12 Minimum Cost Flow Problem

12.1 Transshipments

b-Transshipments and Costs

Given: Digraph $D = (V, A)$, capacities $u : A \longrightarrow \mathbb{R}_{\geq 0}$, arc costs $c : A \longrightarrow \mathbb{R}$.

Definition 12.1

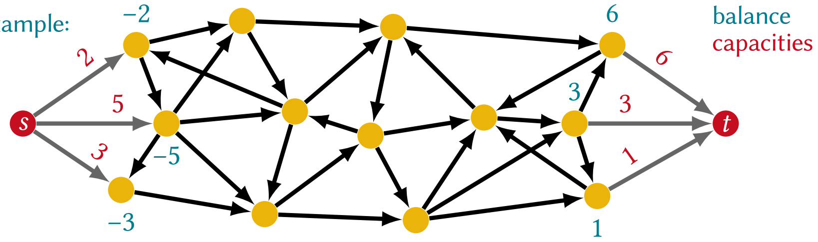
i Let $b : V \longrightarrow \mathbb{R}$. A flow x is called b-transshipment if

$$\mathrm{ex}_x(v) = b(v) \qquad \text{for all } v \in V.$$

ii The cost of a flow x is defined as $c(x) := \sum_{a \in A} c(a) \cdot x(a)$.

Observation 12.2 A feasible b-transshipment can be found by a max-flow computation.

Example:



Minimum-Cost b-Transshipment Problem

Remark. The existence of a b-transshipment implies that $\sum_{v \in V} b(v) = 0$.

Minimum-cost b-transshipment problem

Given: $D = (V, A)$, $u : A \longrightarrow \mathbb{R}_{\geq 0}$, $c : A \longrightarrow \mathbb{R}$, $b : V \longrightarrow \mathbb{R}$

Task: find a feasible b-transshipment of minimum cost

Special cases:

- min-cost s-t-flow problem (for given flow value)
- min-cost circulation problem

Cost of residual arc:

For a given feasible flow x, we extend the cost function c to A_x by defining

$$c(a^{-1}) := -c(a) \qquad \text{for } a \in A.$$

Adding and Subtracting Flows

- we generalize the observations on adding and subtracting flows (Slide 10|8)
- If x is a feasible flow in (D, u) and y a feasible flow in (D_x, u_x), then

$$z(a) := x(a) + y(a) - y(a^{-1}) \quad \text{for } a \in A$$

yields a feasible flow z in D ("$z := x + y$") and $c(z) = c(x) + c(y)$.

Notice that $\text{ex}_z(v) = \text{ex}_x(v) + \text{ex}_y(v)$ for each $v \in V$.

- If x, z are feasible flows in (D, u), then

$$y(a) := \max\{0, z(a) - x(a)\} \qquad \text{for } a \in A \cap A_x,$$
$$y(a^{-1}) := \max\{0, x(a) - z(a)\} \qquad \text{for } a^{-1} \in A^{-1} \cap A_x,$$

yields a feasible flow y in D_x ("$y := z - x$") and $c(y) = c(z) - c(x)$.

Notice that $\text{ex}_y(v) = \text{ex}_z(v) - \text{ex}_x(v)$ for each $v \in V$.

- the following generalizes Theorem 10.15 towards transshipments and costs

Theorem 12.3 For a b-transshipment x in D, there exist dipaths $P_1, \dots, P_k$, where each P_i starts at a node v with $b(v) < 0$ and ends at a node w with $b(w) > 0$, as well as dicycle $C_1, \dots, C_\ell$ in D with $k + \ell \le m + n$ and values $y_{P_1}, \dots, y_{P_k}, y_{C_1}, \dots, y_{C_\ell} \ge 0$ with

$$x(a) = \sum_{i\,:\,a \in P_i} y_{P_i} + \sum_{j\,:\,a \in C_j} y_{C_j} \qquad \text{for all } a \in A. \tag{$\star$}$$

- Theorem 12.3 implies

$$\begin{aligned} c(x) = \sum_{a \in A} x(a) \cdot c(a) &\overset{(\star)}{=} \sum_{a \in A} \left(\sum_{i\,:\,a \in P_i} y_{P_i} + \sum_{j\,:\,a \in C_j} y_{C_j} \right) \cdot c(a) \\ &= \sum_{i=1}^{k} y_{P_i} \cdot c(P_i) + \sum_{j=1}^{\ell} y_{C_j} \cdot c(C_j), \end{aligned}$$

where $c(P_i) = \sum_{a \in P_i} c(a)$ and $c(C_j) = \sum_{a \in C_j} c(a)$.

Theorem 12.4 A feasible b-transshipment x has minimum cost among all feasible b-transshipments if and only if each dicycle of D_x has nonnegative cost.

Proof: "$\Longrightarrow$"

- let C be a dicycle in D_x, then $y := \delta\chi^C$ with $\delta := \min_{a\in C} u_x(a)$ is a feasible circulation in D_x
- then $z := x + y$ is a feasible b-transshipment in D
- $c(x) \leq c(z) = c(x) + c(y)$ as x has minimum cost, thus, $c(y) \geq 0$

"$\Longleftarrow$"

- let z be an arbitrary b-transshipment in D
- $y := z - x$ is a feasible circulation in D_x with $c(y) = c(z) - c(x)$
- consider decomposition of y into flow along dicycles $C_1, \ldots, C_\ell$ in D_x, i.e,

$$y(a) := \sum_{j\,:\,a\in C_j} y_{C_j} \quad \text{for some } y_{C_1}, \ldots, y_{C_\ell} \in\in \mathbb{R}_{\geq 0}$$

- $c(y) = \sum_{j=1}^{\ell} y_{C_j} c(C_j) \geq 0$, so $c(z) \geq c(x)$ and x has minumum cost □

Theorem 12.5 A feasible b-transshipment x has minimum cost among all feasible b-transshipments if and only if there is a feasible potential $y \in \mathbb{R}^V$ in D_x, that is,

$$y_v + c\big((v, w)\big) \geq y_w \qquad \text{for all } (v, w) \in A_x.$$

Proof: x has minimum cost

$\iff$ D_x contains no negative cost dicycle (by Theorem 12.4)

$\iff$ D_x has a feasible potential (by Theorem 8.12) □

Alternative Proof of Theorem 12.5

- consider LP formulation of min-cost b-transshipment problem

Primal LP:

$$\begin{aligned} \min \quad & \sum_{a\in A} c(a)\cdot x_a & \\ \text{s.t.} \quad & \sum_{a\in\delta^-(v)} x_a - \sum_{a\in\delta^+(v)} x_a = b(v) & \text{for all } v\in V \\ & x_a \le u(a) & \text{for all } a\in A \\ & x_a \ge 0 & \text{for all } a\in A \end{aligned}$$

Dual LP:

$$\begin{aligned} \max \quad & \sum_{v\in V} b(v)\cdot y_v + \sum_{a\in A} u(a)\cdot z_a & \\ \text{s.t.} \quad & y_w - y_v + z_{(v,w)} \le c\big((v,w)\big) & \text{for all } (v,w)\in A \\ & z_a \le 0 & \text{for all } a\in A \end{aligned}$$

- result follows from complementary slackness conditions □

Introduction to

Linear and Combinatorial Optimization

12 Minimum Cost Flow Problem

12.2 Cycle Cancelling

Negative-Cycle Canceling Algorithm

Negative-Cycle Canceling Algorithm

i compute a feasible b-transshipment x or determine that none exists;

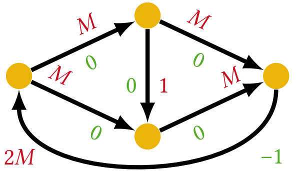
ii while there is a negative-cost dicycle C in D_x

iii   set $x := x + \delta \cdot \chi^C$ with $\delta := \min\{u_x(a) \mid a \in C\}$;

Remarks:

- negative-cost dicycle C in step ii can be found in $O(nm)$ time by the Ford-Bellman Algorithm
- number of iterations is only pseudo-polynomial in the input size
- if arc capacities and b-values are integral, algorithm returns integral min-cost b-transshipment

min-cost circulation



capacity; cost

The mean cost of a dicycle C in D_x is

$$\frac{c(C)}{|C|} = \frac{1}{|C|} \sum_{a \in C} c(a).$$

Theorem 12.6 Choosing a minimum mean-cost dicycle in step ii of the Negative-Cycle Canceling Algorithm, the number of iterations is in $O(n \cdot m^2 \cdot \log n)$.

- the following observation is similar to Observation 8.22 for shortest paths

Observation 12.7 For given arc costs $c \in \mathbb{R}^A$ and node potential $y \in \mathbb{R}^V$, define arc costs $c' \in \mathbb{R}^A$ by $c'_{(v,w)} := c_{(v,w)} + y_v - y_w$. Then, a feasible b-transshipment x has minimum cost w.r.t. c if and only if it has minimum cost w.r.t. c'. Moreover, for a dicycle $C \in D_x$ it holds that $c(C) = c'(C)$.

Proof:

- $c'(C) = \sum_{a=(v,w)\in C}\big(c(a) + y_v - y_w\big) = \sum_{a \in C} c(a) = c(C)$
- result follows from Theorem 12.4 □

Prerequisites for Proof of Thm. 12.6

Let $x_0, x_1, x_2, \ldots$ denote the b-transshipment after iterations $0, 1, 2, \ldots$

Let $A_i := A_{x_i}$ and C_i be the dicycle in A_i chosen in iteration $i + 1$.

By choice of C_i, the value $\varepsilon_i := -c(C_i)/|C_i|$ is minimal such that there is a potential $y^i \in \mathbb{R}^V$ with

$$c(a) + \varepsilon_i \geq y^i_w - y^i_v \qquad \text{for all } a = (v, w) \in A_i. \tag{$\star$}$$

Due to Observation 12.7, we may assume for some fixed i that $y^i \equiv 0$.

Lemma 12.8

- i $\varepsilon_{i+1} \leq \varepsilon_i$ for all $i = 0, 1, 2, \ldots$
- ii $\varepsilon_{i+m} \leq \left(1 - \frac{1}{n}\right) \cdot \varepsilon_i$ for all $i = 0, 1, 2, \ldots$
- iii Let $t := 2 \cdot n \cdot m \cdot \lceil \ln(n) \rceil$; then $\varepsilon_t < \frac{\varepsilon_0}{2n}$.
- iv For $i = 0, 1, 2, \ldots$, there is an arc $a \in C_i$ with $a \notin C_h$ for all $h \geq i + t$.

Proof of Lemma 12.8

wlog, we assume that $i = 0$ and $y^i = 0$

i $\varepsilon_{i+1} \leq \varepsilon_i$

- $c(a) \geq -\varepsilon_0$ for all $a \in A_0$ and $c(a) = -\varepsilon_0$ for all $a \in C_0$
- $A_1 \subseteq A_0 \cup C_0^{-1}$ and $c(a) = \varepsilon_0 > 0$ for all $a \in C_0^{-1}$

  $\Longrightarrow c(a) \geq -\varepsilon_0$ for all $a \in A_1 \Longrightarrow \varepsilon_1 \leq \varepsilon_0$

ii $\varepsilon_{i+m} \leq \left(1 - \frac{1}{n}\right)\varepsilon_i$

- at least one of the dicycles $C_0, \dots, C_{m-1}$ contains arc a with $c(a) \geq 0$ since otherwise all arcs on $C_0, \dots, C_{m-1}$ are negative and each A_k arises from A_{k-1} by deleting at least one arc of negative cost and adding only new arcs of positive cost

  $\Longrightarrow A_m$ has only non-negative arcs, the algorithm terminates, $\varepsilon_m \leq 0$
- let h be smallest index such that C_h contains a with $c(a) \geq 0$

  $\Longrightarrow c(C_h) \geq -(|C_h| - 1)\varepsilon_0$

  $\Longrightarrow \varepsilon_h = -\frac{c(C_h)}{|C_h|} \leq \frac{|C_h|-1}{|C_h|}\varepsilon_0 \leq \frac{n-1}{n}\varepsilon_0$

Proof of Lemma 12.8 (Cont.)

iii $\varepsilon_t < \frac{\varepsilon_0}{2n}$ for $t := 2nm\lceil\ln(n)\rceil$

- $\varepsilon_t \leq \left(1 - \frac{1}{n}\right)^{2n\lceil\ln(n)\rceil} \varepsilon_0 < \varepsilon_0 \left(\frac{1}{e}\right)^{2\lceil\ln(n)\rceil} \leq \frac{\varepsilon_0}{n^2} \leq \frac{\varepsilon_0}{2n}$

iv $\exists a \in C_i$ with $a \notin C_h$ for all $h \geq i + t$

- wlog, assume that $i = 0$, $y^t = 0$ $\quad(\Longrightarrow c(a) \geq -\varepsilon_t, \forall a \in A_t)$
- $c(C_0) = -\varepsilon_0|C_0|$

  $\Longrightarrow$ there is $a_0 \in C_0$ with $c(a_0) \leq -\varepsilon_0 < -2n\varepsilon_t \leq -\varepsilon_t$

  $\Longrightarrow a_0 \notin A_t \Longrightarrow x_t(a_0) = u_{a_0}$ (wlog $a_0 \in A$)
- assume that $x_h(a_0) < x_t(a_0)$ for some $h > t$

  $\Longrightarrow x_t - x_h$ is a circulation in $D_h = (V, A_h)$

  $\Longrightarrow A_h$ contains dicycle C with $a_0 \in C$

  $\Longrightarrow A_t$ contains C^{-1}

  $\Longrightarrow -c(a) = c(a^{-1}) \geq -\varepsilon_t$ for all $a \in C$

  $\Longrightarrow c(C) = c(a_0) + c(C \setminus \{a_0\}) < -2n\varepsilon_t + (|C|-1)\varepsilon_t \leq -n\varepsilon_h \leq -|C|\varepsilon_h$ ⚡

Proof of Theorem 12.6:

- by Lemma 12.8, in every iteration i there is an arc $a \in C_i$ with $a \notin C_h$ for all $h \geq i + 2nm\lceil \ln(n) \rceil$
- after $O(nm^2 \log n)$ iterations no arc can appear in any negative cycle □

Theorem 12.9 A minimum mean-cost dicycle can be found in $O(n \cdot m)$ time.

Proof: cf. sketch on next slides.

Corollary 12.10 A min-cost b-transshipment can be found in $O(n^2 \cdot m^3 \cdot \log n)$ time.

Remarks

- The running time of the Minimum-mean Cycle Canceling Algorithm can be improved to $O(n \cdot m^2 \cdot \log^2 n)$.
- The Minimum-mean Cycle Canceling Algorithm can be interpreted as a generalization of the Edmonds-Karp Algorithm.

Computation of a minimum mean-cost dicycle

Theorem: A minimum mean-cost dicycle can be found in $O(n \cdot m)$ time.

Lemma Let $D = (V, A)$ be a digraph with arc costs $c_a, \forall a \in A$, and denote by $d^k(v)$ the least cost of a walk starting from v and traversing exactly k arcs, $k \geq 0$. Then, the minimum mean-cost of a dicycle in D is equal to

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

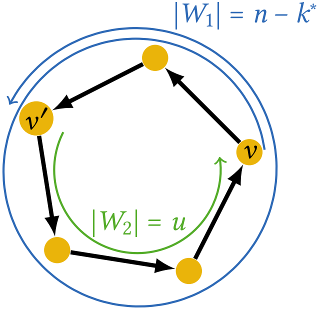
Proof:

- We first prove the lemma in the case that the min cost of a dicycle is 0 (and hence the minimum mean-cost of a dicycle is 0).
- Let $v \in V$ arbitrary. The walk of $d^n(v)$ must have a cycle (of length $\ell > 0$). Removing this cycle yields a walk of length $k = n - \ell < n$, of cost at most $d^n(v)$.
  $\implies d^k(v) \leq d^n(v)$.

  This shows: $\forall v \in V, \exists k < n : d^n(v) \geq d^k(v)$, i.e., $\alpha \geq 0$.

Proof (cont.):

- To prove $\alpha \leq 0$, we need to show $\exists v \in V : \forall k < n, d^n(v) \leq d^k(v)$.
- Let C be a cycle of cost 0, and v' an arbitrary node on the cycle. Let $k^* < n$ such that $d^{k^*}(v')$ is minimal.
- Let $v \in C$ be the node such that walking around C for $n - k^*$ steps ends in v' if we start in v. Let W_1 be this walk, and W_2 be the $v' - v$ path of length u along C.



- For all $0 \leq k \leq n-1$, it holds
$d^n(v) \leq c(W_1) + d^{k^*}(v') \leq c(W_1) + d^{k+u}(v') \leq c(W_1) + c(W_2) + d^k(v) \leq d^k(v)$.

Computation of a minimum mean-cost dicycle

Proof (cont.):

- So far, we have proved $\alpha = 0$ whenever the minimum mean-cost dicycle is 0.
- The general case (min mean-cost $\neq 0$) can be reduced to the above case by modifying the costs of the digraph (cf. exercises). □

Theorem: A minimum mean-cost dicycle can be found in $O(n \cdot m)$ time.

Proof:

There is a dynamic program for computing the minimum mean-cost

$$\alpha := \min_{v \in V} \max_{0 \leq k \leq n-1} \frac{d^n(v) - d^k(v)}{n - k}$$

in $O(nm)$. Moreover, the dynamic program can be adapted to also return a cycle realizing the mean-cost α (see exercises). □

Introduction to

Linear and Combinatorial Optimization

12 Minimum Cost Flow Problem

12.3 Successive Shortest Paths

Remarks

In the following we assume without loss of generality that in a given min-cost b-transshipment problem

- i all arc costs are nonnegative;
- ii there is a dipath of infinite capacity between every pair of nodes.

Theorem 12.11 Let x be a feasible min-cost b-transshipment, $s, t \in V$, and P a min-cost s-t-dipath in D_x with bottleneck capacity $u_x(P) := \min_{a \in P} u_x(a)$. Then,

$$x + \delta \cdot \chi^P \quad \text{with} \quad 0 \leq \delta \leq u_x(P)$$

is a feasible min-cost b'-transshipment with

$$b'(v) := \begin{cases} b(v) + \delta & \text{for } v = t, \\ b(v) - \delta & \text{for } v = s, \\ b(v) & \text{otherwise.} \end{cases}$$

Proof of Theorem 12.11

- $x' = x + \delta\chi^P$ is obviously a feasible b'-transshipment
- to prove that x' has minimum cost, we show that there is a feasible potential in $D_{x'}$
- let $p(v)$ be the cost of a min-cost s-v-dipath in D_x, then p is a feasible potential
- P is a min-cost s-t-dipath in D_x, thus,

$$\begin{aligned} c(a) &\geq p(w) - p(v) \quad \text{for all } a = (v, w) \in A_x \\ c(a) &= p(w) - p(v) \quad \text{for all } a = (v, w) \in P \end{aligned}$$

$\Longrightarrow c(a^{-1}) = p(v) - p(w)$ for all $a^{-1} = (w, v) \in P^{-1}$

- p is a feasible potential for $D_{x'}$ as well because $A_{x'} \subseteq A_x \cup P^{-1}$ □

Successive Shortest Path Algorithm

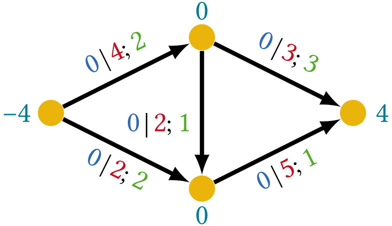
i set $x := 0$; $\bar{b} := b$;

ii while $\bar{b} \neq 0$

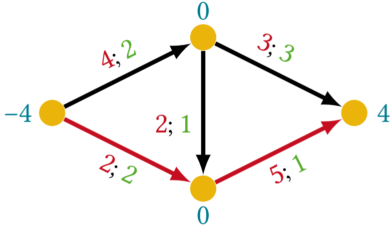
iii find min-cost s-t-dipath P in D_x for $s, t \in V$, $\bar{b}(s) < 0$, $\bar{b}(t) > 0$;

iv set $\delta := \min\{-\bar{b}(s), \bar{b}(t), u_x(P)\}$ and

$$x := x + \delta \cdot \chi^P, \quad \bar{b}(s) := \bar{b}(s) + \delta, \quad \bar{b}(t) := \bar{b}(t) - \delta;$$



flow | capacity; cost
balance



residual capacity; residual cost
imbalance

Successive Shortest Path Algorithm

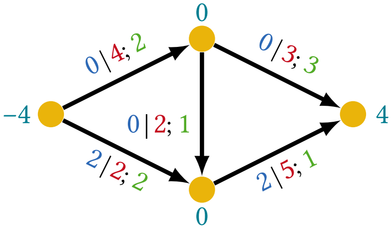
i set $x := 0$; $\bar{b} := b$;

ii while $\bar{b} \neq 0$

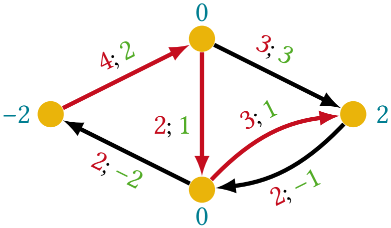
iii find min-cost s-t-dipath P in D_x for $s, t \in V$, $\bar{b}(s) < 0$, $\bar{b}(t) > 0$;

iv set $\delta := \min\{-\bar{b}(s), \bar{b}(t), u_x(P)\}$ and

$$x := x + \delta \cdot \chi^P, \quad \bar{b}(s) := \bar{b}(s) + \delta, \quad \bar{b}(t) := \bar{b}(t) - \delta;$$



flow | capacity; cost
balance



residual capacity; residual cost
imbalance

Successive Shortest Path Algorithm

Successive Shortest Path Algorithm

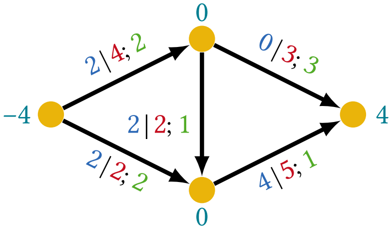
i set $x := 0$; $\bar{b} := b$;

ii while $\bar{b} \neq 0$

iii find min-cost s-t-dipath P in D_x for $s, t \in V$, $\bar{b}(s) < 0$, $\bar{b}(t) > 0$;

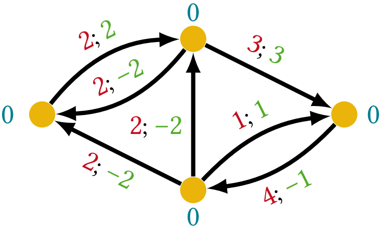
iv set $\delta := \min\{-\bar{b}(s), \bar{b}(t), u_x(P)\}$ and

$$x := x + \delta \cdot \chi^P, \quad \bar{b}(s) := \bar{b}(s) + \delta, \quad \bar{b}(t) := \bar{b}(t) - \delta;$$



flow | capacity; cost

balance



residual capacity; residual cost

imbalance

Proof of Theorem 12.12

Theorem 12.12 If all arc capacities and b-values are integral and $\sum_{v \in V} b(v) = 0$, the Successive Shortest Path Algorithm terminates with an integral min-cost b-transshipment after at most $\frac{1}{2} \sum_{v \in V} |b(v)|$ iterations.

Proof:

- initial flow $x \equiv 0$ is a min-cost-circulation since $c(a) \geq 0$ for all $a \in A$
- by induction and Theorem 12.11, x always satisfies the optimality criterion and is, thus, a min-cost $(b - \bar{b})$–transshipment
- since all arc capacities and b-values are integral, the algorithm maintains an integral flow and an integral imbalance in every iteration

  $\Longrightarrow \delta$ is integral $\Longrightarrow \delta \geq 1$

  $\Longrightarrow \sum_{v \in V} |\bar{b}(v)|$ is decreased by at least 2 in every iteration □

For a flow x and $\Delta > 0$, let $A_x^\Delta := \{a \in A_x \mid u_x(a) \geq \Delta\}$, $D_x^\Delta := (V, A_x^\Delta)$; set $U := \max\{\max_{a \in A} u(a), \max_{v \in V} |b_v|\}$.

Successive Shortest Path Algorithm with Capacity Scaling

i set $x := 0$, $\Delta := 2^{\lfloor \log U \rfloor}$, $p(v) := 0$ for all $v \in V$;

ii while $\Delta \geq 1$

iii for all $a = (v, w) \in A_x^\Delta$ with $c(a) < p(w) - p(v)$

iv   set $b(v) := b(v) + u_x(a)$ and $b(w) := b(w) - u_x(a)$;

   augment x by sending $u_x(a)$ units of flow along arc a;

v set $S(\Delta) := \{v \in V \mid b(v) \leq -\Delta\}$, $T(\Delta) := \{v \in V \mid b(v) \geq \Delta\}$;

vi while $S(\Delta) \neq \emptyset$ and $T(\Delta) \neq \emptyset$

vii   find min-cost s-t-dipath P in D_x^Δ for some $s \in S(\Delta)$, $t \in T(\Delta)$;

   set p to the vector of shortest (min-cost) path distances from s;

   augment Δ flow units along P in x; update b, $S(\Delta)$, $T(\Delta)$, D_x^Δ;

viii $\Delta := \Delta/2$;

Analysis of Running Time

Remark

- Steps iii–iv ensure that optimality conditions are always fulfilled.

Theorem 12.13 If all arc capacities and b-values are integral, the Successive Shortest Path Algorithm with Capacity Scaling terminates with an integral min-cost b-transshipment after at most $O(m \log U)$ calls to a shortest path subroutine.

- a variant of the Successive Shortest Path Algorithm with strongly polynomial running time can be obtained by a refined use of capacity scaling

  [J. B. Orlin: A faster strongly polynomial minimum cost flow algorithm, Oper. Res., 1993]

Proof of Theorem 12.13

- by construction, the optimality criterion is always fullfilled in D_x^Δ
- after last iteration $D_x^1 = D_x$ and the computed b-transshipment has minimum cost
- we claim that at the start of the inner while loop (step vi), we have

$$\sum_{v \in v\,:\,b_v > 0} b_v \leq 2\Delta(n+m) \qquad (\star)$$

  - at the end of the previous inner while loop, either $S(2\Delta) = \emptyset$ or $T(2\Delta) = \emptyset$, thus, either $\sum_{v \in V\,:\,b_v > 0} b_v = -\sum_{v \in V\,:\,b_v < 0} b_v \leq 2n\Delta$
  - (holds also before the first iteration since $S(2U) = T(2U) = \emptyset$)
  - at the beginning of the iteration in steps iii and iv only arcs are saturated with $\Delta \leq u_x(a) \leq 2\Delta$
  - steps iii and iv increase $\sum_{v \in V\,:\,b_v > 0} b_v$ by at most $2\Delta m$
- by $(\star)$, there are at most $O(m)$ iterations of the inner while-loop in step iv
- the number of iterations of the outer while loop is $O(\log U)$ □