HW8 - S15
Warmuth

Justin Wong (jujwong)

6/4/15

1 Disjoint Paths

The original algorithm for finding the max number of edge-disjoint paths is Menger's Theorem, if there is a flow of k then there are at least k edge-disjoint paths.. First finding max flow and using the min number of edges whose removal will disconnect t from s. The min number of cuts is the max number of edge-disjoint paths.

First we will find the maximum flow in our graph G. Then starting from s walk towards t. If there is a cycle, where the path will use a node we have previously visited and will make us loop through to the same edges, then we will reduce the flow at the cycle to be 0. All the edges in the cycle will be 0, effectively removing it from our graph. When t is reached keep track of the path.

The flow of the overall graph should not change because although less edges will have flow on them the amount of flow reaching the sink should be the same. So the number of edge-disjoint paths is still equal to the flow and cycles will be eliminated. The runtime is in O(nf) where n is the number of edges and f is the maximum flow, the algorithm is mostly bounded by the Ford-Fulkerson run in order to figure out the maximum flow.

2 Alternate Hall's Theorem

In a graph with both n men and women where each woman knows k men and each man knows k women, so on the left side we will label the vertices W_i and the right will be labeled M_i. That is show that for k-regular graphs the following holds: For every subset S of the women, |N(S)| $\geq$ | S |, where N(S) is the total set of men they know together.

Let E be the number of edges leaving S and matching k vertices in M being $|S| * k$. $|N(S)|$ is the number of women in the subset, and each women knows k men so let B the sum of the $|N(S)| * k$. B $\geq$ E so all cases.

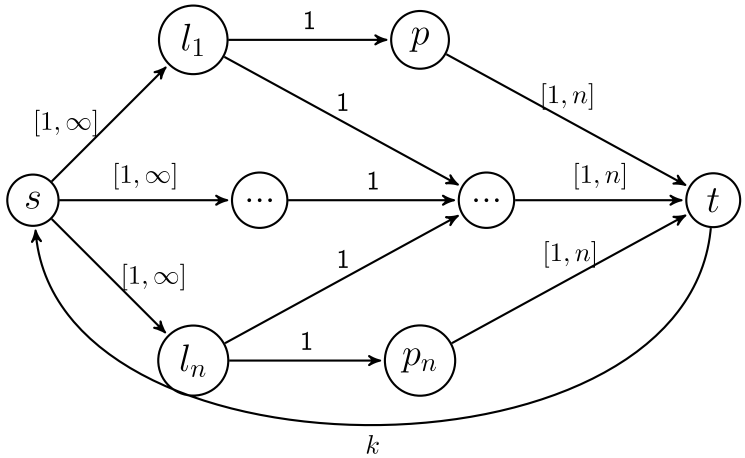
3 KT 12, Page 420

G = (V,E) is a directed graph with a source s in V and a sink t in V. Each edge e in E has a capacity of 1 and k is a paramter passed to us. We want to delete k edges to reduce max flow by as much as possible. Use a set of F subset of E, so that length of F = k and the maximum flow of G' = (V, E-F) is small as possible. f is the value of the max flow.

The only way to change the maximum flow will be to affect the min cuts, due to the max flow-min cut theorem and also that changing now min cut edges will not change the flow of the graph. So if the minimum s-t cut already has size $\leq k$ then it is possible to reduce the flow down to 0. By first finding the minimum cut set labelled {A,B} we will remove edges leaving A and entering B, as these are the flow values that we care about and will force the flow to adjust accordingly because we are now limiting the amount of flow that is possible.

The graph of G' where the set of edges F is the size of k, will have a flow of at least $f - k$. In the cut set of A,B there are at least f edges leaving A into B, thus with the k deletions there are at least $f - k$ edges leaving A. The max flow will have to have changed by at least $f - k$ since the minimum cut will be now at least $f - k$.

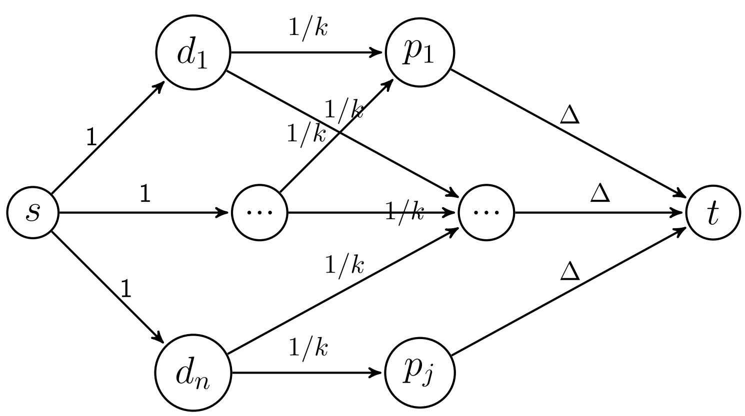
4 KT 21, Page 427



(a)If it is possible to connect at least one laptop connect to at least each access point once, then max flow will be equal to the number of edges between p and a being $|a+p|$.

(b)The s to laptop edges have capacity of [1,∞] as a way to guarantee that every laptop will get a flow. If there is a valid circulation, then the set amount of flow will be at most k. If there is a test set I will be able to create a circulation and my algorithm will use Ford-Fulkerson to ensure that there is a possible max flow of k. The amount of flow going into t has to $\leq$ k. Else if it is not possible to create a valid ciculation there will be no test size of k. This algorithm will run in O(Ef) time where E is the number of edges in the graph and f is the max flow.

5 KT 27, Page 431 Carpool



The edges connect source to days represents the flow needed for one day, 1 driver is neccessary for each day. The edges between days to drivers is the people that need to carpool that day, and so the edge capacity is $1/k$ meaning that the chances of the person having to drive is split between the number of people going to work that day. The Δ connecting the person to sink is the driving obligation that a particular person would have to drive, for example if person 1 had work both on day 1 and 2 and on day 1 they had 3 people in the carpool and on day 2 they had 2 the Δ value would be $1/3 + 1/2$. In addition each driver can have a max of Δ_j

meaning that can't be chosen to drive every day.

The integrality theorem is used due to the $1/k$ between days and drivers, only 1 person can be chosen. So the flow leaving from the day will have a 1 on the person that was chosen and the flow entering the drivers on the right will be 1. Thus when we take a subset of days for our schedule, we know the amount of flow we will need to be equal to the number of the days. The fairness of the schedule is now possible because each person will be required to fulfill their driving duties as the edge leaving each driver to sink corresponds to the number of days that they are in the carpool.

The algorithm runs Ford-Fulkerson to decide if there is a possible max flow of d where d is the number of days run for the schedule. It will run in O(kd) time due to the number of days run for and the possible number of drivers each day.

6 KT 45, Page 444

The counties will represent nodes in a network flow graph with demands. We know each country has a budget surplus s_i where if it is negative then that country is in a deficit. So the countries with a positive budget surplus are known as supply nodes and the negative si countries are demand nodes. The edges between countries are known from e_{ij} which are the trade values of exports from country i to country j, these values will always be non-negative.

To begin we will create a super source, s, and connect it to all the countries will positive si. For countries will negative si we will connect them to a super sink, t. Now we are able to find the value of a the flow in the graph if all edges leaving s and entering t are saturated. Now looking for a subset of countries we want to take the sum of the countries' budget surpluses in the cut and let that be represented by S. Edges leaving the country subset to countries that are not in the subset are the edges we want to examine, as they are the export values the total value represented by E.