

Question:

Can we improve Dinic's method for the computation of a blocking flow?

Idea (Karzanov 1974):

During each step saturate a node instead of an edge.

For $v \in V$ we define the capacity $c(v)$ of the node v by

$$c(v) := \min \left\{ \sum_{e \in \text{IN}(v)} c(e), \sum_{e \in \text{OUT}(v)} c(e) \right\}.$$

A flow f saturates a node v if

$$\sum_{e \in \text{IN}(v)} f(e) = \sum_{e \in \text{OUT}(v)} f(e) = c(v).$$

A node $v \in V$ is blocked if on each path from v to the sink t there is a saturated edge.

Idea:

- Starting in s , we transport as large as possible flow into the network. It is possible that the amount of flow which reaches a node v is larger than $\sum_{e \in \text{OUT}(v)} c(e)$.
- In that case, we reduce the surplus of v

and look for another way to transport this surplus to t .

- We take care that a blocked node remains to be blocked.

First, we need some further notations.

Let $G = (V, E, c, s, t)$ be a flow network. A preflow $p: E \rightarrow \mathbb{R}$ is a function which fulfills the following requirements:

$$1) \quad p(e) \leq c(e) \quad \forall e \in E$$

$$2) \quad \sum_{e \in N(v)} p(e) \geq \sum_{e \in \text{out}(v)} p(e) \quad \forall v \in V \setminus \{s, t\}$$

$\underbrace{\phantom{\sum_{e \in N(v)} p(e)}}$ $\underbrace{\phantom{\sum_{e \in \text{out}(v)} p(e)}}$
 $P_{\text{in}}(v)$ $P_{\text{out}}(v)$

A node $v \in V$ is called unbalanced if

$$P_{\text{in}}(v) > P_{\text{out}}(v).$$

A preflow p is a flow iff all nodes $v \in V \setminus \{s, t\}$ are balanced.

Let $G_f = (V', E', \tilde{c}, s, t)$ be the layered network constructed by the algorithm CONSTRUCTION G_f

Korzanov's method for the computation of a blocking flow separates into two steps which are iterated as long as there is a

unbalanced node. After the last step, the network contains a blocking flow.

~)

Step 1: Forward movement of preflow in G_p

Step 2: Balancing of preflow in G_p .

Note that an unbalanced node is blocked.

We balance the preflow p at an unbalanced node v such that

$$P_{in}(v) = P_{out}(v) = C(v).$$

Hence, the node v remains to be blocked.

After balancing a node v we do not allow any further change of the incoming flow of v .

An edge e is open if it is allowed to change $p(e)$. Otherwise, e is closed.

At the beginning, all edges are open.

Step 1:

For all nodes $v \in V$, we number the outgoing edges of v in any fixed order.

Idea:

The incoming preflow of a node v is transported to the next layer in the

following way:

- Transport the largest possible amount of preflow over the first open, not saturated outgoing edge of v .
- Then transport the largest possible amount of preflow over the second open, not saturated outgoing edge of v

:

and so on.

With respect to the open edges in $\text{OUT}(v)$, the following invariant will be always fulfilled:

• Invariant:

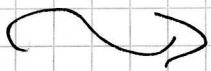
We have a sequence of saturated edges followed by one not saturated edge e with $p(e) > 0$.

This edge is called $k(v)$. All further edges have zero preflow.

For the organisation of Step 2, we use a stack J_w for the management of the incoming edges $e \in \text{IN}(w)$ of a node w with $p(e) > 0$.

- At the moment when an edge $e \in \text{IN}(w)$ obtains a preflow $p(e) > 0$, the edge e is pushed.

- If during Step 2, the incoming flow of the node w has to be reduced, the flow on the uppermost edge in K_w is reduced, then the flow on the next edge on K_w , and so on.



Step 1: Forward movement of preflow in G_f :

- (1) Start in s and transport as much as possible preflow to layer 1; i.e.,

for all $e \in \text{OUT}(s)$

do

$$p(e) := \tilde{c}(e)$$

od ;

$$\text{START} := 1 ;$$

- (2) for $i := \text{START}$ step 1 until $L-1$

do

for all $v \in V_i$

do

$$P := p_{in}(v) ;$$

while $P > 0$ and $k(v)$ defined

do

$$z := k(v) ;$$

$$P' := \min \{ \tilde{c}(e) - p(e), P \} ;$$

$$p(e) := p(e) + P' ;$$

$$P := P - P'$$

od.

od

After the termination of Step 1, for all nodes $v \in V \setminus \{s, t\}$ exactly one of the following two cases is fulfilled:

$$1. \quad P_{in}(v) = P_{out}(v).$$

$$2. \quad P_{in}(v) > P_{out}(v) = \tilde{c}(v) \quad (\text{i.e., } v \text{ is unbalanced}).$$

Step 2: Balancing of the preflow in G_f :

Let V_{max} denote the layer of highest index which contains an unbalanced node.

(1) For all unbalanced nodes $v \in V_{max}$

do

Using \tilde{K}_v , reduce the incoming flow of v until v is balanced; i.e.,

$$P_{in}(v) = P_{out}(v) = \tilde{c}(v);$$

Close the incoming edges of v .

od;

(2) START := max - 1;

Apply (2) of Step 1.

Exercise:

Work out the algorithm of Kerzakov.

For the proof of the correctness of Karzanov's method, it suffices to prove that a blocking flow is constructed. This is a direct consequence of the following lemma.

Lemma 1.7

During the whole algorithm, a blocked node remains to be blocked.

Proof:

After the first performance of Step 1, all outgoing edges of s are saturated

\Rightarrow

s is blocked.

Furthermore, each outgoing edge of an unbalanced node v is saturated

\Rightarrow

v is blocked.

Claim:

Each node which is blocked before Step 2 remains to be blocked after the performance of Step 2.

Proof of claim:

- For all nodes $v \in V_j$, $j \geq \max$ the preflow

$p(e)$ does not change for all edges e on a path from v to t in G_f .

\Rightarrow

If v is blocked before the performance of Step 2 then v is also blocked after the performance of Step 2.

- Consider $v \in V_{\max}$ with v is unbalanced. Note that v is blocked, too.

Step 2 reduces the incoming preflow of v .

Assume that the preflow $p(e)$ for $e = (w, v)$ and $w \in V_{\max-1}$ is reduced by Step 2.

Consider $x \in V_j$, $j < \max$ with x is blocked before the reduction of $p(e)$. Then

- the preflow on all paths from s to t which do not contain the edge e does not change.

and

- all paths from x to t which contain the edge e contain the blocked node v .

\Rightarrow

x remains to be blocked after the reduction of $p(e)$.

Since the source node s is blocked after the first performance of Step 1, Lemma 1.7 implies that Karonov's method computes a blocking flow.

It remains the analysis of the used time.

Lemma 1.8

Karonov's method finds a blocking flow in $O(n^2)$ time.

Proof:

- Note that after the reduction of the preflow on an edge e , the edge e is closed. Hence, the number of preflow reduction is bounded by m .
- Each edge e is saturated at most once. Hence, the number of increases of preflow on an edge e such that e is saturated is bounded by m .
- Between two subsegment balancing steps each not blocked node has at most one edge e such that $p(e)$ is increased and e is not saturated.

\Rightarrow

The number A of such increases is bounded by:

$$A \leq \sum_{i=1}^{n-1} i = \frac{n(n-1)}{2}.$$

Altogether, we have shown that the used time is $\Theta(n^2)$. ■

A clever modification of Karzanov's method has been done by Malhotra, Pramodh Kumar, and Maheshwari (1978):

Let f be a flow in a flow network

$G = (V, E, c, s, t)$ and let $v \in V$. The flow potential $p_f(v)$ of v is defined by

$$p_f(v) := \tilde{c}(v) - \sum_{(u,v) \in E'} f(u,v).$$

$r \in V'$ is called reference node if

$$p_f(r) = \min \{ p_f(v) \mid p_f(v) > 0, v \in V \setminus \{s, t\}\}.$$

Lemma 1.9

Let r be a reference node in the layered flow network $G_f = (V', E', \tilde{c}, s, t)$. Let $p_f(v) > 0 \wedge v \in V' \setminus \{s, t\}$. Furthermore, $\forall v \in V' \setminus \{s, t\} : \exists$ paths from s to v and from v to t in G_f . Then, the flow f can

be increased by $f_f(r)$ to a flow f' such that $f'_f(r) = 0$.

Proof:

Note that $\forall v \in V' \setminus \{s, t\}$ there holds

$$f_f(r) \leq f_f(v).$$

\implies

Starting in r , we can transport an additional flow of size $f_f(r)$ layer by layer to t .

Analogously, starting in r we can obtain backwards an additional flow of size $f_f(r)$ froms \sim to r .

Algorithm MPM

Input: layered network $G_f = (V', E', \tilde{c}, s, t)$

Output: blocking flow \bar{f} in G_f .

Method:

- (1) Determine a reference node r .
- (2) Transport an additional flow of size $f_f(r)$ from r to t .
- (3) Starting in r , transport backwards an additional flow of size $f_f(r)$ from s to r .
- (4) Remove all edges and all nodes such that no additional flow can be transported on these edges and through these nodes.

(5) If the network is not empty, go to (1).

Time analysis:

One performance of (1)-(5) in total

| | | |
|-----------|------------------------|----------------------|
| (1) | $\leq n-1$ comparisons | |
| (2) | n (nonsaturated) | $+ m$ (saturated) |
| (3) | n | $+ m$ |
| (4) + (5) | | $O(n+m)$ |
| <hr/> | | |
| \sum | $O(n^2)$ | |

Exercise:

- a) Work out the algorithm of Karzanov.
- b) Work out the algorithm MPM and its time analysis.