Adam Paz
1363522

Homework 4

1 Crayons

We build the following network with max flow of 4n crayons, where n is the number of kids. We have two sets of kids connecting to the source, so each kid shows up twice. The 1st list of the students each has an arc of max flow 3 and the 2nd list of kids has max flow of 1. The 1st list are connected to the crayons each with an arc of value 1. And the 2nd list of students are all connected to the subjects that they like with an arc of value 1. From their each crayon color has an arc to the sink with max flow t_j where j is a crayon in the list of k crayon colors. This number represents the amount of crayons of this color. The arcs from the subject to the sink also has a max flow of 2, because each subject can have to kids. This also means that there must at least be $n/2$ subjects where n is the number of kids in the class.

We will solve this problem with an instance of max flow.

(i) If the flow in this graph is 4n then we will find that it will follow that all of the kids will get a color and a subject that they enjoy. The minimum cut in the graph is 4n. If the cut between the source and the kids is 4n then all of the arcs will be saturated and it will follow that each kid will get one of 3 colors because the flow of three then meets arcs of flow 1. Which will create an outwards flow of 3n towards the sink, we add this with the n that will flow from students to subjects to the sink. Therefore the cut between the crayons/ subjects to the sink is 4n which is equal to the starting max flow and therefore there is conservation of flow.

(ii) We will prove that not having a flow of 4n will not produce happiness by contrapositive. For an assignment to make all of the kids happy with their crayons each one must receive exactly three colors that are on their list. Therefore each kid must have a flow of exactly three from the source. This flow of three must be split up amongst three of the color nodes that the specific kid is attached to and when ford-fulgerson is run it will because the following arcs have max capacity of 1. In order for the kids to be happy with the subject they receive they must be connected to all the subjects that they like, and a flow of exactly 1 will come from all n kids and if more than 2 students are attached to one subject, some of the kids will not get this subject because it has an arc of only 2 to the sink. With the addition of the flow from all the kids to their crayons of choice and all the kids to their subjects of choice we find that there must be a flow of 4n in order for the flow to work and for all the kids to be happy.

2 Cows

We build the following flow network where the flow must be n, where n is the amount of cows. This is a bipartite graph where we connect our source to the cows, where each arc has a max flow of 1. From their each cow is connected to all the troughs that are within 2 miles. Also having arcs of max flow 1. Then from the troughs to the sink, we have a arc with max flow of$\lceil n/w \rceil$ in order to deal with the balancing and creating an even spread of cows. We only can only send a cow to a trough if the trough does not already have a saturated arc to the sink.

We will solve this problem with an instance of max flow.

(i)If the flow is n then we can create a flow from cows to troughs that makes all the cows and the farmer happy. Due to the integrality theorem we can ask that the flows all be integers and the algorithm will still work. Instead of using fractions we either get a 0 or 1 telling us whether the cow didn't go to or went to the specific trough. Since there are n cows it is easy to follow that the cut between cows and source will be n, and since cows cannot be split and they are integers going into the process they will also be integers going out. And the cut between cows and troughs must also be n. Since there are w troughs and each trough can have at most $\lceil n/w \rceil$ cows. It follows that this cut produces a flow of n and since the beginning cut also provides a flow of n, the algorithm will terminate and return that it is possible for everyone to be happy if and only if the flow is n.

(ii) In order to not get a complaint we can flow 1 to each cow and flow 1 from each cow to a trough, and have a perfectly even spread of cows to troughs. With this case we can create a flow of exactly n, because we have the n cows that get their resource from the source, and then pass this single resource on to the troughs. Since each cow can at max send 1 it will only choose 1 trough to go to due to the integrality theorem. The troughs will collectively receive exactly n cows all together because $w * \lceil n/w \rceil = n$. And the cows will be evenly spread due to the fact that each trough is connected to the sink with a maximum capacity of $\lceil n/w \rceil$. Therefore flow is conserved and n is a solution to a working flow in this network. Therefore by contrapositive if the either the cows or the farmer is unhappy there is not a flow of n.

3 Teams with Artists

We build the following flow network where the flow must be a_j for all $j \in \{1, 2, ..j, ...n\}$ where n is the amount of teams and a_j is the amount of artists assigned to a specific team. We have arcs with max flow 5 linked from the source to every artist. Because each artist has the resources to at max join 5 teams, but also has the choice to join no team. Each Team has it's own set of nodes for each college. And each artist only connects to the college that they are from if they are compatible with the team that the college corresponds with. There is a flow of 1 from each artist to their college for each team they are compatible with, because there is only one artist. Each college however can send up to $a_j - 1$ artists to the team it corresponds with because there must be at least one artist from another college joining this team. They have a minimum size of a_j because each team needs at least a_j artists. From the team to the sink there is an arc of max size $m - 2 * n$, where m is the amount of artists and n is the amount of teams,because each team must have at least 2 artists.

We will solve this problem with an instance of max flow

(i) the max flow would be the some of a_j for all $j \in \{1, 2, ..j, ...n\}$ where n is the total amount of teams. With this flow we will find that since each of the artists have a maximum of 5 resources to pass out, there is an arc of max flow 5 connecting the source to the artists. Each artist can then pass exactly 1 flow to the college they belong to that corresponds to the college they belong to, which then passes this 1 flow on to the team. However, since the arc from the specific college to the team is $a_j - 1$ if the arc is saturated, the arc from the team to the sink still requires one more artist because it has a minimum flow of a_j. Therefore the other 1 flow will have to come from another college. This flow of a_j is achieved by the artists' resources because that is the flow that is being pushed through the graph. And it will collect at the sink as well because each team's arc to the sink has a minimum bound of a_j. Therefore the cuts between the sink and the artists, and the cut between the teams and the sink are equal and are equal to the max flow.

(ii) We will prove that not having a flow of a_j will not produce happiness amongst teams and artists by contrapositive. So given an assignment in which each team gets exactly a_j artists, it means that the colleges that have arcs attached to the teams will have a sum of a_j, which comes from the arcs from artists to the colleges. Since each artist is attached to it's college and cannot change colleges it is necessary to have a list of all colleges for all teams in order for each team to have at least 2 different colleges with artists. Since the cut from colleges to their corresponding team is a_j their sum will be the total flow of the sum of all a_j for each team. Therefore this cut is exactly the summation of a_j for all j. It can achieve this because each college can pass on multiple artists up to $a_j - 1$ because the extra one must come from a separate college. From these teams to the sink there will be a flow of a_j and since there are n teams a flow of a_j for all $j \in \{1, 2, ..j, ...n\}$ is the minimum that will satisfy all the teams with having the correct amount of artists.

Adam Paz
1363522

4 Ancient Rome

If f is the max flow of a network of aqueducts, then to increase the v(f) while only changing one arc we must make use of the residual graph $G_r(f)$. First we find the min cut (A,B) of our graph f. From there we use the residual graph to discover the smallest saturated arc in this minimum cut. From there we follow this arc to the sink, and we trace back from the arc to the source. If all other arcs in a single path that includes arc and reaching both sink and source are non-saturated then increasing this arc will increase the v(f). If there exists another arc that is saturated in every path from source to sink which includes this saturated arc then increasing the flow in this arc will not increase v(f). Therefore we move on to the next smallest saturated arc in the min cut until we have exhausted all options of saturated arcs in the min cut. If there is no path from the source to the arc or the arc to the sink that can fully be reached by non-saturated arcs then there is no way to increase the v(f) by only increasing a single arc. The exception to this is if there is an arc directly connecting sink and source. Increasing this will always increase the v(f).

Timing Lets say there are E edges in this network. And there is a min cut with k edges. So in the worst case we must trace from every arc in the min cut back to the source and forward to the sink. Once we see that a part of any path is saturated we see that we cannot take any other part that is exclusively connected to this arc. We check every arc in the min cut and check all edges that form paths connected to these from source to sink. Since we only must check each edge once in order to tell if the path doesn't work, in the worst case we must reach every single edge. Therefore it is O(E), where is E is the amount of edges in our graph.