http://www.cs.unh.edu/~ruml/cs758

Applications of Cuts and Flows

Max-Flow Min-Cut Theorem

value of a flow = flow across any cut

any flow value $\leq$ capacity of cut

Theorem: these are the same:

1. f is a maximum flow
2. the residual network G_f contains no augmenting paths
3. there exists a cut whose capacity is the value of f

1=2: FF is correct; 1=3: FF also finds minimum cuts

Image Segmentation

an image as a graph!

maximize

$$\sum_{i \in A} a_i + \sum_{i \in B} b_i - \sum_{e \ cutby \ A} p_{i,j}$$

minimize

$$\sum_{i \in A} b_i + \sum_{i \in B} a_i + \sum_{e \ cutby \ A} p_{i,j}$$

cut crosses three types of edges: s_i, t_i, and $p_{i,j}$

Maximum Matching

bipartite graphs: jobs/machines, classes/instructors, . . .

Maximum Matching

bipartite graphs: jobs/machines, classes/instructors, . . .

unit capacities

flow = matching

FF guarantees integer flow

running time? (hint: bound $|f^*|$)

Scheduling

does a feasible schedule exist
using only 3 machines (allowing preemption)?

job	1	2	3	4
processing time	1.5	1.25	2.1	3.6
release date	3	1	3	5
due date	5	4	7	9

does a feasible schedule exist
using only 3 machines (allowing preemption)?

job	1	2	3	4
processing time	1.5	1.25	2.1	3.6
release date	3	1	3	5
due date	5	4	7	9

arcs from s to jobs labeled with job size

intervals: 1–3–4–5–7–9

arcs from job to feasible intervals labeled with length of interval

arcs from interval to t labeled with total achievable work (num machines times length of interval)

Break

- asst 11
- permissible research

Linear Programming

real variables, linear constraints, linear objective

eg, cheapest diet that meets nutrition guidelines

- decision variables: $x_{broccoli}$, $x_{carrots}$, ...
- minimize: $cost = 4.99x_{broccoli} + 2.67x_{carrots} \ldots$
- subject to: $500 < y_{vitaminA} = 2.3x_{broccoli} + 1.7x_{carrots} \ldots$
- $300 < y_{vitaminC} = 1.7x_{broccoli} + 5.2x_{carrots} \ldots$

polynomial time (ellipsoid, Karmarkar's), but simplex method is popular
CPLEX, Gurobi, lpsolve

Example LPs

- shortest paths
- max flow obeying capacities
- min-cost flow meeting demand
- multicommodity flow meeting demands
- earliest finish time subject to job durations

Example: Max Flow

maximize $\sum_v f(s,v) - \sum_v f(v,s)$
subject to

$$\begin{aligned} f(u,v) &\geq 0 \text{ for every edge } (u,v) \\ \sum_v f(u,v) &= \sum_v f(v,u) \text{ for every vertex } u \\ f(u,v) &\leq c(u,v) \text{ for every edge } (u,v) \end{aligned}$$

Example: Shortest Path

maximize d_t
subject to

$$\begin{aligned} d_v &\leq d_u + w(u,v) \text{ for every edge } (u,v) \\ d_s &= 0 \end{aligned}$$

Min-cost Flow Meeting Demand

minimize $\sum_{(u,v)} a(u,v)f(u,v)$
subject to

$$
\begin{aligned}
f(u,v) &\geq 0 \text{ for every edge } (u,v)\\
\sum_v f(u,v) &= \sum_v f(v,u) \text{ for every vertex } u\\
f(u,v) &\leq c(u,v) \text{ for every edge } (u,v)\\
\sum_v f(s,v) - \sum_v f(v,s) &= d
\end{aligned}
$$

minimize 0

subject to

$$\sum_v f_i(u,v) - \sum_v f_i(v,u) = 0$$

$$\begin{aligned} f_i(u,v) &\geq 0 \\ \sum_{i=1}^{k} f_i(u,v) &\leq c(u,v) \\ \sum_v f_i(s_i,v) - \sum_v f_i(v,s_i) &= d_i \end{aligned}$$

Beyond Linear Programming

convex programming: constraints and objective are convex
polynomial time

quadratic programming: constraints and objective are quadratic
some forms are polynomial time

integer linear programming: integer variables, linear constraints and objective
NP-complete

0-1 ILP: 0-1 variables, linear constraints, linear objective
(BIP) NP-complete

combinatorial optimization: variables are discrete

Others

selection

multicommodity flow is NP-hard for integer flows. Use LP for fractional flows.

EOLQs

For example:

- What's still confusing?
- What question didn't you get to ask today?
- What would you like to hear more about?

Please write down your most pressing question about algorithms and put it in the box on your way out.

Thanks!