma (ii)

Lemma 16.2.2. *$p(\cdot)$: potential. $\forall e = (u \to v) \in \mathsf{E}(\mathsf{G})$: $\ell(e) = w(e) - p(v) + p(u)$*
(A) $\ell(\cdot)$ is non-negative for all edges.
(B) $\forall s, t \in \mathsf{V}(\mathsf{G})$: shortest path π of $\mathbf{d}_\ell(s,t)$ also s.p. $\mathbf{d}_\omega(s,t)$.

Proof:
Proof of (A): $w(e) \geq p(v) - p(u) \implies w(e) - p(v) + p(u) \geq 0$.
Proof of (B): $\forall$ $s - t$ path π in G: $\ell(\pi) = \sum_{e=(u \to v) \in \pi} (w(e) - p(v) + p(u)) = w(\pi) + p(s) - p(t)$,
$\implies \mathbf{d}_\ell(s,t) = \mathbf{d}_\omega(s,t) + p(s) - p(t)$.

■

16.2.0.10 Lemma (iii)

Lemma 16.2.3. *G: graph. $p(\cdot)$: potential.*
Compute the shortest path from s to all vertices of G in $O(n \log n + m)$ time, where G has n vertices and m edges

Proof:
(A) Use Dijkstra algorithm on the distances defined by $\ell(\cdot)$.
(B) The shortest paths are preserved under this distance by Lemma (ii), and this distance function is always positive.

■

16.3 Minimum cost flow

16.3.0.11 Min cost flow

Input:

$\mathsf{G} = (\mathsf{V}, \mathsf{E})$: directed graph.
s: source.
t: sink
$\mathsf{c}(\cdot)$: capacities on edges,
ϕ: Desired amount (***value***) of flow.
$\kappa(\cdot)$: Cost on the edges.

Definition - cost of flow ***cost*** of flow f: $\mathrm{cost}(\mathsf{f}) = \sum_{e \in E} \kappa(e) * \mathsf{f}(e)$.

16.3.0.12 Min cost flow problem

Min-cost flow ***minimum-cost s-t flow problem***: compute the flow f of min cost that has value ϕ.

min-cost circulation problem Instead of ϕ we have lower-bound $\ell(\cdot)$ on edges.
(All flow that enters must leave.)

Claim 16.3.1. *If we can solve min-cost circulation $\implies$ can solve min-cost flow.*

HERE: All demands on vertices are zero!

16.3.0.13 Residual graph...

The ***residual graph*** of f is the graph $\mathsf{G}_{\mathsf{f}} = (V, E_{\mathsf{f}})$ where

$$E_{\mathsf{f}} = \left\{ e = (u \to v) \in V \times V \;\middle|\; \begin{array}{c} \mathsf{f}(e) < \mathsf{c}(e) \\ \text{or } \mathsf{f}\left(e^{-1}\right) > \ell\left(e^{-1}\right) \end{array} \right\}.$$

where $e^{-1} = (v \to u)$ if $e = (u \to v)$.

Assumption 16.3.2. $\forall u, v \quad (u \to v) \in \mathsf{E}(\mathsf{G}) \implies (v \to u) \notin \mathsf{E}(\mathsf{G})$.

Cost function is anti-symmetric:

$$\forall (u \to v) \in E_{\mathsf{f}} \qquad \kappa\Big((u \to v)\Big) = -\kappa\Big((v \to u)\Big).$$

16.3.0.14 Some definitions

Definition 16.3.3. *Cycle sign Directed cycle C in G_{f}.*

$e = (u \to v) \in \mathsf{E}(\mathsf{G})$: $\chi_C(e) = \begin{cases} 1 & e \in C \\ -1 & e^{-1} = (v \to u) \in C \\ 0 & \textit{otherwise;} \end{cases}$

Pay 1 if e is in C and -1 if we travel e in the "wrong" direction.

Definition 16.3.4. *Cycle cost The* **cost** *of a directed cycle C in G_{f} is*

$$\kappa(\mathsf{C}) = \sum_{e \in \mathsf{C}} \kappa(e).$$

16.3.0.15 Even more definitions

(A) Circulation comply with capacity and lower-bounds constraints is ***valid***.
(B) flow function that only comply with conservation property is a ***weak circulation***.
(C) Weak circulation might violate capacity and lower bounds.
(D) Weak circulation might not be a valid circulation.

16.3.0.16 Another lemma

Lemma 16.3.5. f, g*: two valid circulations in* $\mathsf{G} = (\mathsf{V}, \mathsf{E})$*. Let* $\mathsf{h} = \mathsf{g} - \mathsf{f}$*.*
(A) h *is a weak circulation,*
(B) if $\mathsf{h}(u \to v) > 0$ *then* $(u \to v) \in \mathsf{G}_{\mathsf{f}}$*.*

Proof...
(A) h is clearly a weak circulation (conservation of flow - verify).
(B) If $\mathsf{h}(u \to v)$ is negative, then $\mathsf{h}(v \to u) = -\mathsf{h}(u \to v)$.
(C) For $e = (u \to v)$, $\mathsf{h}(u \to v) > 0$:
  (i) If $e = (u \to v) \in E$, and $\mathsf{f}(e) < \mathsf{c}(e) \implies e \in \mathsf{G}_{\mathsf{f}}$.
  If $\mathsf{f}(e) = \mathsf{c}(e) \implies \mathsf{h}(e) = \mathsf{g}(e) - \mathsf{f}(e) \leq 0$. Contradicts $\mathsf{h}(u \to v) > 0$.

16.3.0.17 Proof continued...

Proof continued: For $e = (u \to v)$, $\mathsf{h}(u \to v) > 0$, and $e = (u \to v) \notin \mathsf{E}$:
(A) $\implies e^{-1} = (v \to u) \in \mathsf{E}$. Otherwise $\mathsf{h}(u \to v) = 0$.
(B) $0 > \mathsf{h}\left(e^{-1}\right) = \mathsf{g}\left(e^{-1}\right) - \mathsf{f}\left(e^{-1}\right)$.
(C) $\implies \mathsf{f}\left(e^{-1}\right) > \mathsf{g}\left(e^{-1}\right) \geq \ell\left(e^{-1}\right)$.
(D) Flow by f on e^{-1} larger than lower bound.
(E) Can return this flow in the other direction.
(F) $\implies e \in \mathsf{G}_{\mathsf{f}}$. ■

Bibliography

A. V. Goldberg and R. E. Tarjan. Finding minimum-cost circulations by canceling negative cycles. *J. Assoc. Comput. Mach.*, 36(4):873–886, 1989.

R. M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Math.*, 23:309–311, 1978.

A. Schrijver. *Combinatorial Optimization : Polyhedra and Efficiency (Algorithms and Combinatorics)*. Springer, July 2004. ISBN 3540204563.

É. Tardos. A strongly polynomial minimum cost circulation algorithm. *Combinatorica*, 5(3):247–255, 1985. ISSN 0209-9683.