Precedence Constraints: $1|\text{prec}|L_{\text{max}}$

Example

<table>
<thead>
<tr>
<th>$j$</th>
<th>$p_j$</th>
<th>$d_j$</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>B</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>21</td>
</tr>
<tr>
<td>D</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>E</td>
<td>5</td>
<td>15</td>
</tr>
<tr>
<td>F</td>
<td>4</td>
<td>19</td>
</tr>
</tbody>
</table>

Precedence: $A \rightarrow B, B \rightarrow C, B \rightarrow F, D \rightarrow E, E \rightarrow F$

Algorithmic Ideas?: Can we choose which job will run first?
Precedence Constraints: \( 1|\text{prec}|L_{\text{max}} \)

Example

\[
\begin{array}{c|c|c}
 j & p_j & d_j \\
\hline
 A & 2 & 10 \\
 B & 3 & 24 \\
 C & 1 & 21 \\
 D & 6 & 5 \\
 E & 5 & 15 \\
 F & 4 & 19 \\
\end{array}
\]

Precedence: \( A \rightarrow B, B \rightarrow C, B \rightarrow F, D \rightarrow E, E \rightarrow F \)

Algorithmic Ideas?: Can we choose which job will run first? NO, but we can choose which job will run last. Least Cost Last
More General Cost Functions

• Let each job $j$ have its own cost function $f_j(C_j)$.
• Objective $h_{\text{max}} = \max\{h_1(C_1), \ldots, h_n(C_n)\}$.
• For $L_{\text{max}}$, we just have that $h_j(C_j) = C_j - d_j$.

Example

<table>
<thead>
<tr>
<th>$j$</th>
<th>$p_j$</th>
<th>$h_j(C_j)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>$h_1(x) = x$</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>$h_2(x) = x^2$</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>$h_3(x) = 0$ if $x &lt; 5$, 10 o.w.</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>$h_4(x) = 3$</td>
</tr>
</tbody>
</table>

• LCL runs in $O(n^2)$ time.
• LCL is optimal for $1|\text{prec}|h_{\text{max}}$. Proof by exchange argument.