IE170: Algorithms in Systems Engineering: Lecture 25

Jeff Linderoth

Department of Industrial and Systems Engineering
Lehigh University

March 30, 2007

Taking Stock

Last Time

- Flows

This Time

- (Cardinality) Matching
- Homework and Review

The Big Kahuna

Max-Flow Min-Cut Theorem

The following statements are equivalent

1. f is a maximum flow
2. f admits no augmenting path. (No (s,t) path in residual network)
3. $|f| = c(S,T)$ for some cut (S,T)

Ford-Fulkerson Algorithm

- This gave Lester Ford and Del Fulkerson an idea to find he maximum flow in a network:

Ford-Fulkerson(V, E, c, s, t)
for $i \leftarrow 1$ **to** n
do $f[u,v] \leftarrow f[v,u] \leftarrow 0$
while $\exists$ augmenting path P in G_f
do augment f by $c_f(P)$

- Assume all capacities are integers. If they are rational numbers, scale them to be integers.

Analysis

- If the maximum flow is $|f|^*$, then (since the augmenting path must raise the flow by at least 1 on each iteration), we will require $\leq |f|^*$ iterations.
- Augmenting the flow takes $O(|E|)$
- FORD-FULKERSON runs in $O(|f|^*|E|)$
- This is not polynomial in the size of the input.
- If you augment flow along the path with largest residual capacity, one can show that at most $O(|E| \lg U)$ iterations are needed.
  - $U = \max_{(u,v) \in V \times V} c(u,v)$
- The "greedy" (maximum capacity) aumenting path algorithm runs in $O(|E|^2 \lg U)$. This is polynomial in the size of the input, but not strongly polynomial (It still depends on the magnitude of the "numbers" in the instance, not on the size of the instance itself).

Can We Do Better!? – Edmonds-Karp

- Instead of augmenting on an *arbitrary* augmenting path, why don't we augment flow along the shortest augmenting path.
- Here shortest means simply number of edges taken, so all edges have weight 1.
- Therefore shortest pahs can be found just like you did in lab – with BFS
- With some heavy machinery (See book), one can show that if you only augment on shortest paths, then you have to do at most $O(|V||E|)$ augmentations of the flow
- Therefore Edmonds-Karp algorithm runs in $O(|V||E|^2)$ time.
- There are even faster algorithms, such as push-relabel, but we won't cover those.

Maximum Bipartite Matching

- A graph $G = (V, E)$ is bipartite if we can partition the vertices into $V = L \cup R$ such that all edges in E go between L and R
- A matching is a subset of edges $M \subseteq E$ such that for all $v \in V$, ≤ 1 edge of M is incident upon it.

Maximum Bipartite Matching

Given (undirected) bipartite graph $G = (L \cup R, E)$, find a matching M of G that contains the most edges

Applications

There are lots of applications of matching problems

- Airlines
  - L set of planes
  - R set of routes
  - $(u, v) \in E$ if plane u can fly route v
  - Maximize the number of routes served by planes

Solving It

- Bipartite matching is one of many problems that can be equivalently formulated (and solved) via maximum flows.
- Given $G = (L \cup R, E)$, create flow network $G' = (V', E')$
  - $V' = V \cup \{s, t\}$
  - $E' = \{(s, u) \mid u \in L\} \cup E \cup \{(v, t) \mid v \in R\}$
  - $c(u, v) = 1 \ \forall (u, v) \in E'$

Observations

(You can see the book for more formal proofs)

- There is a matching M in G of size $|M|$ if and only if there is an (integer-valued) flow f in G' of value $|f| = |M|$.
- Thus a maximum-matching in a bipartite graph G is the value of the maximum flow in the flow network G'

IE170 Problem Sets

- Here is a table of score distributions for the graded problem sets

Problem Set	Grader	Points	Max	Min	Avg	Median
1	Prof. L	50	47	30	41.3	41.5
2	Abhishek	90	71	16	51.25	52.5
3	Abhishek	50	41	18	29.4	29
4	Mustafa	30	27	10	19.8	23
5/6	Udom	75	47	22	37	39.5
Total		100	74.2	48.5	60.6	60.9

IE170 Points

- You have accumulated roughly 40% of your total score for IE170
  - 18% Problem Sets
  - 15% Quiz #1
  - 7% Participation

Score Distribution

≥ 85	1
$[80, 85)$	3
$[75, 80)$	2
$[70, 75)$	3
$[65, 70)$	0
$[60, 65)$	1
$[55, 60)$	1
< 55	1

IE171 Problem Sets

- Here is a table of score distributions for the graded labs. (All out of 100)

Lab	Grader	Max	Min	Avg	Median
1	Prof. L	100	100	100	100
2/3	Abhishek	98	12	47	42
4	Mustafa	108	27.5	82	97.5
5/6	Udom	100	10	65	75
Total		100.3	62.5	85.4	86.7

IE171 Average

- You have accumulated roughly 50% of your total score for IE171
- Two more labs.
- Coding Quiz (25%)
- Lowest Lab score tossed

Score Distribution

≥ 100	1
$[95, 100)$	2
$[90, 95)$	3
$[85, 90)$	0
$[80, 85)$	2
$[75, 80)$	2
< 75	2

Start Studying!

- Dynamic Programming (15.[1,3])
- Greedy Algorithms (16.[1,2])
- Graphs and Search (22.*)
- Spanning Trees (23.*)
- (Single Source) Shortest Paths (24.[1,2,3])
- (All Pairs) Shortest Paths (25.[1,2])
- Max Flow (26.[1,2,3])

Next Time

- Review. No Lab. But we will meet in lab for a review session for a while.
- Quiz: April 4
- Programming Quiz: April 23