Max Flow: Capacity Scaling

Assume that all capacities $c(e)$ are integers. Here we give an algorithm that takes $O(m^2 \log C)$ time, where $C = \max_{e \in E} c(e)$. We define a subgraph of the residual graph G_f as follows: $G_f(\Delta)$ keeps only those edges satisfying $r(e) \geq \delta$. Notice that $G_f(1) = G_f$.

Initially, let $\Delta = 2^{\lfloor \lg C \rfloor}$. Our algorithm can be described as follows.

> As long as there is a s-t path in $G_f(\Delta)$, augment as much flow as possible along such a path. When there is no more s-t path, divide Δ by 2. In case that $\Delta < 1$, we stop the algorithm.

Let us call a sequence of augmentations an *epoch* when Δ is fixed. The following observation is critical:

Lemma 1. *Each epoch has at most* $4m$ *pushes.*

Proof. We first make the following observation: when an epoch stops, the maximum flow is at most $2m\Delta$ larger than the actual flow f. To see this, note that when this epoch stops, $G_f(\Delta)$ is separated into two parts $(U, \overline{U})$, where $s \in U$ and $t \in \overline{U}$. For every edge $e \in \delta^+(U)$, $r(e) = c(e) - f(e) < \Delta$; conversely, $e \in \delta^-(U)$, $r(e) = f(e) < \Delta$. Therefore, denoting the actual and maximum flow values by $v(f)$ and v^* respectively,

$$v(f) = \sum_{e \in \delta^+(U)} f(e) - \sum_{e \in \delta^-(U)} f(e) > \sum_{e \in \Delta^+ U} c(e) - 2m\Delta \geq v^* - 2m\Delta.$$

where the last inequality we use the duality that a cut is always an upper bound of v^*.

In the next epoch, as each push augments the flow value by at least $\Delta/2$, we conclude that an epoch has at most $4m$ pushes.

□

The correctness of the algorithm is easy to see, since $G_f(1) = G_f$. After the last epoch, we essentially have no more augmenting path in G_f, thus having the same stopping condition as Ford-Fulkerson.

Finally, as there can be only $O(\lg C)$ epochs and each push needs $O(m)$ to reconstruct the residual network, we conclude that this algorithm takes $O(m^2 \lg C)$ time.

Application: A Theorem about Matrix

The following theorem does not have a specific name. It has a very elegant proof due to Lex Schrijver. Let A be a n-by-n $\{0, 1\}$ matrix, where each row and each column has exactly k ones. Then given any integer h, $1 \leq h < k$, there exists another matrix A', in which every row and every column has exactly h ones. Furthermore, $A'(i, j) = 1$ only if $A(i, j) = 1$. We now present a proof based on max-flow-min-cut theorem.

Construct a flow network as follows. Source s has a directed edge to each column node with capacity k. The column node i has a directed edge to row node j if and only if $A(i, j) = 1$. The capacity for such an edge is 1. Finally, the row node has a directed edge to t with capacity k. It is easy to see that the network has a max-flow of value nk. Now we modify the flow by multiplying it with h/k and change the capacity of edges from s to column nodes and from row nodes to t from k to h. The flow is still a max-flow in the new network (why?) and its value is hn, although it is not an integral flow. Recall that if all edge capacities are integers, we always have an integral maximum flow (whose value is hn in this case). The "integral" flow yields the matrix A' that we want.