The Max Flow Problem – Push-Relabel algorithms

Jesper Larsen & Jens Clausen

jla,jc@imm.dtu.dk

Informatics and Mathematical Modelling
Technical University of Denmark

- Push-relabel algorithms for the Max-Flow problem are also sometime called **preflow-push** algorithms.
- Consider again a digraf $G = (V(G), E(G))$, in which each edge e has a capacity $u_e \in \mathcal{R}_+$.
- Furthermore, two “special” vertices r and s are given; these are called resp. the **source** and the **sink**.

Relax balance constraint

- A **preflow** in G is a function $x : E \to \mathcal{R}_+$ satisfying:

$$\forall v \in V \setminus \{r, s\} \;:\; \sum_{(w,v) \in E} x_{wv} - \sum_{(v,w) \in E} x_{vw} \geq 0$$

$$\forall (v, w) \in E \;:\; 0 \leq x_{vw} \leq u_{vw}$$

- In other words: for any vertex v except r and s, the flow excess $f_x(v)$ is non-negative - “more runs into v than out of v”. If $f_x(v) > 0$ the v is called **active**.

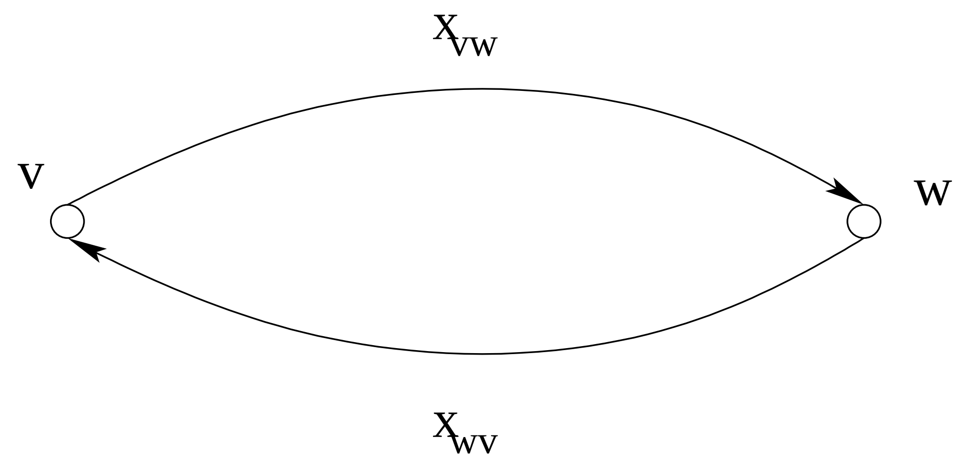
Feasible preflows

- So looking at a pair (u, v) so that $\tilde{u}_{vw} > 0$ and $f_x(v) > 0$ pushing up to:

$$\epsilon = \min(\tilde{u}_{vw}, f_x(v))$$

  will produce a new feasible preflow.
- A preflow with no active nodes is a flow.

- The Residual graph G_x remains defined as previously.



- The total flow we can push from v to w is $\tilde{u}_{vw} = u_{vw} - x_{vw} + x_{wv}$.

General idea

- The general idea now is to push as much as possible from the sink towards the sink.
- However when no more flow can be pushed towards the sink s, **and** there are still active nodes in order to restore balance of flow we push the excess back toward the source r.

Labelling

- In order to control the direction of the pushes we introduce estimates on the distances in G_x.
- A **valid labelling** of the vertices in V wrt. a preflow x is a function $d[.] : V \rightarrow \mathcal{Z}$ satisfying:

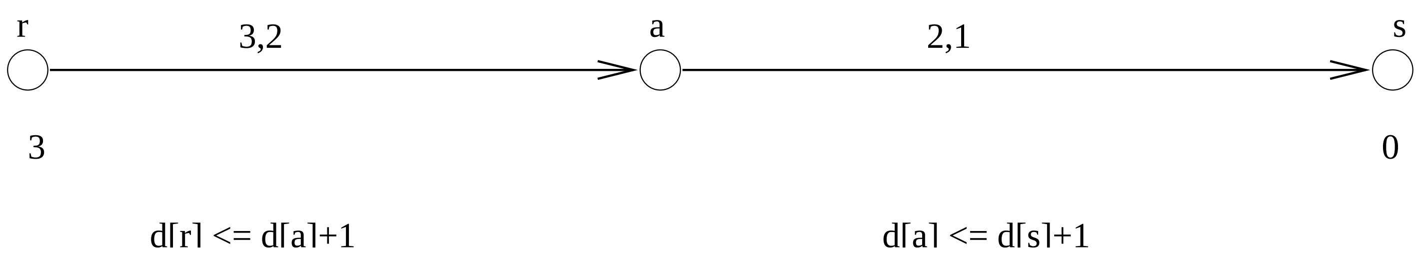
$$d[r] = n \ \wedge \ d[s] = 0$$

$$\forall \ (v, w) \ \in \ E_x : \quad d[v] \leq d[w] + 1$$

- In other words: for each vertex v, $d[v]$ ia a lower bound on the number of edges in a dipath in G_x from v to s, if such a path exists.

Preflows vs. valid labellings

- It is not al feasible preflows that admits a valid labelling.



Initialize

- It is easy to construct some feasible preflow and a valid labelling for it: (called **initialize** x, d)
  1. set $x_e = u_e$ for all outgoing arcs of r
  2. set $x_e = 0$ for all remaining arcs.
  3. set $d[r] = n$ and $d[v] = 0$ otherwise
- A valid labelling for a preflow implies an important property of the preflow, that it “saturates a cut”.

Admissible edge

- With the definition d we can be prove that for any feasible preflow x and any valid labelling d for x we have

$$d_x(v,w) \geq d[v] - d[w]$$

- We try to push flow towards nodes w having $d[w] < d[v]$, since such nodes are estimated to be closer to the ultimate destination.

Admissible edge II

- Having $d[w] < d[v]$ and a valid labelling means that push is only applied to arcs (v, w) is v is active and $d[v] = d[w] + 1$.
- An edge (v, w) is called **admissible** wrt. a preflow x and a valid labelling d wrt. x, if $(v, w) \in E_x$, v is active and $d[v] = d[w] + 1$.

Estimates on distances

- As special cases this gives us:
  - $d[v]$ is a lower bound on the distance from v to s.
  - $d[v] - n$ is a lower bound on the distance from v to r.
  - If $d[v] \geq n$ this means that there is no path from v to s.
- Admissible edges can be used to push flow excess towards s (or, if s is not reachable, towards r).

Push

The Push-operation: Consider an admissible edge (v, w). Calculate the amount of flow, which can be pushed to w (min$\{f_x(v), (x_{wv} + (u_{vw} - x_{vw}))\}$). Push this by first reducing x_{wv} as much as possible and then increasing x_{vw} as much as possible until teh relevant amount has been pushed.

Relabel

- What should we do if there are no more admissible edges and a node v is still active?
- **The Relabel-operation:** Consider an active vertex v, for which **no** edge $(v, w) \in E_x$ with $d(v) = d(w) + 1$. Increase $d(v)$ to $\min\{d(w) + 1 | (v, w) \in E_x\}$ - resulting in a new valid labelling.
- Notice that this will not violate the validity of the labelling.

Running times

- Analysis of the running times of the preflow-push algorithms is based on an analysis of the number of saturating pushes and non-saturating pushes.
- It can be shown that the push-relabel maximum flow algorithm can be implemented to run in time $O(n^2m)$ or $O(n^3)$.