

Advanced Algorithms and Data Structures

Hand-in 1

Philip Pickering
email@here.com

Thomas Bracht Laumann Jespersen
ntl316@alumni.ku.dk

The problem could be formulated and solved as a max-flow problem. There are a few things to observe before that: the network (as in the example) is undirected, in which case we would want to make it directed. We also have several sources that may themselves be connected (such as nodes 2 and 3 in the example).

Let the network be an undirected graph called $N = (V, E)$. We want to construct a max-flow graph $G = (V', E')$ from N . For all nodes $u, v \in V$ and $(u, v) \in E$, we add to our max-flow graph nodes u, v and an extra node w . We add directed edges $(u, v), (v, w)$ and (w, u) . If we let k denote the capacity of the edge $(u, v) \in E$, then we let $c(u, v) = c(v, w) = c(w, u) = k$ in G . Lastly, we add a super source s and an edge (s, v) to all the nodes v that were a source in N .

The number of edges in G is $|V'| = |V| + |E| + 1$ and $|E'| = 3|E| + p$, where p is the number of edges added from the super source s . p is exactly the number of sources in N .

Now we have the original problem stated as a max-flow problem, which can be easily expressed as a linear program

$$\begin{aligned} \max.: \quad & \sum_{v \in V_G} f(s, v) - \sum_{v \in V_G} f(v, s) \\ \text{s.t.:} \quad & f(u, v) \leq c(u, v), \quad \forall u, v \in V_G \\ & \sum_{v \in V_G} f(v, u) - \sum_{v \in V_G} f(u, v) = 0 \quad \forall u, v \in V_G \\ & f(u, v) = 0, \quad \forall u, v \in V_G \end{aligned}$$

As small example graph can be found in fig.1.

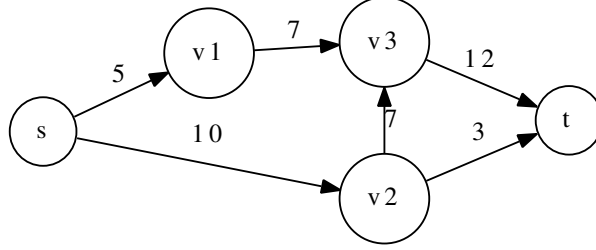


Figure 1: An example max-flow graph

The linear program corresponding to the graph in fig.1 is the following:

$$\begin{array}{ll}
\text{max.:} & f_{s,v_1} + f_{s,v_2} \\
\text{s.t.:} & f_{s,v_1} \leq c(s, v_1) = 5 \\
& f_{s,v_2} \leq c(s, v_2) = 10 \\
& f_{v_1,v_3} \leq c(v_1, v_3) = 7 \\
& f_{v_2,v_3} \leq c(v_1, v_3) = 7 \\
& f_{v_2,t} \leq c(v_2, t) = 3 \\
& f_{v_3,t} \leq c(v_3, t) = 12 \\
& f_{s,v_1} = f_{v_1,v_3} \\
& f_{s,v_2} = f_{v_2,v_3} + f_{v_2,t} \\
& f_{v_1,v_3} + f_{v_2,v_3} = f_{v_3,t} \\
& f_{s,v_1}, f_{s,v_2}, f_{v_1,v_3}, f_{v_2,v_3}, f_{v_2,t}, f_{v_3,t} \geq 0
\end{array}$$

We have minimized the above linear program, such that edges with capacity zero have been omitted. Without showing the work of the simplex algorithm, we arrive at the solution: $f_{s,v_1} = 5, f_{s,v_2} = 10$.

The dual of the above stated problem can be stated by restating the above program in standard form $Ax \leq b$ along with a coefficient vector $c = (1 \ 1 \ 0 \ 0 \ 0 \ 0)^T$. The dual is then given by $A^T y \geq c$, where we minimize $b^T y$. y is a vector of size proportional to the number of constraints, in our case 12, because each of the equality constraints give rise to two inequalities. Now we can state the dual:

$$\begin{array}{ll}
\text{min.:} & 5y_1 + 10y_2 + 7y_3 + 7y_4 + 3y_5 + 12y_6 \\
\text{s.t.:} & \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} y \geq \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}
\end{array}$$

In order to find cheapest critical connection,

1. $S = \{s\}, T = V - \{s\}$
2. $S = \{s, v_2\}, T = V - S$
3. $S = V - \{t\}, T = \{t\}$

1. Give a precise explanation how the company could efficiently determine the largest transfer rate to the server.
2. Write a linear program that finds this transfer rate
3. Give a small example graph and write the optimal solution to the linear program given above.
4. The cost of increasing the capacity of a connection with one unit is directly proportional to the capacity of that connection.

Give a precise explanation how the company could efficiently determine the cheapest 'critical connection' to improve in order to get a better maximum transfer rate from the workstations to the server.

5. Write the dual of the linear program you gave before and find the optimal solution in the example you created before.
6. How could the critical connections be determined from the dual?
7. Some of the relay stations are not owned by the company, so they have to pay a leasing price of 1 øre for each byte transferred in and out of the company's network (see Figure 2 for an example).

Assuming that the company has already determined the maximum transfer rate, and wants to maintain this, give a linear program that finds the lowest possible price they will have to pay for it.

8. Attached is a Java class (Network.java) specifying a network as just described.

Report both the largest bit rate and the lowest leasing-price for this problem and give a brief description of how you obtained this result. It is preferable if you use a linear programming package that can be called from your programming language of choice. The second assignment will require this as well.