CMPUT 675: Topics in Combinatorics and Optimization **Fall 2016**

Lecture 4 (Sep 14): Push-Relabel Analysis

Lecturer: Zachary Friggstad *Scribe: Zachary Friggstad*

4.1 Push-Relabel Analysis

We analyze the Goldberg-Tarjan Push-Relabel algorithm that was described at the end of last lecture. We use the notation in the notes from Lecture 3.

Our main goal is the following.

Theorem 1 *In the Push-Relabel algorithm from Lecture 3, if the active vertex v selected has $\psi(v) = \max\{\psi(w) : w \text{ active}\}$ then the number of iterations is $O(n^2 \cdot \sqrt{m})$.*

We will also show in Lemma 4 that there is no $s-t$ path in G_f at any point of the algorithm, including when the algorithm terminates. Recall that a flow f is a maximum flow if and only if there is no $s-t$ path in G_f. Putting this all together, we have the following.

Corollary 1 *The Push-Relabel algorithm computes a maximum flow in $O(n^2 \cdot \sqrt{m})$ time.*

Proof. The algorithm terminates in $O(n^2 \cdot \sqrt{m})$ iterations (Theorem 1). When this happens, there are no active vertices so in fact f is a flow (i.e. $f(\delta^{in}(v)) = f(\delta^{out}(v))$). As there is no $s-t$ path in G_f (Lemma 4), then f is a maximum flow. ■

We will briefly comment on implementing the algorithm to run in $O(n^2 \cdot \sqrt{m})$ at the end of these notes.

4.1.1 Invariants

Claim 1 *Every relabeling step for v increases $\psi(v)$ by at least 1 (and there is at least one edge $vw \in \delta^{out}_{E_f}(v)$).*

Proof. First, v is active so $\text{ex}_f(v) > 0$. This means $f(e) > 0$ for some $e \in \delta^{in}(v)$, so $\overleftarrow{e} \in E_f$.

Next, the fact that ψ is a distance label yet v is not active means $\psi(v) \leq \psi(w)$ for each $vw \in E_f$. So relabeling $\psi(v)$ to $\min\{\psi(w) + 1 : vw \in E_f\}$ increases $\psi(v)$ by at least 1. ■

Lemma 1 *At the start of the algorithm and at the end of each iteration, f is a preflow and ψ is a distance label with respect to the current preflow f.*

Proof. Initially f is a preflow because the only edges e with $f(e) > 0$ are in $\delta^{in}(s)$ and the were set to carry flow $\mu(e)$. Also, ψ is a distance label because $\psi(s)$ is set to n, $\psi(t)$ to 0, and every $vw \in E_f$ has $v \neq s$ (as all edges exiting s are saturated) so $0 = \psi(v) \leq \psi(w) + 1$.

Inductively, suppose f, ψ are the flow and distance label at the start of an iteration and let f', ψ' be the resulting values after the iteration. If the iteration relabeled v, then $f' = f$ so f' is a preflow. To show ψ' is a distance label with respect to $f' = f$, first note that because v is active then $\psi(v) \neq s, t$ so we still have $\psi'(s) = n, \psi(t) = 0$. Next note that $\psi(v)$ was relabelled to $\psi'(v)$ in a way that ensures $\psi'(v) \leq \psi'(w) + 1$ for each $vw \in E_f = E_{f'}$. For other edges $e = uv \in E_{f'}$ note,

$$\psi'(u) = \psi(u) \leq \psi(v) + 1 \leq \psi'(v) + 2$$

by Claim 1.

Finally, suppose the iteration involved a push operation across $vw \in E_f$. The push increases the excess at w (it either increases flow on an edge entering w or decreases flow on an edge exiting w) and the amount of flow that is pushed is at most $\mathrm{ex}_f(v)$ so $\mathrm{ex}_{f'}(v) \geq 0$. The only potential edge in $E_{f'}$ that is not in E_f is wv. But the fact we only push along admissible edges shows

$$\psi'(w) = \psi(w) = \psi(v) - 1 \leq \psi(v) + 1 = \psi'(v) + 1.$$

■

4.1.2 Basic Properties of Preflows and Distance Labels

Lemma 2 *Let f be a preflow and ψ a distance label with respect to f. For any v with $\mathrm{ex}_f(v) > 0$ there is a $v - s$ path in G_f.*

Proof. Let $R \subseteq V$ be the set of nodes reachable from v in G_f. Then

$$\sum_{v \in R} \mathrm{ex}_f(v) = f(\delta^{in}(R)) - f(\delta^{out}(R)) = -f(\delta^{out}(R)) \leq 0.$$

The first is because the net contribution of any edge $e = uw$ with $u, w \in R$ to the sum is 0. The second is because if there was some $uw \in \delta^{in}(R)$ with $f(uw) > 0$, then $wu \in E_f$. But $w \in R$ is reachable from v, so u would also be reachable and should then be in R, a contradiction.

Finally, we know $\mathrm{ex}_f(v) > 0$ because v is active. Since $v \in R$ and the total excess of nodes in R is ≤ 0, some $w \in R$ has $\mathrm{ex}_f(v) < 0$. Since f is a preflow, we must then have $w = s$, so $s \in R$. ■

Lemma 3 *For any $u, w \in V$, if there is a $u - w$ path in G_f then $\psi(u) \leq \psi(w) + n - 1$.*

Proof. Let $u = v_0, v_1, v_2, \ldots, v_{k-1}, v_k = w$ be a path in G_f. Note $k \leq n - 1$, as any path has at most $n - 1$ edges.

For each $1 \leq i \leq k$, we have that $v_{i-1}v_i \in E_f$ so $\psi(v_{i-1}) \leq \psi(v_i) + 1$. Chaining these bounds together, we see

$$\psi(u) = \psi(v_0) \leq \psi(v_k) + k \leq \psi(w) + n - 1.$$

■

Lemma 4 *Let f be a preflow and ψ a distance label with respect to f. There is no $s - t$ path in G_f.*

Proof. If there was such a path, then Lemma 3 shows

$$n = \psi(s) \leq \psi(t) + n - 1 = n - 1,$$

which is impossible. ■

4.1.3 Running Time Analysis

Bounding the number of relabelings is easy.

Lemma 5 *The total number of relabeling steps is at most $2n^2$.*

Proof. We show each vertex is relabelled at most $2n - 1$ times.

The moment after v is relabelled to, say, $\psi'(v)$ it is still active. By Lemma 2, there is a $v - s$ path in $G_{f'}$ where f' is the preflow just after the relabeling. Lemma 3 then shows $\psi'(v) \leq \psi'(s) + n - 1 = 2n - 1$.

By Claim 1, any relabelling increases the label by at least 1 and each $v \neq s$ has initial distance label 0, so the distance label for v changes at most $2n - 1$. ■

To count the number of push operations, we distinguish between two types of pushes.

Definition 1 *A push operation along $vw \in E_f$ (where f is the current preflow) is called a* **saturating push** *if vw is not in the resulting preflow (equivalently, vw was augmented by $\mu_f(vw)$). Every other push operation is called a* **nonsaturating push***.*

Lemma 6 *The number of saturating pushes is at most $2mn$.*

Proof. Recall that E_R is the set of all possible edges and their reverses, so $|E_R| = 2m$. We show each $vw \in E_R$ is involved in a saturating push at most n times. To that end, let f, ψ denote the preflow and distance label at the start of a saturating push for vw and let f', ψ' denote the preflow and distance label at the start of the next push operation for wv (i.e. just before vw reentered the residual graph).

Because vw was admissible during the saturating push, $\psi(v) = \psi(w) + 1$. Because wv was admissible during the push that reintroduced vw to the residual graph, then $\psi'(w) = \psi'(v) + 1$. By Claim 1,

$$\psi(w) = \psi(v) - 1 \leq \psi'(v) - 1 = \psi'(w) - 2.$$

That is, the distance label for w increases by at least 2 between saturating pushes across vw. The proof of Lemma 5 shows the distance label for w never increases beyond $2n - 1$. This shows the number of times vw is involved in a saturated push is at most n. ■

Finally, it is possible to show the number of nonsaturating pushes is $O(n^2 m)$ no matter which active vertex is selected. With a bit more work, one can improve the running time bound if the active vertex with maximum distance label is selected.

Lemma 7 *If each iteration selects an active vertex v with $\psi(v) = \max\{\psi(w) : w \text{ active}\}$ then the number of nonsaturating pushes is $O(n^2 \cdot \sqrt{m})$.*

Proof. Let $\Psi := \max\{\psi(v) : v \text{ active}\}$. We view the algorithm as operating in **phases**, broken up by iterations where Ψ changes. See Figure 4.1 for a depiction of phases and how Ψ changes. Call a phase **cheap** if the number of unsaturating pushes in that phase is at most $\sqrt{m}$. Otherwise call the phase **expensive**.

Bounding the number of phases
Observe how Ψ changes. Each relabel operation increases Ψ, but the total increase of Ψ due to relabels is at most $2n^2$ because $\sum_v \psi(v) \leq 2n^2$ (as was shown in the proof of Lemma 5). In a push operation across an edge vw even if w becomes active then $\psi(w) = \psi(v) - 1 < \Psi$ so Ψ cannot increase.

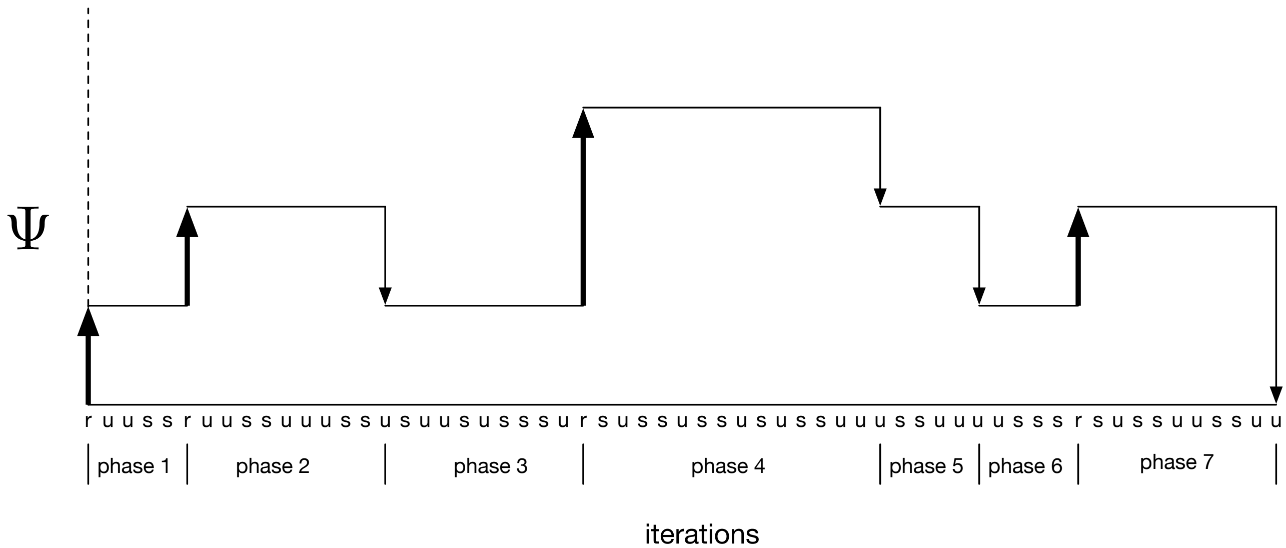

Figure 4.1: Depicting the change of Ψ over time. The bold edges are the increasing parts. The labels **s, u, r** denote an iteration with a saturating push, unsaturating push, and relabeling (respectively). A phase is the sequence of iterations between changes of Ψ (including the last iteration when Ψ changes). **Note**: it might not be possible to construct this exact sequence of increases and decreases of Ψ in an actual example, this is just depicting the relevant parts for the arguments in the proof.

Overall, the total increase is at most $2n^2$, Ψ is initially 0, and is never negative. So the number of steps where Ψ decreases is also at most $2n^2$. That is, there are at most $4n^2$ phases.

Unsaturating pushes in cheap phases
By definition and the bound on the number of phases in total, the total number of unsaturating pushes occuring in cheap phases is at most $4n^2 \cdot \sqrt{m}$.

Unsaturating pushes in expensive phases
We track this with another potential. Let

$$\Phi = \sum_{v \text{ active}} |\{w \in V : \psi(w) \le \psi(v)\}|.$$

Note $0 \le \Phi \le n^2$ always. We track how Φ changes.

When v is relabeled, the increase in Φ due to term v is at most n and no other term increases. Therefore, the total increase in Φ due to relabel steps is at most $O(n^3)$ (*c.f.* Lemma 5).

When a saturating push occurs along an edge vw, at most one new vertex becomes active and no distance labels change so the total increase in Ψ is at most n. Therefore, the total increase in Φ due to saturating pushes is at most $O(n^2 \cdot m)$ (*c.f.* Lemma 6).

Finally, an unsaturating push along an edge vw does not increase Φ because v goes inactive and w goes active (if it was not already). But every u with $\psi(u) \le \psi(w)$ satisfies $\psi(u) \le \psi(v)$ because vw is an admissible edge. So the increase in Φ due to w becoming active (if it does) at most the decrease due to v becoming inactive.

So the total increase of Φ plus the initial value of Φ is $O(n^2 \cdot m)$ (here we use assumption $n - 1 \le m$). What remains is to show that each unsaturating push in an expensive phase decreases Φ by at least $\sqrt{m}$. If so, then the number of such pushes is at most $O\left(\frac{n^2 \cdot m}{\sqrt{m}}\right) = O(n^2 \cdot \sqrt{m})$.

Expensive unsaturating pushes decrease Φ by $\sqrt{m}$
Focus on a single expensive phase and let S be the set of vertices v such that there was an unsaturating push along some edge in $\delta^{out}(v)$ in this phase. We have $\psi(v) = \Psi$ for each $v \in S$ because Ψ does not change throughout the phase and we always choose active vertices with maximum ψ value.

If vw is augmented in an unsaturating push then v goes inactive. The only way for v to become active again is for some push uv to happen, but this requires $\psi(u) = \psi(v) + 1$ so it cannot happen in this phase. Therefore, no two saturating pushes in this phase come from the same vertex, so $|S| \geq \sqrt{m}$.

Note that a relabeling step terminates a phase, so the distance labels ψ are the same throughout all unsaturating pushes in this phase. Let vw be an edge in an unsaturating push in this phase. Then $v \in S$ yet $w \notin S$ because $\psi(w) = \psi(v) - 1 < \Psi$. Therefore,

$$S \subseteq \{u : \psi(u) \leq \psi(v)\} - \{u : \psi(u) \leq \psi(w)\}.$$

In other words, the decrease in Φ due to v becoming inactive is at least $|S| \geq \sqrt{m}$ more than the increase due to w (possibly) becoming active. So this unsaturating push decreases Φ by at least $\sqrt{m}$. ■

Note: the choice of $\sqrt{m}$ in the definition of a cheap phase was to optimize the running time analysis. If we set it to some initially unspecified value α, we would have the number of unsaturating pushes in cheap phases is $O(n^2\alpha)$ and the number of unsaturating pushes in expensive phases is $O\left(\frac{n^2 m}{\alpha}\right)$ for a running time of $O\left(n^2\left(\alpha + \frac{m}{\alpha}\right)\right)$. So setting $\alpha := \sqrt{m}$ (asymptotically) minimizes the running time analysis.

4.1.4 Efficient Implementation

We comment on how to implement the algorithm to run in $O(n^2 \cdot \sqrt{m})$ time.

For each $0 \leq i \leq 2n-1$, let L_i be a doubly-linked list that holds all active vertices v with $\psi(v) = i$. Furthermore, for each nonempty L_i include a reference to the next and the previous nonempty linked list (think another doubly linked list whose entries are the lists L_i themselves). For each active v, keep a reference to the link in the appropriate list containing v.

When a vertex w is made active, we know that $\psi(v) = \psi(w) + 1$ so whether $L_{\psi(w)}$ is empty or not it is easy, using the previous and next references for the lists, to insert w into L_i (if needed) and update the references. When a vertex v is deactived (due to an unsaturating push) it is similarly easy to update these lists in constant time.

When a label $\phi(v)$ is increased, it is increased to $\phi(w) + 1$ for some w. It is a little harder to update the list pointers in constant time when moving v to the new list, but we can find the largest i between the old and new labels of v with $L_i \neq$ using a simple linear scan. For each v, the total time spent scanning over all relabels of v is $O(n)$ since different scans for v are over non-overlapping ranges.

Finally, we need to determine how to find an admissible edge quickly (or determine none exists). Each vertex keeps a list of potential edges to push along. When looking for a potential edge, scan through the list and discard those that are not admissible until an admissible edge is encountered. Every time v is relabelled (and also initially at the start of the algorithm), this list includes all possible edges that exit v.

When a push vw occurs, add wv to the list for w. There are $O(n)$ relabels for each vertex so the total time an edge is added to v's list due to a relabel is $O(mn)$. The total number of times some edge is added to some list due to a push is at most the number of push operations.