CSC373H1 Summer 2014 Assignment 3

Names: John Armstrong, Henry Ku

SNs\CDF username: 993114492\g2jarmst, 998551348\g2kuhenr

Question #	Score
1	
2	
3	
4	
5	
6	
7	
8	
Total	

Acknowledgements:

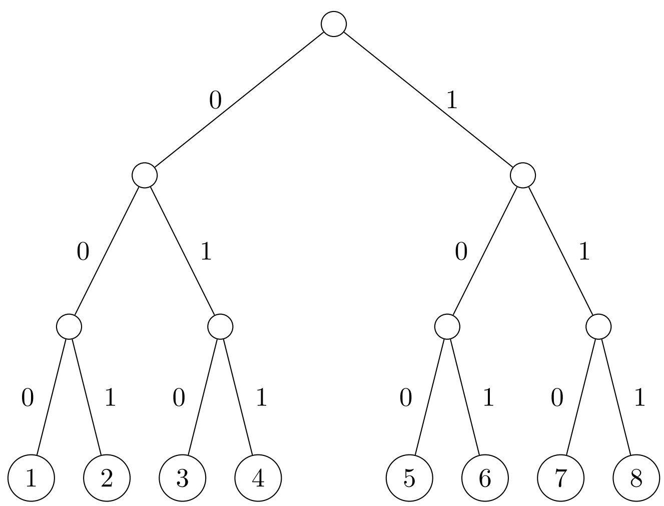
"We declare that we have not used any outside help in completing this assignment."

Name: John Armstrong, Henry Ku Date: July 15, 2014

Q1. Suppose G(V, E) is representative of a network where V = {s, t, v_1, ..., v_n} and E is the set of edges that make up this network. Suppose we have sets S and T such that s $\in$ S, t $\in$ T, (S $\cup$ T) - {s, t} = V, and S $\cap$ T = $\emptyset$. Also, suppose that for all possible arrangement of vertices in S and T, c(S, T) = f(S, T), which means that G(V, E) has an exponential number of minimum cuts between s and t. Precisely, 2^n minimum cuts as each vertex is either in S or T and cannot be in both, thus it is much like a binary string of |V| elements, which can be expressed in 2^n possible ways.

To construct a graph that is representative of there being 2^n possible minimum cuts, a tree like structure seems appropriate since there are 2^n leaves in a full and complete tree for n levels in the tree. Taking inspiration from Huffman encoding, we wish to have a graph that for each leaf in a full tree will represent a unique binary string that represents whether a vertex is in S or T. Suppose we arrange the the string as follows, v_1, ..., v_n, where if v_i is in S then $v_i = 0$ and $v_i = 1$ if it is in T. The tree/graph can be constructed so that each left edge appends 0 to the vertex string, and WLOG each right edge appends 1. Finally, when reaching any leaf we have an n length binary string that represents a unique cut S, T. In total there are 2^n such possible cuts.

For example, suppose we had |V| = 3 and we are certain that there are an exponential number of minimum cuts. We construct a full and complete binary tree as follows to represent the minimum cuts:



If we were to trace each leaf from the node, in a similar fashion to Huffman encoding, we would have 1: 000, 2: 001, 3: 010, 4: 011, 5: 100, 6: 101, 7: 110, and 8: 111. So, it is clear that each binary string representing v_1, v_2, v_3 will represent all possible sets S, T such that in total the constructed graph can represent an exponential number of minimum cuts.

Q2. Claim:

Function f, such that f(S) is the number of edges (u, v) with $u \in S$, $v \in V \backslash S$, is submodular.

Proof:

Using the fact provided in the question, all that is necessary to show that f is submodular is to show that for any two subsets A, $B \subseteq V$, $f(A) + f(B) \geqslant f(A \cup B) + f(A \cap B)$.

We begin by establishing a fact. Suppose you have disjoint sets X, $Y \subseteq V$. Now suppose in sets X and Y each have a selection of m + n vertices in G(V, E). In X, there are m vertices that appear in edges in G(V, E), these m are startpoints of edges. In Y there are n vertices that that appear in edges in G(V, E) and are startpoints. However, in X there are n vertices that are the endpoints to all edges with m startpoints in Y, similarly for Y but m vertices endpoints to X's m startpoints. Indeed, $f(X \cup Y) = 0$, as this has been designed such that there are no edges (u, v) such that $u \in (X \cup Y)$, and $v \in V \backslash (X \cup Y)$. In this case, it follows that $f(X) + f(Y) > f(X \cup Y)$. Now suppose we remove from X the n endpoint vertices, and from Y remove the m endpoints. It would then follow that $f(X) + f(Y) = f(X \cup Y)$. Each of these cases represent the extreme scenarios, and we can conclude that $f(X) + f(Y) \geqslant f(X \cup Y)$.

Now, suppose we have some arbitrary subsets A, $B \subseteq V$. Indeed it is possible to seperate both A and B as follows. Let A' represent all vertices that do not appear in B from set A, and let A" be those vertices that do appear in B. Similarly, Let B' represent all vertices that do not appear in A from set B, and let B" be those vertices that do appear in A. Then, f(A) + f(B) = f(A') + f(A") + f(B') + f(B"). Clearly, A" = B" = $(A \cap B)$, then $f(A) + f(B) = (f(A') + f(A \cap B) + f(B')) + f(A \cap B)$. By the fact proven above, $(f(A') + f(A \cap B) + f(B')) \geqslant f(A' \cup (A \cap B) \cup B')$. However, $(A' \cup (A \cap B) \cup B') = (A \cup B)$. Finally,

$f(A) + f(B) = (f(A') + f(A \cap B) + f(B')) + f(A \cap B) \geqslant f(A \cup B) + f(A \cap B)$.

Indeed, $f(A) + f(B) \geqslant f(A \cup B) + f(A \cap B)$ implies that function f, such that f(S) is the number of edges (u, v) with $u \in S$, $v \in V \backslash S$, is submodular, by definition of submodular functions.

Q3.

Q4.

Q5.

Q6.

Q7.

Q8.