Flows in networks

Stanislav Palúch

Fakulta riadenia a informatiky, Žilinská univerzita

28. apríla 2016

Definition

A capacitated network *is a weakly connected arc weighted digraph* $\overrightarrow{G} = (V, H, c)$ *containing two distinguished vertices*

- s – *source with* $\operatorname{ideg}(s) = 0$ *and*
- t – *sink or target with* $\operatorname{odeg}(t) = 0$

and in which arc weight $c(h) > 0$ *of every arc* $h \in H$ *is integer and represents capacity of arc* h*.*

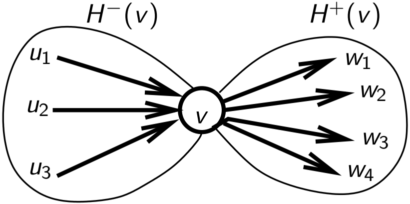
Notation: Let $v \in V$ be a vertex of a digraph $\overrightarrow{G} = (V, H, c)$.

- $H^+(v)$ is the set of all arcs outgouing from vertex v.
- $H^-(v)$ is the set of all arcs incomming into vertex$v$.

Sets $H^+(v)$ and $H^-(v)$

It holds for the sets $H^+(v)$, $H^-(v)$:

$$\begin{aligned} H^-(v) &= \{(u,j) \mid j = v,\ (u,j) \in H\}, \\ H^+(v) &= \{(i,w) \mid i = v,\ (i,w) \in H\}. \end{aligned}$$



Set $H^-(v) = \{(u_1, v), (u_2, v), (u_3, v)\}$
and set $H^+(v) = \{(v, w_1), (v, w_2), (v, w_3), (v, w_4)\}$

Definition

A flow y in the network $\overrightarrow{G} = (V, H, c)$ *is an integer function* $\mathbf{y} : H \to \mathbb{R}$ *defined on the arc set* H *for which it holds:*

1. $\mathbf{y}(h) \geq 0$ *for all* $h \in H$ (1)
2. $\mathbf{y}(h) \leq c(h)$ *for all* $h \in H$ (2)
3. $\sum_{h \in H^+(v)} \mathbf{y}(h) = \sum_{h \in H^-(v)} \mathbf{y}(h)$ *for all vertices* $v \in V$, *such that* $v \neq s$, $v \neq t$ (3)
4. $\sum_{h \in H^+(s)} \mathbf{y}(h) = \sum_{h \in H^-(t)} \mathbf{y}(h)$ (4)

The value of flow y *is the number* $F(\mathbf{y}) = \sum_{h \in H^+(s)} \mathbf{y}(h)$ *(which is equal to* $\sum_{h \in H^-(t)} \mathbf{y}(h)$*).*

Flow in the network

Definition

A flow y in the network $\overrightarrow{G} = (V, H, c)$ *is an integer function* $\mathbf{y} : H \to \mathbb{R}$ *defined on the arc set* H *for which it holds:*

$$1. \quad \mathbf{y}(h) \geq 0 \quad \text{for all } h \in H \tag{1}$$

$$2. \quad \mathbf{y}(h) \leq c(h) \quad \text{for all } h \in H \tag{2}$$

$$3. \sum_{h \in H^+(v)} \mathbf{y}(h) = \sum_{h \in H^-(v)} \mathbf{y}(h) \quad \text{for all vertices } v \in V, \quad \text{such that } v \neq s, \ v \neq t \tag{3}$$

$$4. \sum_{h \in H^+(s)} \mathbf{y}(h) = \sum_{h \in H^-(t)} \mathbf{y}(h) \tag{4}$$

The value of flow y *is the number* $F(\mathbf{y}) = \sum_{h \in H^+(s)} \mathbf{y}(h)$ *(which is equal to* $\sum_{h \in H^-(t)} \mathbf{y}(h)$*).*

Definition

A flow y in the network $\overrightarrow{G} = (V, H, c)$ *is an integer function* $\mathbf{y} : H \to \mathbb{R}$ *defined on the arc set* H *for which it holds:*

$$1. \quad \mathbf{y}(h) \geq 0 \quad \text{for all } h \in H \tag{1}$$

$$2. \quad \mathbf{y}(h) \leq c(h) \quad \text{for all } h \in H \tag{2}$$

$$3. \sum_{h \in H^+(v)} \mathbf{y}(h) = \sum_{h \in H^-(v)} \mathbf{y}(h) \quad \text{for all vertices } v \in V, \quad \text{such that } v \neq s, \; v \neq t \tag{3}$$

$$4. \sum_{h \in H^+(s)} \mathbf{y}(h) = \sum_{h \in H^-(t)} \mathbf{y}(h) \tag{4}$$

The value of flow y *is the number* $F(\mathbf{y}) = \sum_{h \in H^+(s)} \mathbf{y}(h)$ *(which is equal to* $\sum_{h \in H^-(t)} \mathbf{y}(h)$*).*

Flow in the network

Definition

A flow y in the network $\overrightarrow{G} = (V, H, c)$ *is an integer function* $\mathbf{y} : H \to \mathbb{R}$ *defined on the arc set* H *for which it holds:*

$$1. \quad \mathbf{y}(h) \geq 0 \quad \textit{for all } h \in H \tag{1}$$

$$2. \quad \mathbf{y}(h) \leq c(h) \quad \textit{for all } h \in H \tag{2}$$

$$3. \sum_{h \in H^+(v)} \mathbf{y}(h) = \sum_{h \in H^-(v)} \mathbf{y}(h) \quad \textit{for all vertices } v \in V, \quad \textit{such that } v \neq s, \ v \neq t \tag{3}$$

$$4. \sum_{h \in H^+(s)} \mathbf{y}(h) = \sum_{h \in H^-(t)} \mathbf{y}(h) \tag{4}$$

The value of flow y *is the number* $F(\mathbf{y}) = \sum_{h \in H^+(s)} \mathbf{y}(h)$ *(which is equal to* $\sum_{h \in H^-(t)} \mathbf{y}(h)$*).*

Flow in the network

Definition

A flow y in the network $\overrightarrow{G} = (V, H, c)$ *is an integer function* $\mathbf{y} : H \rightarrow \mathbb{R}$ *defined on the arc set* H *for which it holds:*

$$1. \quad \mathbf{y}(h) \geq 0 \quad \textit{for all } h \in H \tag{1}$$

$$2. \quad \mathbf{y}(h) \leq c(h) \quad \textit{for all } h \in H \tag{2}$$

$$3. \sum_{h \in H^+(v)} \mathbf{y}(h) = \sum_{h \in H^-(v)} \mathbf{y}(h) \quad \textit{for all vertices } v \in V, \quad \textit{such that } v \neq s, \; v \neq t \tag{3}$$

$$4. \sum_{h \in H^+(s)} \mathbf{y}(h) = \sum_{h \in H^-(t)} \mathbf{y}(h) \tag{4}$$

The value of flow y *is the number* $F(\mathbf{y}) = \sum_{h \in H^+(s)} \mathbf{y}(h)$ *(which is equal to* $\sum_{h \in H^-(t)} \mathbf{y}(h)$*).*

Maximum flow in a capacitated network

Definition

A maximum flow in a capacitated network $\overrightarrow{G}$ *is a flow* $\mathbf{y}^*$ *having the maximum value* $F(\mathbf{y}^*)$, *i,e. if* $F(\mathbf{y}) \leq F(\mathbf{y}^*)$ *for every flow* $\mathbf{y}$ *in* $\overrightarrow{G}$.
An arc $h \in H$ *is* **saturated**, *if* $\mathbf{y}(h) = c(h)$.

Remark

- *A flow in a network is a real function* $\mathbf{y} : H \to \mathbb{R}$ *defined on the set of all arcs.*

  The number $\mathbf{y}(h)$ *is the value of function* $\mathbf{y}$ *for certain element* h *of its domain.*

  (Compare $\mathbf{y}$ *and* $\mathbf{y}(h)$ *with two notions: function* $\log$ *and value* $\log(2)$*).*

  The value $\mathbf{y}(h)$ *will be called* **a flow along arc** h.

- *A flow* $\mathbf{y}$ *in the network* $\overrightarrow{G}$ *is in fact another edge weight, therefore a network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *can be considered as a digraph* $\overrightarrow{G} = (V, H, c, \mathbf{y})$ *with two edge weights.*

Definition

A maximum flow in a capacitated network $\overrightarrow{G}$ *is a flow* $\mathbf{y}^*$ *having the maximum value* $F(\mathbf{y}^*)$*, i,e. if* $F(\mathbf{y}) \le F(\mathbf{y}^*)$ *for every flow* $\mathbf{y}$ *in* $\overrightarrow{G}$.
An arc $h \in H$ *is* **saturated**, *if* $\mathbf{y}(h) = c(h)$.

Remark

- *A flow in a network is a real function* $\mathbf{y} : H \to \mathbb{R}$ *defined on the set of all arcs.*

  The number $\mathbf{y}(h)$ *is the value of function* $\mathbf{y}$ *for certain element* h *of its domain.*

  (Compare $\mathbf{y}$ *and* $\mathbf{y}(h)$ *with two notions: function* log *and value* $\log(2)$*).*

  The value $\mathbf{y}(h)$ *will be called* **a flow along arc** h.

- *A flow* $\mathbf{y}$ *in the network* $\overrightarrow{G}$ *is in fact another edge weight, therefore a network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *can be considered as a digraph* $\overrightarrow{G} = (V, H, c, \mathbf{y})$ *with two edge weights.*

Definition

A maximum flow in a capacitated network $\overrightarrow{G}$ *is a flow* $\mathbf{y}^*$ *having the maximum value* $F(\mathbf{y}^*)$, *i,e. if* $F(\mathbf{y}) \leq F(\mathbf{y}^*)$ *for every flow* $\mathbf{y}$ *in* $\overrightarrow{G}$.
An arc $h \in H$ *is* **saturated**, *if* $\mathbf{y}(h) = c(h)$.

Remark

- *A flow in a network is a real function* $\mathbf{y} : H \to \mathbb{R}$ *defined on the set of all arcs.*

  The number $\mathbf{y}(h)$ *is the value of function* $\mathbf{y}$ *for certain element* h *of its domain.*

  (Compare $\mathbf{y}$ *and* $\mathbf{y}(h)$ *with two notions: function* $\log$ *and value* $\log(2)$*).*

  The value $\mathbf{y}(h)$ *will be called* **a flow along arc** h.

- *A flow* $\mathbf{y}$ *in the network* $\overrightarrow{G}$ *is in fact another edge weight, therefore a network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *can be considered as a digraph* $\overrightarrow{G} = (V, H, c, \mathbf{y})$ *with two edge weights.*

Reserve path and augmenting path

Definition

Let $\overrightarrow{G} = (V, H)$ is a digraph, let v, $w \in V$, let $\boldsymbol{\mu}(v, w)$ is a v–w quasi-path in $\overrightarrow{G}$

$$\boldsymbol{\mu}(v, w) = (v = v_1, h_1, v_2, \dots, v_i, h_i, v_{i+1}, \dots, v_{k-1}, h_k, v_k = w).$$

Arc h_i is called **a forward arc of quasi-path $\boldsymbol{\mu}(v, w)$** *if $h_i = (v_i, v_{i+1})$.*
Arc h_i is called **a backward arc of quasi-path $\boldsymbol{\mu}(v, w)$** *if $h_i = (v_{i+1}, v_i)$.*

Definition

Let $\overrightarrow{G} = (V, H, c, \mathbf{y})$ is a capacitated network with flow $\mathbf{y}$, let v, $w \in V$.
Let $\boldsymbol{\mu}(v, w)$ is a v–w quasi-path, let h be a arc of this quasi-path.
The reserve $r(h)$ of an arc h in a quasi-path $\boldsymbol{\mu}(v, w)$ *is:*

$$r(h) = \begin{cases} c(h) - \mathbf{y}(h) & \text{if the arc } h \text{ is a forward arc of } \boldsymbol{\mu}(v, w) \\ \mathbf{y}(h) & \text{if the arc } h \text{ is a backward arc of } \boldsymbol{\mu}(v, w) \end{cases} \tag{5}$$

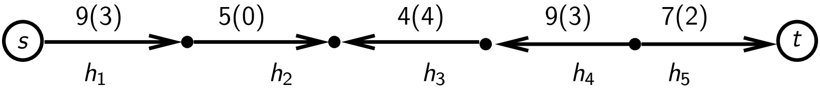
Augmenting quasi-path

Definition

The reserve $r(\boldsymbol{\mu}(v,w))$ ***of quasi-path*** $\boldsymbol{\mu}(v,w)$ *is the minimum of reserves of arcs of this quasi-path.*

A quasi-path $\boldsymbol{\mu}(v,w)$ *is* **a reserve quasi-path** *if* $r(\boldsymbol{\mu}(v,w)) > 0$*, i.e. if it has positive reserve.*

A reserve quasi-path $\boldsymbol{\mu}(s,t)$ *form source to sink is called* **an augmenting quasi-path***.*



$r(h_1) = 9 - 3$ $r(h_2) = 5 - 0$ $r(h_5) = 7 - 2$

$r(h_1) = 6$ $r(h_2) = 5$ $r(h_3) = 4$ $r(h_4) = 3$ $r(h_5) = 5$

Augmenting quasi-path.
Notation 9(3) of arc h_1 means that $c(h_1) = 9$, $\mathbf{y}(h_1) = 3$.

The reserve of quasi-path is $\min\{6, 5, 4, 3, 5\} = 3$.

Augmenting quasi-path gives a hint how to increase the flow

Theorem

If there exists an augmenting quasi-path in the network $\overrightarrow{G} = (V, H, c)$ *with flow* $\mathbf{y}$ *then the flow* $\mathbf{y}$ *is not maximal.*

PROOF.

Let $\mu(z, u)$ be an augmenting s–t quasi-path from source to sink. having reserve r.

Let us define a new flow $\mathbf{y}'$:

$$\mathbf{y}'(h) = \begin{cases} \mathbf{y}(h) & \text{if } h \notin \mu(z, u) \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } \mu(z, u) \\ \mathbf{y}(h) - r & \text{if } h \text{ is a backward arc of } \mu(z, u) \end{cases}$$

Reserve of augmenting quasi-path was calculated as the minimum of reserves of all arcs of this quasi-path defined by equations (9), therefore values $\mathbf{y}'(h)$ of flow $\mathbf{y}'$ have to fulfill (1) (i.e. $\mathbf{y}'(h) \geq 0$), (2) (i.e. $\mathbf{y}'(h) \leq c(h)$).

Augmenting quasi-path gives a hint how to increase the flow

Theorem

If there exists an augmenting quasi-path in the network $\overrightarrow{G} = (V, H, c)$ with flow $\mathbf{y}$ then the flow $\mathbf{y}$ is not maximal.

PROOF.
Let $\boldsymbol{\mu}(z, u)$ be an augmenting s–t quasi-path from source to sink. having reserve r.
Let us define a new flow $\mathbf{y}'$:

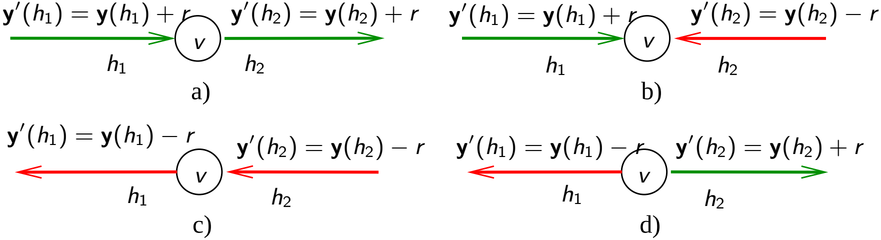
$$\mathbf{y}'(h) = \begin{cases} \mathbf{y}(h) & \text{if } h \notin \boldsymbol{\mu}(z, u) \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } \boldsymbol{\mu}(z, u) \\ \mathbf{y}(h) - r & \text{if } h \text{ is a backward arc of } \boldsymbol{\mu}(z, u) \end{cases}$$

Reserve of augmenting quasi-path was calculated as the minimum of reserves of all arcs of this quasi-path defined by equations (9), therefore values $\mathbf{y}'(h)$ of flow $\mathbf{y}'$ have to fulfill (1) (i.e. $\mathbf{y}'(h) \geq 0$), (2) (i.e. $\mathbf{y}'(h) \leq c(h)$).

Flow property (3) $\sum_{h\in H^+(v)} \mathbf{y}(h) = \sum_{h\in H^-(v)}$ holds for flow $\mathbf{y}'$

For flow $\mathbf{y}$ it holds (3):

$$\sum_{h\in H^+(v)} \mathbf{y}(h) = \sum_{h\in H^-(v)} \mathbf{y}(h) \quad \text{for all } v \in V, \quad \text{such that } v \neq s, \ v \neq t$$



Four possibilities of direction of arcs incident with vertex v on aumenting quasi-path.

a) $\mathbf{y}'(h_1)$ increases $\sum_{h\in H^-(v)} \mathbf{y}(h)$ by r, $\mathbf{y}'(h_2)$ increases $\sum_{h\in H^+(v)} \mathbf{y}(h)$ by r

b) $\mathbf{y}'(h_1)$ increases $\sum_{h\in H^-(v)} \mathbf{y}(h)$ by r, $\mathbf{y}'(h_2)$ decreases $\sum_{h\in H^-(v)} \mathbf{y}(h)$ by r

c) $\mathbf{y}'(h_1)$ decreases $\sum_{h\in H^+(v)} \mathbf{y}(h)$ by r, $\mathbf{y}'(h_2)$ decreases $\sum_{h\in H^-(v)} \mathbf{y}(h)$ by r

d) $\mathbf{y}'(h_1)$ decreases $\sum_{h\in H^+(v)} \mathbf{y}(h)$ by r, $\mathbf{y}'(h_2)$ increases $\sum_{h\in H^+(v)} \mathbf{y}(h)$ by r

Augmenting path increases flow

First arc of augmenting quasi-path belongs to $H^+(s)$,
last arc of augmenting quasi-path belongs to $H^-(t)$.
Therefore

$$F(\mathbf{y}') = \sum_{h \in H^+(s)} \mathbf{y}'(h) = \sum_{h \in H^+(s)} \mathbf{y}(h) + r = F(\mathbf{y}) + r \qquad (6)$$

$$\sum_{h \in H^-(t)} \mathbf{y}'(h) = \sum_{h \in H^-(t)} \mathbf{y}(h) + r = F(\mathbf{y}) + r \qquad (7)$$

It follows from (6), (7) that flow property (4) holds for $\mathbf{y}'$ whereas flow value $F(\mathbf{y}')$ of new flow $\mathbf{y}'$ is greater by r then the flow value $F(\mathbf{y})$ of old flow $\mathbf{y}$. □

Ford – Fulkers maximum flow theorem

Theorem (Ford – Fulkerson)

Flow $\mathbf{y}$ *in the network* $\overrightarrow{G} = (V, H, c)$ *with source* s *and sink* t *is the maximum flow if and only if there does not exist a* s*–*t *augmenting quasi-path.*

Algorithm

Fordov – Fulkerson maximum flow algorithm in a capacitated network $\overrightarrow{G} = (V, H, c)$.

- **Step 1.** *Take an initial feasible flow* $\mathbf{y}$ *e.g. zero flow.*
- **Step 2.** *Find an augmenting quasi-path* $\mu(s,t)$ *in network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$.
- **Step 3.** *If there is no augmenting quasi-path in network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *then the flow* $\mathbf{y}$ *is the maximum flow.*

  STOP.
- **Step 4.** *If* $\mu(s,t)$ *is an augmenting quasi-path with reserve* r *then change the flow* $\mathbf{y}$ *as follows:*

$$\mathbf{y}(h) := \begin{cases} \mathbf{y}(h) & \text{if } h \text{ is not an arc of } \mu(s,t) \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } \mu(s,t) \\ \mathbf{y}(h) - r & \text{ak } h \text{ is a backwoard arc of } \mu(s,t) \end{cases}$$

GOTO Step 2.

Algorithm

Fordov – Fulkerson maximum flow algorithm in a capacitated network $\overrightarrow{G} = (V, H, c)$.

- **Step 1.** *Take an initial feasible flow* $\mathbf{y}$ *e.g. zero flow.*
- **Step 2.** *Find an augmenting quasi-path* $\boldsymbol{\mu}(s,t)$ *in network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$.
- **Step 3.** *If there is no augmenting quasi-path in network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *then the flow* $\mathbf{y}$ *is the maximum flow.*

  STOP.
- **Step 4.** *If* $\boldsymbol{\mu}(s,t)$ *is an augmenting quasi-path with reserve* r *then change the flow* $\mathbf{y}$ *as follows:*

$$\mathbf{y}(h) := \begin{cases} \mathbf{y}(h) & \text{if } h \text{ is not an arc of } \boldsymbol{\mu}(s,t) \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } \boldsymbol{\mu}(s,t) \\ \mathbf{y}(h) - r & \text{ak } h \text{ is a backwoard arc of } \boldsymbol{\mu}(s,t) \end{cases}$$

GOTO Step 2.

Algorithm

Fordov – Fulkerson maximum flow algorithm in a capacitated network $\overrightarrow{G} = (V, H, c)$.

- **Step 1.** *Take an initial feasible flow* $\mathbf{y}$ *e.g. zero flow.*
- **Step 2.** *Find an augmenting quasi-path* $\boldsymbol{\mu}(s, t)$ *in network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$.
- **Step 3.** *If there is no augmenting quasi-path in network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *then the flow* $\mathbf{y}$ *is the maximum flow.*

  STOP.
- **Step 4.** *If* $\boldsymbol{\mu}(s, t)$ *is an augmenting quasi-path with reserve* r *then change the flow* $\mathbf{y}$ *as follows:*

$$\mathbf{y}(h) := \begin{cases} \mathbf{y}(h) & \text{if } h \text{ is not an arc of } \boldsymbol{\mu}(s,t) \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } \boldsymbol{\mu}(s,t) \\ \mathbf{y}(h) - r & \text{ak } h \text{ is a backwoard arc of } \boldsymbol{\mu}(s,t) \end{cases}$$

GOTO Step 2.

Algorithm

Fordov – Fulkerson maximum flow algorithm in a capacitated network $\overrightarrow{G} = (V, H, c)$.

- **Step 1.** *Take an initial feasible flow* $\mathbf{y}$ *e.g. zero flow.*
- **Step 2.** *Find an augmenting quasi-path* $\boldsymbol{\mu}(s,t)$ *in network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$.
- **Step 3.** *If there is no augmenting quasi-path in network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *then the flow* $\mathbf{y}$ *is the maximum flow.*

  STOP.
- **Step 4.** *If* $\boldsymbol{\mu}(s,t)$ *is an augmenting quasi-path with reserve* r *then change the flow* $\mathbf{y}$ *as follows:*

$$\mathbf{y}(h) := \begin{cases} \mathbf{y}(h) & \text{if } h \text{ is not an arc of } \boldsymbol{\mu}(s,t) \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } \boldsymbol{\mu}(s,t) \\ \mathbf{y}(h) - r & \text{ak } h \text{ is a backwoard arc of } \boldsymbol{\mu}(s,t) \end{cases}$$

GOTO Step 2.

Algorithm to find an augmenting quasi-path

Algorithm

Algorithm to find an augmenting quasi-path $\mu(z, u)$ in the capacitated $\overrightarrow{G} = (V, H, c)$ with flow y.

Assign a label $x(i)$ to all vertices $i \in V$ with following meaning:

- *If $x(i) = \infty$, then no reserve s–i quasi-path was found till now.*
- *If $x(i) < \infty$, then there exist a reserve s–i quasi-path having last but on vertex equal to $|x(i)|$ (absolte velue of $x(i)$).*
- *If moreover $x(i) > 0$, then the last arc of this quasi-path is forward arc $(x(i), i)$.*

  If $x(i) < 0$, then the last arc of this quasi-path is bacward arc $(i, x(i))$.
- *Set $x(s) := 0$ for source s.*

Algorithm

Algorithm to find an augmenting quasi-path $\mu(z, u)$ in the capacitated $\overrightarrow{G} = (V, H, c)$ with flow y.
Assign a label $x(i)$ to all vertices $i \in V$ with following meaning:

- *If $x(i) = \infty$, then no reserve s–i quasi-path was found till now.*
- *If $x(i) < \infty$, then there exist a reserve s–i quasi-path having last but on vertex equal to $|x(i)|$ (absolte velue of $x(i)$).*
- *If moreover $x(i) > 0$, then the last arc of this quasi-path is forward arc $(x(i), i)$.*

  If $x(i) < 0$, then the last arc of this quasi-path is bacward arc $(i, x(i))$.
- *Set $x(s) := 0$ for source s.*

Algorithm to find an augmenting quasi-path

Algorithm

Algorithm to find an augmenting quasi-path $\mu(z,u)$ in the capacitated $\overrightarrow{G} = (V, H, c)$ with flow y.
Assign a label $x(i)$ to all vertices $i \in V$ with following meaning:

- *If $x(i) = \infty$, then no reserve s–i quasi-path was found till now.*
- *If $x(i) < \infty$, then there exist a reserve s–i quasi-path having last but on vertex equal to $|x(i)|$ (absolte velue of $x(i)$).*
- *If moreover $x(i) > 0$, then the last arc of this quasi-path is forward arc $(x(i), i)$.*

  If $x(i) < 0$, then the last arc of this quasi-path is bacward arc $(i, x(i))$.
- *Set $x(s) := 0$ for source s.*

Algorithm to find an augmenting quasi-path

Algorithm

Algorithm to find an augmenting quasi-path $\mu(z,u)$ in the capacitated $\overrightarrow{G} = (V, H, c)$ with flow y.
Assign a label $x(i)$ to all vertices $i \in V$ with following meaning:

- *If $x(i) = \infty$, then no reserve s–i quasi-path was found till now.*
- *If $x(i) < \infty$, then there exist a reserve s–i quasi-path having last but on vertex equal to $|x(i)|$ (absolte velue of $x(i)$).*
- *If moreover $x(i) > 0$, then the last arc of this quasi-path is forward arc $(x(i), i)$.*

  If $x(i) < 0$, then the last arc of this quasi-path is bacward arc $(i, x(i))$.
- *Set $x(s) := 0$ for source s.*

Algorithm to find an augmenting quasi-path

Algorithm (– continuation)

Denote::

- $\mathcal{E}$ – the set of vertices with finite label $x(\ )$ the neighborhood of which is not explored till now.

  If $i \in \mathcal{E}$ then there exists a reserve s–i quasi-path and there is a chance that this quasi-path can be extended by one arc.

Remark

The set $\mathcal{E}$ has similar finction as the set $\mathcal{E}$ in label set a label correct algorithm.

Algorithm (– continuation)

Denote::

- *$\mathcal{E}$ – the set of vertices with finite label $x(\ )$ the neighborhood of which is not explored till now.*

  If $i \in \mathcal{E}$ then there exists a reserve s–i quasi-path and there is a chance that this quasi-path can be extended by one arc.

Remark

The set $\mathcal{E}$ has similar finction as the set $\mathcal{E}$ in label set a label correct algorithm.

Algorithm (– continuation)

Denote::

- $\mathcal{E}$ – *the set of vertices with finite label* $x(\ )$ *the neighborhood of which is not explored till now.*

  If $i \in \mathcal{E}$ *then there exists a reserve* s–i *quasi-path and there is a chance that this quasi-path can be extended by one arc.*

Remark

The set $\mathcal{E}$ *has similar finction as the set* $\mathcal{E}$ *in label set a label correct algorithm.*

Algorithm to find an augmenting quasi-path

Algorithm (– continuation)

- **Step 1.** *Initialization.*
  $\mathcal{E} := \{s\}$.
  Set $x(s) := 0$ *and for all* $i \in V$, $i \neq s$ *set* $x(i) := \infty$.
- **Step 2.** *If* $x(t) < \infty$, *create augmenting s–t quasi-path using labels* $|x(\ )|$:
  $$(s = |x^{(k)}(t)|,\ |x^{(k-1)}(t)|,\ \dots,\ |x^{(2)}(t)|,\ |x(t)|,\ t,)$$
  and STOP.
- **Step 3.** *If* $\mathcal{E} = \emptyset$, *then there there does not exist an augmenting quasi-path* $\mu(s,t)$.
  STOP.
- **Step 4.** *Extract a vertex* $i \in \mathcal{E}$ *from* $\mathcal{E}$. *Set* $\mathcal{E} := \mathcal{E} - \{i\}$.

  For every vertex $j \in V^{+}(i)$ *such that* $x(j) = \infty$ *do:*

  If $\mathbf{y}(i,j) < c(i,j)$, *then set* $x(j) := i$, $\mathcal{E} := \mathcal{E} \cup \{j\}$.

  For every vertex $j \in V^{-}(i)$ *such that* $x(j) = \infty$ *do:*

  If $\mathbf{y}(j,i) > 0$, *then set* $x(j) := -i$, $\mathcal{E} := \mathcal{E} \cup \{j\}$.
  GOTO Step 2.

♣

Algorithm (– continuation)

- **Step 1.** *Initialization.*
  $\mathcal{E} := \{s\}$.
  Set $x(s) := 0$ *and for all* $i \in V$, $i \neq s$ *set* $x(i) := \infty$.
- **Step 2.** *If* $x(t) < \infty$, *create augmenting s–t quasi-path using labels* $|x(\ )|$*:*
  $$(s = |x^{(k)}(t)|,\ |x^{(k-1)}(t)|,\ \dots,\ |x^{(2)}(t)|,\ |x(t)|,\ t,)$$
  and STOP.
- **Step 3.** *If* $\mathcal{E} = \emptyset$, *then there there does not exist an augmenting quasi-path* $\mu(s,t)$.
  STOP.
- **Step 4.** *Extract a vertex* $i \in \mathcal{E}$ *from* $\mathcal{E}$. *Set* $\mathcal{E} := \mathcal{E} - \{i\}$.
  For every vertex $j \in V^{+}(i)$ *such that* $x(j) = \infty$ *do:*
  If $\mathbf{y}(i,j) < c(i,j)$, *then set* $x(j) := i$, $\mathcal{E} := \mathcal{E} \cup \{j\}$.
  For every vertex $j \in V^{-}(i)$ *such that* $x(j) = \infty$ *do:*
  If $\mathbf{y}(j,i) > 0$, *then set* $x(j) := -i$, $\mathcal{E} := \mathcal{E} \cup \{j\}$.
  GOTO Step 2.

♣

Algorithm to find an augmenting quasi-path

Algorithm (– continuation)

- **Step 1.** *Initialization.*
  $\mathcal{E} := \{s\}$.
  Set $x(s) := 0$ *and for all* $i \in V$, $i \neq s$ *set* $x(i) := \infty$.
- **Step 2.** *If* $x(t) < \infty$, *create augmenting s–t quasi-path using labels* $|x(\ )|$:
  $$(s = |x^{(k)}(t)|,\ |x^{(k-1)}(t)|,\ \dots,\ |x^{(2)}(t)|,\ |x(t)|,\ t,)$$
  and STOP.
- **Step 3.** *If* $\mathcal{E} = \emptyset$, *then there there does not exist an augmenting quasi-path* $\boldsymbol{\mu}(s,t)$.
  STOP.
- **Step 4.** *Extract a vertex* $i \in \mathcal{E}$ *from* $\mathcal{E}$. *Set* $\mathcal{E} := \mathcal{E} - \{i\}$.
  For every vertex $j \in V^{+}(i)$ *such that* $x(j) = \infty$ *do:*
  If $\mathbf{y}(i,j) < c(i,j)$, *then set* $x(j) := i$, $\mathcal{E} := \mathcal{E} \cup \{j\}$.
  For every vertex $j \in V^{-}(i)$ *such that* $x(j) = \infty$ *do:*
  If $\mathbf{y}(j,i) > 0$, *then set* $x(j) := -i$, $\mathcal{E} := \mathcal{E} \cup \{j\}$.
  GOTO Step 2.

♣

Algorithm (– continuation)

- **Step 1.** *Initialization.*
  $\mathcal{E} := \{s\}$.
  Set $x(s) := 0$ *and for all* $i \in V$, $i \neq s$ *set* $x(i) := \infty$.
- **Step 2.** *If* $x(t) < \infty$, *create augmenting s–t quasi-path using labels* $|x(\ )|$:
  $$(s = |x^{(k)}(t)|,\ |x^{(k-1)}(t)|,\ \dots,\ |x^{(2)}(t)|,\ |x(t)|,\ t,)$$
  and STOP.
- **Step 3.** *If* $\mathcal{E} = \emptyset$, *then there there does not exist an augmenting quasi-path* $\boldsymbol{\mu}(s,t)$.
  STOP.
- **Step 4.** *Extract a vertex* $i \in \mathcal{E}$ *from* $\mathcal{E}$. *Set* $\mathcal{E} := \mathcal{E} - \{i\}$.

  For every vertex $j \in V^+(i)$ *such that* $x(j) = \infty$ *do:*

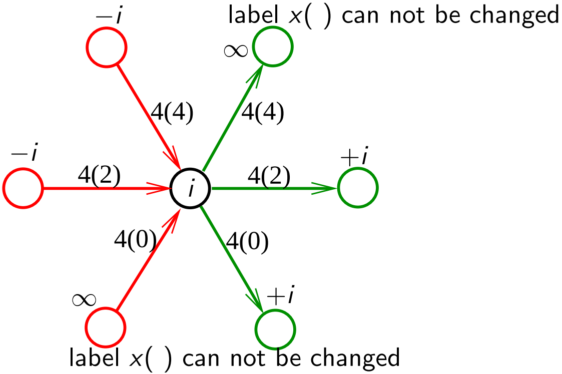
  If $\mathbf{y}(i,j) < c(i,j)$, *then set* $x(j) := i$, $\mathcal{E} := \mathcal{E} \cup \{j\}$.

  For every vertex $j \in V^-(i)$ *such that* $x(j) = \infty$ *do:*

  If $\mathbf{y}(j,i) > 0$, *then set* $x(j) := -i$, $\mathcal{E} := \mathcal{E} \cup \{j\}$.
  GOTO Step 2.

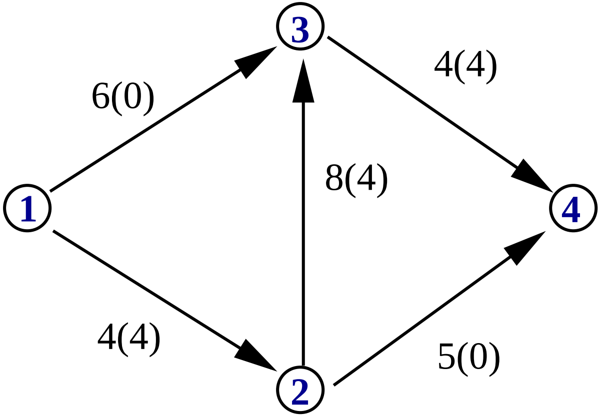
♣

Way how to set labels for vertices of $V^+(i)$, $V^+(i)$



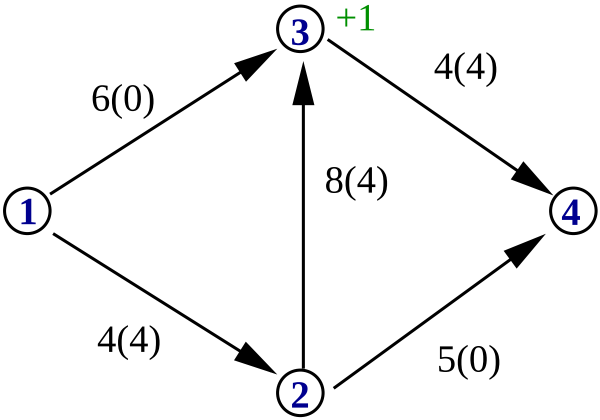
Way how to set labels for vertices of $V^+(i)$, $V^+(i)$.
Symbol 4(2) means that corresponding arc has capacity 4 and flow 2 flows along this arc.
Green circles represent vertices of the set $V^+(i)$,
Red circles represent vertices of the set $V^-(i)$.

Example – searching for an augmenting path



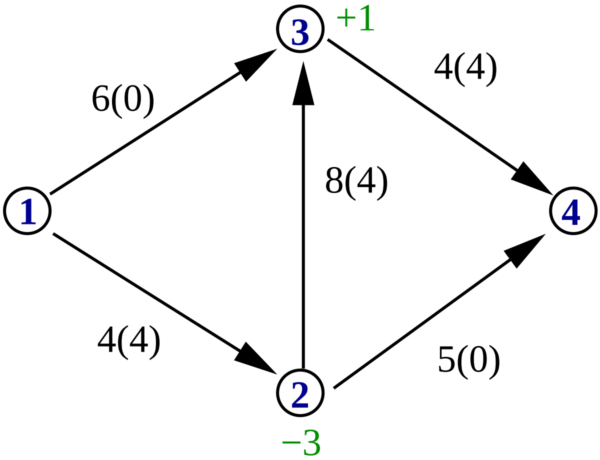
$\mathcal{N} = \{2, 3, 4\}$
$\mathcal{E} = \{1\}, \quad \mathcal{E} = \mathcal{E} - \{1\}, \quad i = 1 \quad V^{+}(1) \cap \mathcal{N} = \{2, 3\},\ V^{-}(1) \cap \mathcal{N} = \{\ \}$

Example – searching for an augmenting path



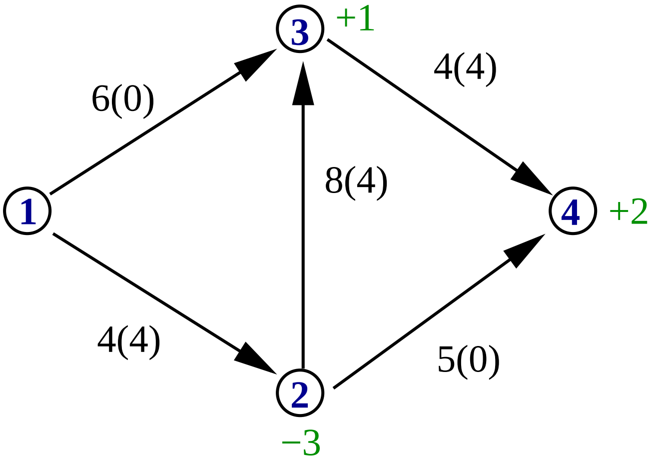
$\mathcal{N} = \mathcal{N} - \{3\} = \{2, 4\}$
$\mathcal{E} = \{3\}, \quad \mathcal{E} = \mathcal{E} - \{3\}, \quad i = 3 \quad V^+(3) \cap \mathcal{N} = \{4\},\ V^-(3) \cap \mathcal{N} = \{2\}$

Example – searching for an augmenting path

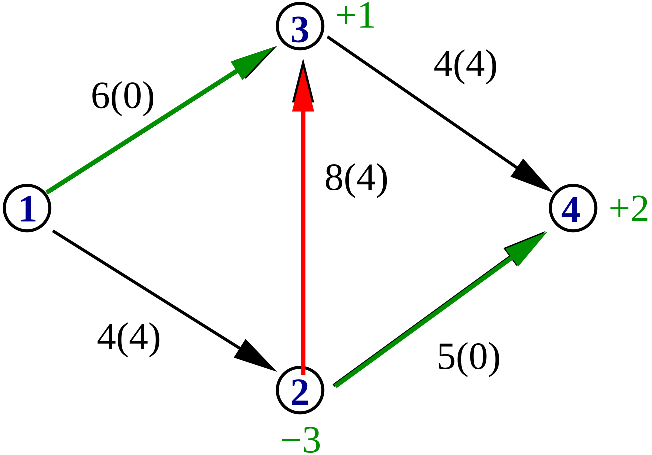


$\mathcal{N} = \mathcal{N} - \{2\} = \{4\}$
$\mathcal{E} = \{2\}$, $\mathcal{E} = \mathcal{E} - \{2\}$, $i = 2$ $V^+(2) \cap \mathcal{N} = \{4\}$, $V^-(2) \cap \mathcal{N} = \{\ \}$

Example – searching for an augmenting path



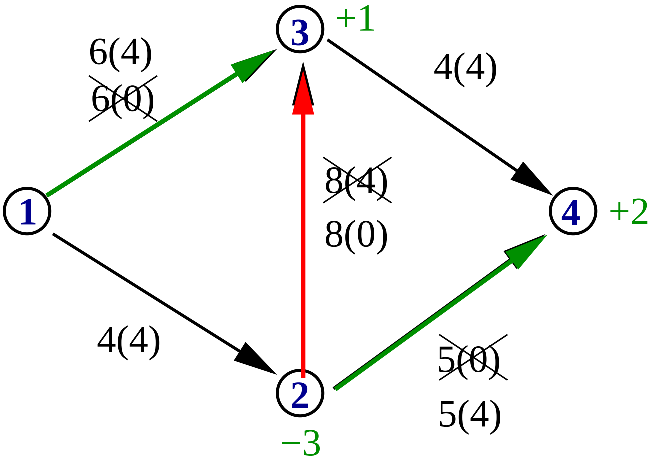
Example – searching for an augmenting path



Augmenting quasi-path is $(1, (1, 3), (2, 3), 2, (2, 4), 4)$.
Reserve of arc $(1, 3)$ is 6, reserve of arc $(2, 3)$ is 4, reserve of arc $(2, 4)$ is 5.

Reserve of augmenting quasi-path is $\min\{6, 4, 5\} = 4$.

Example – searching for an augmenting path



Augmenting quasi-path is $(1, (1,3), (2,3), 2, (2,4), 4)$.
Reserve of arc $(1,3)$ is 6, reserve of arc $(2,3)$ is 4, reserve of arc $(2,4)$ is 5.

Reserve of augmenting quasi-path is $\min\{6,4,5\} = 4$.

Minimum cost maximum flow

Definition

Let $\overrightarrow{G} = (V, H, c, d)$ be a capacitated network where $d(h)$ is another arc weight of arc h representing the cost for a flow unit transported along arc h.
Let $\mathbf{y}$ be a flow in the capacitated network $\overrightarrow{G}$.
The cost of flow y *is defined as:*

$$D(\mathbf{y}) = \sum_{h \in H} d(h).\mathbf{y}(h)$$

Definition

The minimum cost flow *with flow value F is the flow with value F which has the least cost from all flows with flow value F.*

Remark

The maximum cost flow can be defined similarly.

Remark

Very often problem is to find minimum cost flow having maximum value

Minimum cost maximum flow

Definition

Let $\overrightarrow{G} = (V, H, c, d)$ be a capacitated network where $d(h)$ is another arc weight of arc h representing the cost for a flow unit transported along arc h.
Let $\mathbf{y}$ be a flow in the capacitated network $\overrightarrow{G}$.
The cost of flow y *is defined as:*

$$D(\mathbf{y}) = \sum_{h \in H} d(h).\mathbf{y}(h)$$

Definition

The minimum cost flow *with flow value F is the flow with value F which has the least cost from all flows with flow value F.*

Remark

The maximum cost flow can be defined similarly.

Remark

Very often problem is to find minimum cost flow having maximum value

Minimum cost maximum flow

Definition

Let $\overrightarrow{G} = (V, H, c, d)$ be a capacitated network where $d(h)$ is another arc weight of arc h representing the cost for a flow unit transported along arc h.
Let $\mathbf{y}$ be a flow in the capacitated network $\overrightarrow{G}$.
The cost of flow y *is defined as:*

$$D(\mathbf{y}) = \sum_{h \in H} d(h).\mathbf{y}(h)$$

Definition

The minimum cost flow *with flow value F is the flow with value F which has the least cost from all flows with flow value F.*

Remark

The maximum cost flow can be defined similarly.

Remark

Very often problem is to find minimum cost flow having maximum value

Reserve of an arc in quasi-cycle, rezerve quasi-cycle

Definition

Let $\overrightarrow{G} = (V, H, c, d)$ be a capacitated network with flow $\mathbf{y}$, let C be a quasi-cycle in $\overrightarrow{G}$.

Reserve $r(h)$ **of an arc** h **in quasi-cycle** C *is*

$$r(h) = \begin{cases} c(h) - \mathbf{y}(h) & \text{if arc } h \text{ is a forward arc of } C \\ \mathbf{y}(h) & \text{if arc } h \text{ is a backward arc of } C \end{cases}$$

Reserve of quasi-cycle C *is the minimum of reserves of its arcs.*

Quasi-cycle C is called **a reserve quasi-cycle** *if its reserve is positive.*

The cost $d(C)$ **of quasi-cycle** C *is defined as total sum weights $d(\ )$ of its forward arcs minus total sum of weights $d(\ )$ of its backward arcs.*

Reserve of an arc in quasi-cycle, rezerve quasi-cycle

Definition

Let $\overrightarrow{G} = (V, H, c, d)$ *be a capacitated network with flow* $\mathbf{y}$*, let* C *be a quasi-cycle in* $\overrightarrow{G}$*.*

Reserve $r(h)$ **of an arc** h **in quasi-cycle** C *is*

$$r(h) = \begin{cases} c(h) - \mathbf{y}(h) & \text{if arc } h \text{ is a forward arc of } C \\ \mathbf{y}(h) & \text{if arc } h \text{ is a backward arc of } C \end{cases}$$

Reserve of quasi-cycle C *is the minimum of reserves of its arcs.*

Quasi-cycle C *is called* **a reserve quasi-cycle** *if its reserve is positive.*

The cost $d(C)$ **of quasi-cycle** C *is defined as total sum weights* $d(\ )$ *of its forward arcs minus total sum of weights* $d(\ )$ *of its backward arcs.*

Reserve of an arc in quasi-cycle, rezerve quasi-cycle

Definition

Let $\overrightarrow{G} = (V, H, c, d)$ *be a capacitated network with flow* $\mathbf{y}$*, let* C *be a quasi-cycle in* $\overrightarrow{G}$.

Reserve $r(h)$ **of an arc** h **in quasi-cycle** C *is*

$$r(h) = \begin{cases} c(h) - \mathbf{y}(h) & \text{if arc } h \text{ is a forward arc of } C \\ \mathbf{y}(h) & \text{if arc } h \text{ is a backward arc of } C \end{cases}$$

Reserve of quasi-cycle C *is the minimum of reserves of its arcs.*

Quasi-cycle C *is called* **a reserve quasi-cycle** *if its reserve is positive.*

The cost $d(C)$ **of quasi-cycle** C *is defined as total sum weights* $d(\ )$ *of its forward arcs minus total sum of weights* $d(\ )$ *of its backward arcs.*

Ford Fulkerson Min-Cost-Max-Flow Theorem

Theorem

Flow $\mathbf{y}$ *in the capacitated network* $\overrightarrow{G} = (V, H, c, d)$ *is the minimum cost flow of its flow value if and only if there does not exist a reserve quasi-cycle with negative cost in* $\overrightarrow{G}$.

Algorithm

Algorithm to find minimum cost flow with given value in capacitated network $\overrightarrow{G} = (V, H, c, d)$.

- **Step 1.** *Start with flow* $\mathbf{y}$ *having given value in the network* $\overrightarrow{G} = (V, H, c, d)$.
- **Step 2.** *Find a reserve quasi-cycle with negative cost in the network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *or find out that such a quasi-cycle does not exist.*
- **Step 3.** *If there does not exist a reserve quasi-cycle with negative cost then the flow* $\mathbf{y}$ *is minimum cost flow with its flow value. STOP.*
- **Step 4.** *If a reserve quasi-cycle C with negative cost does exist then denote by r its reserve and change the flow* $\mathbf{y}$ *as follows:*

$$\mathbf{y}(h) := \begin{cases} \mathbf{y}(h) & \text{if } h \text{ is not an arc of } C \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } C \\ \mathbf{y}(h) - r & \text{if } h \text{ is a backward arc of } C \end{cases}$$

GOTO Step 2.

Algorithm

Algorithm to find minimum cost flow with given value in capacitated network $\overrightarrow{G} = (V, H, c, d)$.

- **Step 1.** *Start with flow* $\mathbf{y}$ *having given value in the network* $\overrightarrow{G} = (V, H, c, d)$.
- **Step 2.** *Find a reserve quasi-cycle with negative cost in the network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *or find out that such a quasi-cycle does not exist.*
- **Step 3.** *If there does not exist a reserve quasi-cycle with negative cost then the flow* $\mathbf{y}$ *is minimum cost flow with its flow value. STOP.*
- **Step 4.** *If a reserve quasi-cycle* C *with negative cost does exist then denote by* r *its reserve and change the flow* $\mathbf{y}$ *as follows:*

$$\mathbf{y}(h) := \begin{cases} \mathbf{y}(h) & \text{if } h \text{ is not an arc of } C \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } C \\ \mathbf{y}(h) - r & \text{if } h \text{ is a backward arc of } C \end{cases}$$

GOTO Step 2.

Algorithm

Algorithm to find minimum cost flow with given value in capacitated network $\overrightarrow{G} = (V, H, c, d)$.

- **Step 1.** *Start with flow* $\mathbf{y}$ *having given value in the network* $\overrightarrow{G} = (V, H, c, d)$.
- **Step 2.** *Find a reserve quasi-cycle with negative cost in the network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *or find out that such a quasi-cycle does not exist.*
- **Step 3.** *If there does not exist a reserve quasi-cycle with negative cost then the flow* $\mathbf{y}$ *is minimum cost flow with its flow value. STOP.*
- **Step 4.** *If a reserve quasi-cycle* C *with negative cost does exist then denote by* r *its reserve and change the flow* $\mathbf{y}$ *as follows:*

$$\mathbf{y}(h) := \begin{cases} \mathbf{y}(h) & \text{if } h \text{ is not an arc of } C \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } C \\ \mathbf{y}(h) - r & \text{if } h \text{ is a backward arc of } C \end{cases}$$

GOTO Step 2.

Algorithm

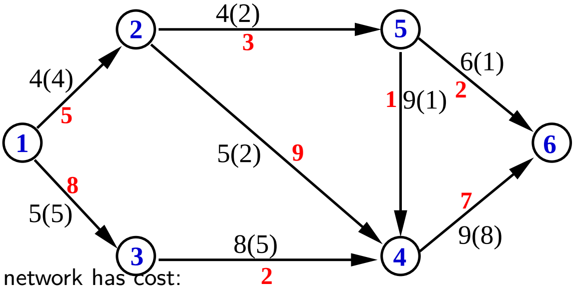
Algorithm to find minimum cost flow with given value in capacitated network $\overrightarrow{G} = (V, H, c, d)$.

- **Step 1.** *Start with flow* $\mathbf{y}$ *having given value in the network* $\overrightarrow{G} = (V, H, c, d)$.
- **Step 2.** *Find a reserve quasi-cycle with negative cost in the network* $\overrightarrow{G}$ *with flow* $\mathbf{y}$ *or find out that such a quasi-cycle does not exist.*
- **Step 3.** *If there does not exist a reserve quasi-cycle with negative cost then the flow* $\mathbf{y}$ *is minimum cost flow with its flow value. STOP.*
- **Step 4.** *If a reserve quasi-cycle* C *with negative cost does exist then denote by* r *its reserve and change the flow* $\mathbf{y}$ *as follows:*

$$\mathbf{y}(h) := \begin{cases} \mathbf{y}(h) & \text{if } h \text{ is not an arc of } C \\ \mathbf{y}(h) + r & \text{if } h \text{ is a forward arc of } C \\ \mathbf{y}(h) - r & \text{if } h \text{ is a backward arc of } C \end{cases}$$

GOTO Step 2.

Example - searching for minimum cost flow



Flow in the network has cost:

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.2 + 9.2 + 2.5 + 1.1 + 2.1 + 7.8 = 153$$

Reserve quasi-cycle found: $(6, (4, 6), 4, (5, 4), 5, (5, 6), 6)$ with reserve 1 and negative cost $-7 - 1 + 2 = -6$.

New flow in the network has cost

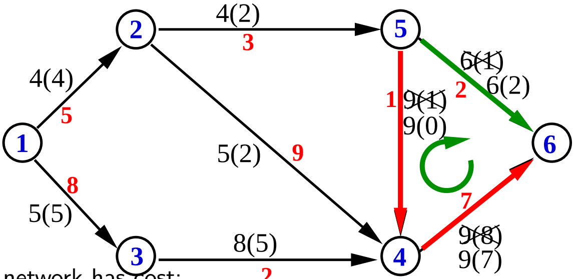
$$D(\mathbf{y}) = 5.4 + 8.5 + 3.2 + 9.2 + 2.5 + 1.0 + 2.2 + 7.7 = 147$$

Reserve quasi-cycle found: $(6, (4, 6), 4, (2, 4), 2, (2, 5), 5, (5, 6), 6)$ with reserve 2 and negative cost $-7 - 9 + 3 + 2 = -11$.

New flow in the network has cost

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.4 + 9.0 + 2.5 + 1.0 + 2.4 + 7.5 = 125$$

Example - searching for minimum cost flow



Flow in the network has cost:

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.2 + 9.2 + 2.5 + 1.1 + 2.1 + 7.8 = 153$$

Reserve quasi-cycle found: $(6, (4,6), 4, (5,4), 5, (5,6), 6)$ with reserve 1 and negative cost $-7 - 1 + 2 = -6$.

New flow in the network has cost

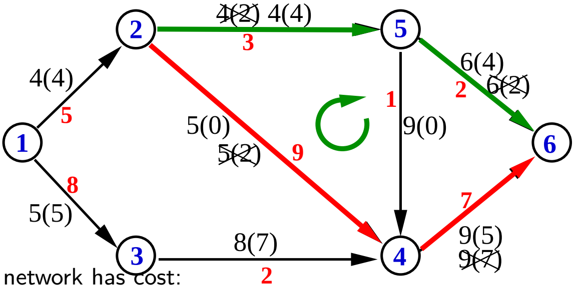
$$D(\mathbf{y}) = 5.4 + 8.5 + 3.2 + 9.2 + 2.5 + 1.0 + 2.2 + 7.7 = 147$$

Reserve quasi-cycle found: $(6, (4,6), 4, (2,4), 2, (2,5), 5, (5,6), 6)$ with reserve 2 and negative cost $-7 - 9 + 3 + 2 = -11$.

New flow in the network has cost

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.4 + 9.0 + 2.5 + 1.0 + 2.4 + 7.5 = 125$$

Example - searching for minimum cost flow



Flow in the network has cost:

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.2 + 9.2 + 2.5 + 1.1 + 2.1 + 7.8 = 153$$

Reserve quasi-cycle found: $(6, (4, 6), 4, (5, 4), 5, (5, 6), 6)$ with reserve 1 and negative cost $-7 - 1 + 2 = -6$.

New flow in the network has cost

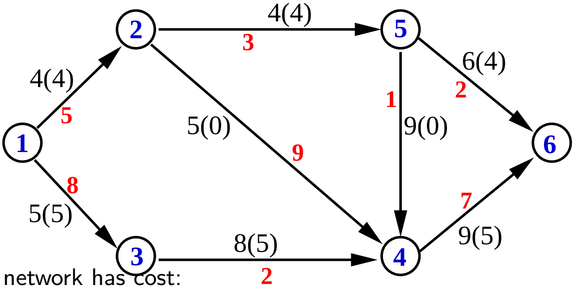
$$D(\mathbf{y}) = 5.4 + 8.5 + 3.2 + 9.2 + 2.5 + 1.0 + 2.2 + 7.7 = 147$$

Reserve quasi-cycle found: $(6, (4, 6), 4, (2, 4), 2, (2, 5), 5, (5, 6), 6)$ with reserve 2 and negative cost $-7 - 9 + 3 + 2 = -11$.

New flow in the network has cost

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.4 + 9.0 + 2.5 + 1.0 + 2.4 + 7.5 = 125$$

Example - searching for minimum cost flow



Flow in the network has cost:

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.2 + 9.2 + 2.5 + 1.1 + 2.1 + 7.8 = 153$$

Reserve quasi-cycle found: $(6, (4, 6), 4, (5, 4), 5, (5, 6), 6)$ with reserve 1 and negative cost $-7 - 1 + 2 = -6$.

New flow in the network has cost

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.2 + 9.2 + 2.5 + 1.0 + 2.2 + 7.7 = 147$$

Reserve quasi-cycle found: $(6, (4, 6), 4, (2, 4), 2, (2, 5), 5, (5, 6), 6)$ with reserve 2 and negative cost $-7 - 9 + 3 + 2 = -11$.

New flow in the network has cost

$$D(\mathbf{y}) = 5.4 + 8.5 + 3.4 + 9.0 + 2.5 + 1.0 + 2.4 + 7.5 = 125$$