MAX FLOW / MIN CUT ALGORITHM

MA 252, 2010

Let $N = (G, V, c, s, t)$ be a network. We explain here the Ford-Fulkerson augmenting paths algorithm to compute a maximal flow in a graph.

Assumption: We assume that if $(u, v) \in E$, then $(v, u) \notin E$. This is a harmless assumption, as if $(v, u) \in E$, we can add a new vertex w, and replace (v, u) by two new edges (v, w) and (w, u), with $c(v, w) = c(w, u) = c(v, u)$.

Max Flow Algorithm.

Input: A network $N = (G, V, c, s, t)$.

Output: A flow f, and a cut (S, T) with $|f| = C(S, T)$.

(1) Initialize: Set $f(e) = 0$ for all $e \in E$. Define a vector $(p(v) : v \in V)$ (the *predecessor vector*) and a vector $(s(v) : v \in V)$ (the *slack vector*). Set `treached` = true.

(2) While `treached` = true do

  (a) Initialize: $p(v), s(v)$ are set to be unlabelled for all v. Set $s(s) = \infty$. `treached` = false. $S = L = \{s\}$. S is the set of *searched vertices*, and L is the set of *labelled vertices*.

  (b) While $S \neq \emptyset$ and `treached` = false do

    (i) Pick $v \in S$. Set $S = S \setminus \{v\}$.

    (ii) For all $(v, w) \in E$ with $w \notin L$ and $f(v, w) < c(v, w)$:

      (A) Set $L = L \cup \{w\}$.

      (B) Set $s(w) = \min(c(v, w) - f(v, w), s(v))$.

      (C) Set $p(w) = v$.

      (D) If $w \neq t$, then set $S = S \cup \{w\}$. Otherwise set `treached` =true.

    (iii) For all $(w, v) \in E$ with $w \notin L$ and $f(w, v) > 0$:

      (A) Set $L = L \cup \{w\}$.

      (B) Set $s(w) = \min(f(v, w), s(v))$.

      (C) Set $p(w) = v$.

      (D) If $w \neq t$, then set $S = S \cup \{w\}$. Otherwise set `treached` =true.

  (c) If `treached` = true, then augment the path by $s(t)$:

    (i) Set $v = t$.

(ii) While $v \neq s$ do:
  (A) if $(p(v), v) \in E$, then set $f(p(v), v) = f(p(v), v) + s(t)$.
  (B) if $(v, p(v)) \in E$, then set $f(v, p(v)) = f(v, p(v)) - s(t)$.
  (C) Set $v = p(v)$.

(3) Output f, $(L, V \setminus L)$.