CO 351 : Network Flow Theory A10

Term: Spring 2019

Instructor: M. Pei

1. preflow push

   (a) We can assume that $s \in N_2$ WLOG because since $D = (N, A)$ is a bipartite graph and we characterized N_1 and N_2 to to be the two set of nodes that are disjoint sets it the doens't really matter which set s belongs to the reasoning will work the exactly the same. ie if $s \in N_1$ we would just rename N_1 and N_2 and the algorithm still holds.

   (b) At initialization we see that by the preflow push algorithm we push all the flow out of s. This means all arcs from s are now saturated and so in $D^{'}$ su does not exist and only us exists for all u connected to s in D. So no s,t dipath exists. As the algorithm runs the only method to create su in $D^{'}$ is to have to push excess back to s from u. By doing so arc su is no longer saturated in D. However, we only push back to s from u if all other arcs uv in D are saturated and there is still $e(u) > 0$ on u meaning that in $D^{'}$, only vu exists. This means that there are no forwards arcs coming out of u in $D^{'}$ so no u,t dipath exists and therefore no s,t dipath exists in $D^{'}$ in the algorithm.

   (c) At initialization s pushes all of its flow into its neighbours. Since D is a bipartite graph its neighbours are in N_1 by definition so $e(v) = 0$ for all $v \in N_2 \setminus \{s, t\}$. As the algorithm runs we see that it pushes excess if the proper height is maintained. Since all excess starts on a $u \in N_1$ by our intialization, it ends up on a $w \in N_1$ by the algorithm. Since in order to push there needs to be a uv in $D^{'}$ where $h(v) = h(u) - 1$ and a $vw \in D^{'}$ where $h(w) = h(v) - 1$ so every iteration of pushing requires us to push on a uw path. Pushing fron uv gives us excess on v which is in N_2 but the algorithm immediately requires a push from v to w where $w \in N_1$ The algorithm also requires the amount pushed to be $\min\{e(u), r_{uv}, r_{vw}\}$ so it is not possible to push more than the minimum residual of the two arcs making it impossible to leave excess on v. By induction this reasoning holds for $n + 1$ iterations so so $e(v) = 0$ for all $v \in N_2 \setminus \{s, t\}$ in the algorithm.

   (d) Assume for a contradiction that there exists a $h(v) > 4|N_1|$ and $v \in N_1$. This means $e(v) > 0$ and there exists a v, s dipath in $D^{'}$. This path has at most has the length of $|N| - 1$. Since $|N_1| \leq |N_2|$ we see that $|N_1| - 1 \leq \frac{1}{2}(|N| - 1)$ so $2(|N_1| - 1) \leq (|N| - 1)$. Since in order for the heights to be compatible the height needs to decrease by 1 in each arc, so $4|N_1| + 1 - (2(|N_1| - 1)) \geq 2|N_1| + 1$ which is a contradiction since $h(s)$ is initialized to be $2|N_1| + 1$.

2. preflow push 2

   (a) Relabel operations: Each node has at most height of $4|N_1|$ everytime we relabel a node in N_2 a node in N_1 has to be re-relabeled again according to the algorithm, so for every node in N_2 we relabel each node in N_1 twice. So the total number of relabel operations is $2|N_2| * 4|N_1| = 8|N_1||N_2|$
   Saturated pushes: Similar to the proofs from class in order to do another sat push on vw, w needs to be relabeled at least 4 times and to push from u to v v would be relabeled twice. Since each node has a max height of $4|N_1|$ there is at most $4|N_1|$ sat pushes since two sat pushes occur. There are $2|A|$ arcs so in total there are $8|N_1||A|$ saturated pushes.

   (b) We define a function $\Phi(x,h) = \sum(h(v)|v \in N_1, e(v) > 0)$ Where it is 0 at initialization and it is always non-negative.
   Proof: The effect of $\Phi(x,h)$ on each operation of this modified preflow-push algorithm
   Relabeling: A relabel operation is only done on nodes with excess units so only $v \in N_1$ which means $\Phi(x,h)$ increases by 1. And over the algorithm the maximum increase from relabeling is $8|N_1||N_2|$ by the result from part a
   Saturated pushes: Consider a saturated push on uv, vw this decreases $e(u)$ and increases $e(w)$. If $e(w) = 0$ before the push then after the push we add $h(v)$ to $\Phi(x,h)$. This adds at most $4|N_1|$ to $\Phi(x,h)$ since that is what we defined as the maximum height for any node in D. Over the algorithm we can see that the maximum increase is $8|N_1||N_2| + 4|N_1|(8|N_1||A|) \leq 32|N_1|^2|A|$ since there are at most $8|N_1||A|$ sat pushes with the maximum height being $4|N_1|$.
   Unsaturated pushes: Considering a non-saturated push then we see that $e(u)$ becomes 0 and $e(v)$ is 0 and $e(w)$ becomes positive. So we add $h(w)$ to $\Phi(x,h)$ and subtract $h(u)$ from $\Phi(x,h)$. However we also see that by the algorithm $h(u) = h(w) + 2$ so $\Phi(x,h)$ decreases by at least 2 per unsaturated push. Since by the end of the algorithm there should be no excess $\Phi(x,h)$ should be 0 at the end. But since we decrease by 2 for every unsaturated push there should be at most $\frac{1}{2}(8|N_1||N_2| + 32|N_1|^2|A|)$ unsat pushes. So there are at most $18|N_1|^2|A|$ unsat pushes. $(|N_1|^2 \leq |N_1||N_2|)$