Math 443/543 Graph Theory Notes 7: Pipelines, maximal and feasible flows on networks

David Glickenstein

October 15, 2008

1 Pipeline problems

Suppose you have several pipelines arranged in a complicated way (with junctions and multiple input and output). Each pipe has a maximum capacity. We might ask:

- What is the maximum amount of stuff (oil, water, electricity, etc) that can be moved by the network from the inputs to the outputs?
- Is a certain collection of assigned inputs and outputs able to be attained by adjustments in the flow through the pipes?

This work is mostly from BM Chapter 11.

2 Networks and flows

We will recall some definitions for networks and then talk about flows.

Definition 1 *A network N is a digraph G together with a capacity function $c : E_+(G) \to [0, \infty]$ and two disjoint sets of vertices $X, Y \subset V(G)$. The vertices X are called the sources and the vertices Y are called the sinks. Vertices in $G - (X \cup Y)$ are called intermediate vertices and denoted as I.*

Definition 2 *We will consider functions f from the directed edges $E_+(G)$ to some set of numbers (usually positive real or positive integer. We denote*

$$f(K) = \sum_{e \in K} f(e)$$

if $K \subset E_+(G)$. Suppose $S \subset V(G)$. Let (S, S^c) denote the set of all directed edges from vertices in S to vertices in $S^c = V(G) - S$. We denote

$$f(S, S^c) = f^+(S)$$
$$f(S^c, S) = f^-(S).$$

In particular, $f^+(v)$ *is the sum of all values of* f *on arcs from* v *and* $f^-(v)$ *is the sum of all values of* f *on arcs to* v*. Also note that* $f^+(S) = f^-(S^c)$ *and* $f^-(S) = f^+(S^c)$*.*

Definition 3 *A flow through a network* N *is a function* $f : E_+(G) \to \mathbb{Z}_{\geq 0}$ *such that*

$$\begin{aligned} f(e) &\leq c(e) \text{ for all } e \in E_+(G) \\ f^-(v) &= f^+(v) \text{ if } v \in I. \end{aligned}$$

We think of f as specifying the amount of stuff flowing through a particular directed edge in the network. The first condition says we cannot exceed the capacity of any one pipe. The second is a conservation condition, saying that everything enters and leaves the network via X and Y.

Definition 4 *If* $S \subset V(G)$ *and* f *is a flow then we define the* resultant flow out of S relative to f *to be*

$$f^+(S) - f^-(S).$$

Similarly, the resultant flow into S relative to f *is*

$$f^-(S) - f^+(S).$$

The resultant flow tells how much net stuff leaves S (like a flux). Note the following:

Proposition 5 *For any* $S \subset V(G)$ *and flow* f*,*

$$f^+(S) - f^-(S) = \sum_{v \in S} \left[f^+(v) - f^-(v) \right].$$

Note that it is not true that

$$f^+(S) = \sum_{v \in S} f^+(v).$$

Proposition 6 *The resultant flow out of* X *is equal to the resultant flow into* Y*.*

Proof. We know that $f^+(v) = 0$ if $v \in I$, and so

$$\begin{aligned} f^+(X) - f^-(X) &= \sum_{v \in X} \left[f^+(v) - f^-(v) \right] \\ &= \sum_{v \in Y^c} \left[f^+(v) - f^-(v) \right] \\ &= f^+(Y^c) - f^-(Y^c) \\ &= f^-(Y) - f^+(Y). \end{aligned}$$

■

Definition 7 *The* value of f *is defined as*

$$\operatorname{val} f = f^{+}(X) - f^{-}(X) = f^{-}(Y) - f^{+}(Y).$$

The value tells how much stuff is flowing through the network.

Definition 8 *A flow f on a network N is a* maximal flow *if there is no other flow on N with larger value.*

Thus a maximal flow is one which transmits the most stuff through the network.

Proposition 9 *For any network N, there is a new network N' such that $X' = \{x\}$, $Y' = \{y\}$, and there is a one-to-one correspondence of flows f on N and flows f' on N' such that*

$$\operatorname{val} f' = \operatorname{val} f.$$

Proof. Let N' be the network obtained from N by adding vertices x and y, arcs from x to each element of X and arcs from each element of Y to y. Give the new arcs capacity equal to infinity. Given a flow f' on N', there is an obvious subflow f on N. Given a flow f on N, we can construct the flow f' by setting

$$f'(a) = \begin{cases} f(a) & \text{if } a \in E_{+}(N) \\ f^{+}(v) - f^{-}(v) & \text{if } a = (x, v) \\ f^{-}(v) - f^{+}(v) & \text{if } a = (v, y) \end{cases}.$$

We see that $\operatorname{val} f' = \operatorname{val} f$. ■

For this reason, we will often confine ourselves to networks with a single source x and a single sink y.

Definition 10 *Let N be a network with a single source x and a single sink y. A* cut *in N is a set (S, S^c) of arcs where $x \in S$ and $y \in S^c$.*

Show examples (Figure 11.4, for instance).

Definition 11 *The* capacity *of a cut K is equal to*

$$\operatorname{cap} K = \sum_{a \in K} c(a).$$

A minimum cut *is a cut K such that there is no cut K' with* $\operatorname{cap} K' < \operatorname{cap} K$.

A minimum cut is like the "weakest link" in the chain. If one could turn the network into a linear path from x to y, the minimum cut would be the smallest capacity in that chain.

The key theorem about maximum flows and minimum cuts is the following.

Theorem 12 (Max Flow/Min Cut Theorem) *If f^* is the maximum flow and K_* is the minimum cut, then*

$$\operatorname{val} f^* = \operatorname{cap} K_*.$$

We will prove this soon, but first let's prove a more modest few things.

Lemma 13 *For any flow f and any cut (S, S^c) in N,*

$$\operatorname{val} f = f^+(S) - f^-(S).$$

Proof. We know that

$$f^+(x) - f^-(x) = \operatorname{val} f$$

and that

$$f^+(v) - f^-(v) = 0$$

for any $v \in S - x$. Thus we get that

$$\operatorname{val} f = \sum_{v \in S} \left[f^+(v) - f^-(v)\right] = f^+(S) - f^-(S).$$

■

Theorem 14 *For any flow f and any cut $K = (S, S^c)$ in N,*

$$\operatorname{val} f \leq \operatorname{cap} K.$$

Equality holds only if and only if $f(a) = c(a)$ for all $a \in (S, S^c)$ and if $f(a) = 0$ for all $a \in (S^c, S)$.

Corollary 15 *If f^* is the maximum flow and K_* is the minimum cut, then*

$$\operatorname{val} f^* \leq \operatorname{cap} K_*.$$

Note, we have proved one half of the Max Flow/Min Cut Theorem. The other inequality will be proven later.

Corollary 16 *If f is a flow and K is a cut such that $\operatorname{val} f = \operatorname{cap} K$, then f is a maximum flow and K is a minimum cut.*

Proof. We have that

$$\operatorname{val} f \leq \operatorname{val} f^* \leq \operatorname{cap} K_* \leq \operatorname{cap} K,$$

but the assumptions imply that these are all equalities. In particular, f is a maximum flow and K is a minimum cut. ■

Corollary 17 *For any flow f and any cut $K = (S, S^c)$ in N, if $f(a) = c(a)$ for all $a \in (S, S^c)$ and if $f(a) = 0$ for all $a \in (S^c, S)$, then f is a maximum flow and K is a minimum cut.*

Proof of Theorem 14. We know that

$$\begin{aligned} f^+(S) &\leq \operatorname{cap} K \\ f^-(S) &\geq 0 \end{aligned}$$

so

$$\begin{aligned} \operatorname{val} f &= f^+(S) - f^-(S) \\ &\leq \operatorname{cap} K. \end{aligned}$$

The equality is if $f^+(S) = \operatorname{cap} K$ and $f^-(S) = 0$, so the second statement follows. ■

3 Proof of Max Flow/Min Cut Theorem

In this section, we will consider the following types of paths (which are different from directed paths considered earlier).

Definition 18 *A* v_0v_{k+1}-semipath *is a list* $v_0, a_0, v_1, a_1, v_2, a_2, v_3, \ldots, a_k, v_{k+1}$ *where* v_i *are vertices and* a_i *are arcs such that either* $a_i = (v_i, v_{i+1})$ *or* $a_i = (v_{i+1}, v_i)$, *and no vertex is repeated. Arcs of the first type are called* forward arcs *and arcs of the second type are called* reverse arcs*.*

We note that given a flow f on a network N together with a semipath P from x to y, we can produce a new flow $\tilde{f}$ by making

$$\tilde{f}(a) = \begin{cases} f(a) + \varepsilon & \text{if } a \text{ is a forward arc} \\ f(a) - \varepsilon & \text{if } a \text{ is a reverse arc} \\ f(a) & \text{otherwise} \end{cases},$$

as long as $f(a) + \varepsilon \leq c(a)$ and $f(a) - \varepsilon \geq 0$. The construction is designed to ensure that $f^+(v) = f^-(v)$ if $v \in I$.

We will now consider a way to use these semipaths to increase the value of a flow. For a xy-path P, define

$$\iota(a) = \begin{cases} c(a) - f(a) & \text{if } a \text{ is a forward arc in } P \\ f(a) & \text{if } a \text{ is a reverse arc in } P \end{cases}$$

and define

$$\iota(P) = \min_{a \in P} \iota(v).$$

Note that $\iota(a)$ is how much we can increase the forward flow or decrease the backward flow. We can now choose a new semipath

$$\hat{f}(a) = \begin{cases} f(a) + \iota(P) & \text{if } a \text{ is a forward arc} \\ f(a) - \iota(P) & \text{if } a \text{ is a reverse arc} \\ f(a) & \text{otherwise} \end{cases}.$$

Note that $\hat{f}$ is a new flow, since it satisfies the conditions to ensure $0 \leq \hat{f}(a) \leq c(a)$. Also note that

$$\operatorname{val} \hat{f} = \operatorname{val} f + \iota(P).$$

Theorem 19 *A flow* f *is a maximum flow if and only if* N *contains no* xy*-semipaths* P *with* $\iota(P) > 0$.

Proof. If N contains such a semipath P, we have shown how to increase the value of f, and so f is not a maximum. Now suppose N contains no such semipaths. We let S be the set of all vertices v such that there is a xv-semipath P_v such that $\iota(P_v) > 0$, together with x. We know that y is not in this set (by assumption), and so (S, S^c) is a cut. We will now show that each arc in (S, S^c) satisfies $f(a) = c(a)$ and every arc in (S^c, S) satisfies $f(a) = 0$. By Corollary, 17 this would imply that f is a maximum flow. Now suppose $a \in (S, S^c)$ and $a = (v, w)$. Then There is a xv-path P_v in N such that $\iota(P_v) > 0$. if $f(a) < c(a)$, then we could extend P_v to a xw-path, so we must have that $f(a) = c(a)$. Similarly, if we have $a \in (S^c, S)$ and $a = (w, v)$, then if $f(a) > 0$ then we could extend P_v to a xw-semipath. This completes the proof. ■

Thus, in the process of the proof, we have shown that, given a flow, we can construct a maximum flow by incrementally considering xy-semipaths P with $\iota(P) > 0$ (these are called f-incremental paths in BM), finding new flows $\hat{f}$, and continuing until there are no such semipaths left. This flow will be a maximum and its value will be equal to the minimum cut, also shown in the proof. Thus, we have proven the Max Flow/Min Cut Theorem.

4 Feasible flow theorem

Consider a network N. Suppose that for each source $x \in X$, we are given a nonnegative integer $\sigma(x)$ called the supply and for each sink $y \in Y$ we are given a nonnegative integer $\delta(y)$ called the demand.

Definition 20 *A flow f on N is* feasible *if*

$$f^+(x) - f^-(x) \leq \sigma(x)$$

and

$$f^-(y) - f^+(y) \geq \delta(y)$$

for all $x_i \in X$ and $y_j \in Y$.

In other words, the flow is feasible if it sends out less than the supply at each source and receives at least the demand at each sink. We would like natural conditions on the supply, demand, and network such that a feasible flow exists. For the sequel, for $S \subset V$, define

$$\sigma(S) = \sum_{v \in S} \sigma(v)$$
$$\delta(S) = \sum_{v \in S} \delta(v).$$

Theorem 21 *There exists a feasible flow in N if and only if for all $S \subset V$,*

$$c(S, S^c) \geq \delta(Y \cap S^c) - \sigma(X \cap S^c).$$

Proof. Construct a new network N' by:

1. Add two new vertices x and y to N
2. Join x to all $x_i \in X$ by arcs with capacity $\sigma(x_i)$
3. Join all $y_j \in Y$ to y by arcs with capacity $\delta(y_j)$

4. Designate x as the source and y as the sink for N'.

If N' has a flow f' such that $f'(y_j, y) = c'(y_j, y) = \delta(y_j)$ for all $y_j \in Y$, then restricting to N we get a feasible flow on N. Clearly, if such a flow exists, it has value equal to $\delta(Y)$, which is equal to the capacity of the cut $(N \cup \{x\}, \{y\})$, and thus it is a maximum flow. If we knew that $(N \cup \{x\}, \{y\})$ is a minimum cut, then we would know that the max flow has this property, and thus there is a feasible flow. The property that $(N \cup \{x\}, \{y\})$ is a minimum cut is that for any subset $S \subset N$,

$$\operatorname{cap}(S \cup \{x\}, S^c \cup \{y\}) \geq \operatorname{cap}(N \cup \{x\}, \{y\}) = \delta(Y)$$

(note, we think of S^c as the complement in N). We now note that

$$\begin{aligned}\operatorname{cap}(S \cup \{x\}, S^c \cup \{y\}) &= \operatorname{cap}(S, S^c) + \operatorname{cap}(\{x\}, S^c) + \operatorname{cap}(S, \{y\}) \\ &= \operatorname{cap}(S, S^c) + \sigma(X \cap S^c) + \delta(Y \cap S),\end{aligned}$$

so the condition is

$$\operatorname{cap}(S, S^c) + \sigma(X \cap S^c) + \delta(Y \cap S) \geq \delta(Y),$$

or

$$\operatorname{cap}(S, S^c) \geq \delta(Y \cap S^c) - \sigma(X \cap S^c).$$

■

5 Max Flow/Min Cut algorithm

Now we look at an algorithm for finding the maximum flow. It is originally due to Ford-Fulkerson and called the labelling method.

Recall that we wish to find xy-semipaths P such that $\iota(P) > 0$. We need a systematic way of finding these paths. One method is to construct a tree T of all paths P of this form starting at x. If y is not in this tree, we know that our flow is a max flow. If y is in the tree, then we can make the flow larger as in the proof of the Max Flow/Min Cut Theorem. We then do this and start over.

Here is how we construct the tree T. We make T a labeled tree, meaning that each vertex v in T has a label $\ell(v)$. $\ell(v)$ will be equal to $\iota(P_v) > 0$ for the unique xv-semipath in T.

1. Let $\ell(x) = \infty$. Let $L = \{x\}$, the set of labeled vertices.

2. If $a = (u, v)$ is an arc such that u is labeled (i.e., $u \in L$) and v is not labeled, then label $\ell(v) = \min\{\ell(u), c(a) - f(a)\}$ if this is positive and add v to L.

3. If $a = (v, u)$ is an arc such that u is labeled and v is not labeled, then label $\ell(v) = \min\{\ell(u), f(a)\}$ if this is positive and add v to L.

4. Continue to steps 2 and 3 until y is labeled or all labeled vertices have been scanned and no new vertices may be added to L.

5. If $y \in L$, then replace flow f with $\hat{f}$ and go to 1.

6. If $y \notin L$, then the flow is a maximum.

We note that this produces a tree (since we only add arcs if the new vertex is not already in the tree). The algorithm is correct by the Max Flow/Min Cut Theorem. Note that this algorithm is not good, but one can improve it by doing a breadth-first search (adding vertices into a queue and then testing 2 and 3 using the queue. See BM Figure 11.7.

Note that we can use this algorithm to also solve the feasible flow problem! Just augment the network as in the proof and if the max flow meets the demands, then there is a feasible flow.

6 Applications

(This is taken from a text by W.D. Wallis) A company has two factories F_1 and F_2 producing a commodity sold at two retail outlets M_1 and M_2. The product is marketed by four distributors a, b, c, d. Each factory can produce 50 items per week. The weekly demand at M_1 and M_2 are 35 units and 50 units respectively. The distribution network is given below. Can the weekly demands be met?

Picture to be done later.