10 Algorithms (djw1005)

An *undirected network* is an undirected graph in which each edge $u \leftrightarrow v$ has an associated capacity $c(u \leftrightarrow v) > 0$. Let there be a source vertex s and a sink vertex t. We wish to find the maximum flow from s to t such that on every edge $u \leftrightarrow v$ either there is flow $u \to v$, or there is flow $v \to u$, or there is no flow at all. It is not allowed to have flow in both directions simultaneously. The flow on an edge, whichever direction it is in, must not exceed capacity. We can write the flow constraints in mathematical notation as

$$\min\bigl\{f(u \to v),\, f(v \to u)\bigr\} = 0 \quad \text{and} \quad \max\bigl\{f(u \to v),\, f(v \to u)\bigr\} \le c(u \leftrightarrow v)$$

where $f(u \to v)$ is the flow from u to v on edge $u \leftrightarrow v$. Note that this differs from the conventional Ford–Fulkerson setup, in which each *directed* edge has a capacity constraint.

(*a*) Define the *value* of a flow. State the flow conservation equation. [2 marks]

(*b*) Give an algorithm for finding a maximum flow. [6 marks]

(*c*) Prove that your algorithm returns a valid flow. [4 marks]

(*d*) Prove that your algorithm returns a maximum flow. [4 marks]

(*e*) Define the *capacity of a cut in an undirected network*. Show that the maximum flow value is equal to the minimum cut capacity. [4 marks]

[*Hint:* You may refer to code from lecture notes without repeating it in your answer. You may quote without proof any theorems from lecture notes.]