COMPSCI 530: Design and Analysis of Algorithms	09/12/2013
Lecture # 6	
Lecturer: Debmalya Panigrahi	*Scribe: Abhishek Kumar Dubey*

1 Overview

Goldberg-Rao algorithm for the maximum flow problem acheives $O(min\{m^{\frac{2}{3}}, n^{\frac{1}{2}}\} \cdot m \cdot logn \cdot lognU)$ running time, where U is the largest arc capacity in graph and n and m are the number of vertex and edge respectively. It is a weakly polynomial algorithm as its run time depends on the largest arc capacity U. This is the best known algorithm for maximum flow in this category. Also, a better data structure further improves the run time bound to $O(min\{m^{\frac{2}{3}}, n^{\frac{1}{2}}\} \cdot m \cdot log\frac{n^2}{m} \cdot lognU)$.

2 Definitions

Definition 1. *Length function is defined as*

$$l(v,w) : E(v,w) \rightarrow R_+$$

It assigns some non-negative number to every arc of the graph.

Example 1. Example of a length function is below:

$$l(v,w) = 1, \ \forall \text{ (v,w)} \in \text{A}$$

Here, each edge is assigned a length of 1.

Definition 2. *A distance labeling d:V $\rightarrow$ Z^+ is a function with respect to the length function l if d(t)=0 and every arc (v,w) satisfies $d(v) \leq d(w) + l(v,w)$.*

Definition 3. *Volume of the network is defined as*

$$\sum_{e \in E} u(e) l(e)$$

u(e) is the capacity of the edge e and l(e) is length of the edge e.

Example 2. Volume of an unit capacity network with respect to l (as defined in example 1):

$$\sum_{e \in E} u(e) * 1 = m$$

Definition 4. *A graph is admissible if it satisfies one of these two constraints:*

1. *l(x,y) = 1 and d(x) = d(y) - 1*
2. *l(x,y) = 0 and d(x) = d(y)*

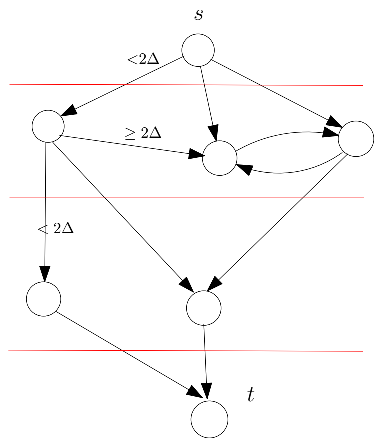

Figure 1: Layered Cut on an admissible graph

Definition 5. *A s-t cut that separates the vertices with distance labeling $\leq k$ to the rest of vertices is called layered cut.*

Example 3. Red lines on Figure 1 are example of layered cut.

Lemma 1. *In a network*

$$Residual\ flow \leq \frac{volumn}{d_l(s,t)} \tag{1}$$

$d_l(s,t)$ is the distance between s and t with respect to any length function l.

Proof. Using the flow decomposition, we can decompose flow F(e) into $F_p(e)$. Here, $F_p(e)$ is the flow carried on edge e by decomposed path p.

$$\begin{aligned}
volumn &= \sum_e l(e)u(e) \\
&\geq \sum_e l(e)F(e) \\
&= \sum_e l(e)\sum_p F_p(e) \\
&= \sum_p F_p \sum_{e\in p} l(e) \\
&= \text{F } d_l(s,t) \\
\Longrightarrow F &\leq \frac{volumn}{d_l(s,t)}
\end{aligned}$$

□

In previous lecture, we saw that residual flow of an unit capacity network is at most $\frac{m}{d(s,t)}$. By substituting the value of volumn and $d_l(s,t)$ (derived for unit capacity network in example 1,2), we get the same upper bound for the residual flow.

3 Algorithm

Goldberg-Rao Algorithm.

Start with very rough estimate F = nU
compute $\Delta = \frac{F}{\sqrt{m}}$ and G(f, Δ)
while F ≥ 1 **do**
  compute F_l
  while $F_l \geq \frac{F}{2}$ **do**
    find a blocking flow in G(f, Δ)
    if flow $\geq \Delta$ **then**
      reduce the flow to Δ
    end if
    augment flow f
    compute the F_l and G(f, Δ)
  end while
  F = $\frac{F}{2}$
end while

Length Function.

This algorithm uses following binary length function:

$$\begin{aligned} l(x,y) &= 1, \;\; u_f(v,w) \geq 3\Delta \text{ or (v, w) is special} \\ &= 0, \;\; \text{otherwise} \end{aligned}$$

An arc (v,w) is special if it satisfies following constraints:

1. $2\Delta \leq u_f(v,w) \leq 3\Delta$
2. d(v) = d(w)
3. $d_f(v,w) \geq 3\Delta$

Avoid Cycle.

To avoid the cycle in the admissible graph, all strongly connected component in the same level is contracted.

4 Run time analysis

In this section, we derive the upper bound on runtime of Goldberg-Rao algorithm. Algorithm iterates O(lognU) time on outer loop as it starts with residual flow F as nU and residual flow decreases by F/2 in each iteration until reaches 1. Later in this section, we show that number of iteration of inner loop is O($\sqrt{m}$). Also, we know the algorithm for finding the blocking flow in $O(mlog(\frac{n^2}{m}))$ time. So, overall running time of Goldberg-Rao algorithm is O($lognU$. $\sqrt{m}$. $mlog(\frac{n^2}{m})$).

Lemma 2. *After blocking flow, $d(s,t)$ increases by 1*

Lemma 3. *Minimum layered cut*

$$F_l \leq \frac{m.3\Delta}{d_l(s,t)}$$

Proof.

$$\begin{aligned}\text{Minimum layered cut} &\leq \text{Average layered cut}\\ &= \frac{\text{total flow capacity}}{\text{\# of layered cuts}}\\ &= \frac{\text{m 3 }\Delta}{d_l(s,t)}\end{aligned}$$

□

Corollary 4. *Number of blocking flow augmentation in inner loop is at most $O(\sqrt{m})$*

Proof. Every blocking flow increases the $d_l(s)$ by at least 1. So, number of steps required to decrease flow from F to F/2 is at most

$$\begin{aligned}&\frac{\text{m . }3\Delta}{F/2}\\ =\ &\frac{\text{m . }3\frac{F}{\sqrt{m}}}{F/2}\\ =\ &O(\sqrt{m})\end{aligned}$$

□

5 Summary

In this lecture, we learn the Goldberg-Rao algorithm for finding the maximum flow in $O(min\{m^{\frac{2}{3}}, n^{\frac{1}{2}}\}\ .\ m\ .\ log\frac{n^2}{m}\ .\ lognU)$. It is the best weakly polynomial algorithm till date.