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

2 handouts: slides, asst 11
Applications of Cuts and Flows
value of a flow = flow across any cut

any flow value ≤ 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
an image as a graph!

maximize

$$\sum_{i \in A} a_i + \sum_{i \in B} b_i - \sum_{\text{cut by } A} p_{i,j}$$

minimize

$$\sum_{i \in A} b_i + \sum_{i \in B} a_i + \sum_{\text{cut by } A} p_{i,j}$$

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

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

unit capacities

flow = matching

FF guarantees integer flow

running time? (hint: bound $|f^*|$)
does a feasible schedule exist using only 3 machines (allowing preemption)?

<table>
<thead>
<tr>
<th>job</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>processing time</td>
<td>1.5</td>
<td>1.25</td>
<td>2.1</td>
<td>3.6</td>
</tr>
<tr>
<td>release date</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>due date</td>
<td>5</td>
<td>4</td>
<td>7</td>
<td>9</td>
</tr>
</tbody>
</table>
does a feasible schedule exist using only 3 machines (allowing preemption)?

<table>
<thead>
<tr>
<th>job</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>processing time</td>
<td>1.5</td>
<td>1.25</td>
<td>2.1</td>
<td>3.6</td>
</tr>
<tr>
<td>release date</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>due date</td>
<td>5</td>
<td>4</td>
<td>7</td>
<td>9</td>
</tr>
</tbody>
</table>

arcs from $s$ to jobs labeled with job size

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 length of interval)
selection

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

Applications
- Max-Flow Thm
- Segmentation
- Matching
- Scheduling
- Others
- Break
- EOLQs

- asst 10
- asst 11
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!*