Automatic Strengthening of Crude-Oil Scheduling Operations Models

Sylvain Mouret, Ignacio E. Grossmann and Pierre Pestiaux

Enterprise-Wide Optimization, 10 March 2010
Motivation

- Scheduling models based on discrete-time formulations are usually efficiently solved by commercial MILP solvers (CPLEX, Xpress)
  - Often solved at root node (after presolve) by pure cutting plane method
  - Also, they have small integrality gaps
- For continuous-time formulations, many authors have presented tightening constraints based on the time representation
  - However, commercial solvers are not able to generate cutting planes for most mixed-integer scheduling constraints
- Our main goal is to automatically derive strengthened formulations from the scheduling constraints
Scheduling approach

**Time representations**

- Mathematical models for 4 different scheduling formulations
- Common set of variables
- **Operation and priority-slot** based representations

An operation is an action that can be executed one or several times.

Set of priority-slots \( T = \{1, \ldots, n\} \)

Any operation assigned to priority-slot \( i \) is given scheduling priority \( i \).
## Scheduling approach

### Time representations
- Mathematical models for 4 different scheduling formulations
- Common set of variables
- **Operation** and **priority-slot** based representations

### Set of operations $W$
An operation is an **action** that can be executed one or several times.
Scheduling approach

<table>
<thead>
<tr>
<th>Time representations</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔ Mathematical models for 4 different scheduling formulations</td>
</tr>
<tr>
<td>✔ Common set of variables</td>
</tr>
<tr>
<td>✔ <strong>Operation</strong> and <strong>priority-slot</strong> based representations</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set of operations $W$</th>
</tr>
</thead>
<tbody>
<tr>
<td>An operation is an <strong>action</strong> that can be executed one or several times.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set of priority-slots $T = {1, \ldots, n}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Any operation assigned to priority-slot $i$ is given <strong>scheduling priority</strong> $i$.</td>
</tr>
</tbody>
</table>
Scheduling approach

Time representations

- Mathematical models for 4 different scheduling formulations
- Common set of variables
- Operation and priority-slot based representations

Set of operations $W$

An operation is an action that can be executed one or several times.

Set of priority-slots $T = \{1, \ldots, n\}$

Any operation assigned to priority-slot $i$ is given scheduling priority $i$.

Type of scheduling problems

The proposed time representations can be used to model and solve scheduling problems in which operations can be sequenced as a whole.

- No sequencing of events such as start or end times
- Unsupported features: cumulative resource constraints, simultaneous inventory charging and discharging, ...
Multi-Operation Sequencing (MOS)

- 8 possible operations / 6 priority-slots
- Given 2 non-overlapping operations \( v, w \in W \)
  - \( v \) and \( w \) cannot be assigned to the same priority-slot
  - \( v \) and \( w \) are sequenced according to their scheduling priority
- Example: unloading operations 1 and 2 are assigned to slots 3 and 6

Figure: A solution schedule obtained using the MOS time representation
MOS with Synchronized Start Time (MOS-SST)

- 8 possible operations / 7 priority-slots
- Same features as the MOS representation
- Specific feature:
  - All operations assigned to priority-slot $i$ must start at the same time
  - The scheduling horizon is divided into variable adjacent time intervals

Figure: A solution schedule obtained using the MOS-SST time representation
MOS with Fixed Start Time (MOS-FST)

- 8 possible operations / 16 priority-slots
- Same features as the MOS-SST representation
- Specific feature:
  - All operations assigned to priority-slot $i$ must start at fixed time point
  - The scheduling horizon is divided into fixed adjacent time intervals

**Figure:** A solution schedule obtained using the MOS-FST time representation
Single-Operation Sequencing (SOS)

- 8 possible operations / 10 priority-slots
- Same features as the MOS representation
- Specific feature:
  - At most one operation can be assigned to each priority-slot
  - The solution can be represented as a single sequence of operations

![Diagram of SOS representation]

**Figure:** A solution schedule obtained using the SOS time representation
A few comments

For a given number of priority-slots $n$

- MOS-SST, MOS-FST and SOS models are special cases of the MOS model.
- They can be obtained from the MOS model by adding new constraints.
- The strengthened formulation for MOS models can also be applied to other time representations.
- This is not usually the case for symmetry-breaking constraints which heavily depends on the time representation used.
Mathematical Model: MOS

- Assignment variables
  \[ Z_{iv} \in \{0, 1\} \quad i \in T, v \in W \]

- Start, duration, end variables
  \[ S_{iv}, D_{iv}, E_{iv} \geq 0 \quad i \in T, v \in W \]

- Assignment constraint
  \[ Z_{iv_1} + Z_{iv_2} \leq 1 \quad i \in T, v_1, v_2 \in W, NO_{v_1v_2} = 1 \]

- Non-overlapping constraint
  \[ E_{i_1v_1} \leq S_{i_2v_2} + H \cdot (1 - Z_{i_2v_2}) \quad i_1, i_2 \in T, i_1 \leq i_2, v_1, v_2 \in W, NO_{v_1v_2} = 1 \]
Crude-oil operations scheduling problem

- Introduced in Lee et al. (1996)
- Scheduling horizon \([0, H]\)
- 3 types of operations:
  - Unloading: Vessel unloading to storage tanks
  - Transfer: Transfer from storage tanks to charging tanks
  - Distillation: Distillation of charging tanks

- 4 types of resources:
  - Crude-oil marine vessels
  - Storage tanks
  - Charging tanks
  - Crude Distillation Units (CDUs)

- Each charging tank is dedicated to a specific type of crude-oil blends (e.g. sulfur content limitations)
Crude-oil operations schedule

Logistics constraints

(i) Only one docking station available for vessel unloadings
(ii) A tank is either being filled, discharged, or idle
(iii) A tank can charge only one CDU at a time
(iv) A CDU can be charged by only one tank at a time
(v) Continuous operation of CDUs

Refinery operations

Gantt chart
Non-overlapping graph

Non-overlapping constraints

(i) Only one docking station available for vessel unloadings
(ii) A tank is either being filled, discharged, or idle
(iii) A tank can charge only one CDU at a time
(iv) A CDU can be charged by only one tank at a time

Non-overlapping graph \( G_{NO} = (V, E) \):

- \( V = \mathcal{W} \) (set of operations)
- \( E = \{\{v_1, v_2\}, NO_{v_1v_2} = 1\} \)

(set of non-overlapping relations)

Self-loops are not displayed!
Non-overlapping graph

Non-overlapping constraints

(i) Only one docking station available for vessel unloadings

(ii) A tank is either being filled, discharged, or idle

(iii) A tank can charge only one CDU at a time

(iv) A CDU can be charged by only one tank at a time

Non-overlapping graph $G_{NO} = (V, E)$:

- $V = W$ (set of operations)
- $E = \{\{v_1, v_2\}, NO_{v_1v_2} = 1\}$ (set of non-overlapping relations)

Self-loops are not displayed!
Non-overlapping graph

**Non-overlapping constraints**

(i) Only one docking station available for vessel unloadings

(ii) A tank is either being filled, discharged, or idle

(iii) A tank can charge only one CDU at a time

(iv) A CDU can be charged by only one tank at a time

Non-overlapping graph $G_{NO} = (V, E)$:

- $V = W$ (set of operations)
- $E = \{\{v_1, v_2\}, NO_{v_1v_2} = 1\}$ (set of non-overlapping relations)

Self-loops are not displayed!

Crude Vessels    | Storage Tanks | Charging Tanks | CDUs
---|---|---|---
1 | 4 | 11 | 5
2 | 7 | 13 | 9
3 | 10 | 14 | 12

Diagram of the non-overlapping graph with nodes representing operations and edges representing non-overlapping constraints.
Non-overlapping graph

Non-overlapping constraints

(i) Only one docking station available for vessel unloadings
(ii) A tank is either being filled, discharged, or idle
(iii) A tank can charge only one CDU at a time
(iv) A CDU can be charged by only one tank at a time

Non-overlapping graph $G_{NO} = (V, E)$:

- $V = W$ (set of operations)
- $E = \{\{v_1, v_2\}, NO_{v_1v_2} = 1\}$ (set of non-overlapping relations)

Self-loops are not displayed!
Non-overlapping graph

Non-overlapping constraints

(i) Only one docking station available for vessel unloadings
(ii) A tank is either being filled, discharged, or idle
(iii) A tank can charge only one CDU at a time
(iv) A CDU can be charged by only one tank at a time

Non-overlapping graph \( G_{NO} = (V, E) \): 

- \( V = W \) (set of operations)
- \( E = \{\{v_1, v_2\}, NO_{v_1v_2} = 1\} \) (set of non-overlapping relations)

Self-loops are not displayed!

Crude Vessels  Storage Tanks  Charging Tanks  CDUs

1  4  7  11
2  5  8  12
3  6  9  13

11 / 19
Graph cliques

- A **clique** of $G_{NO}$ is a subset of the set of operations $W' \subset W$ such that any two operations in $W'$ must not overlap.
- A **maximal clique** is a clique that is not a subset of any other cliques.

$\Rightarrow$ 4 maximal cliques of 3 vertices:

- $\{1, 2, 3\}$ due to (i)
- $\{5, 12, 13\}$ due to (ii) and (iii)
- $\{7, 12, 13\}$ due to (ii) and (iii)
- $\{9, 12, 13\}$ due to (ii) and (iii)
Clques

Graph cliques

- A clique of $G_{NO}$ is a subset of the set of operations $W' \subseteq W$ such that any two operations in $W'$ must not overlap.
- A maximal clique is a clique that is not a subset of any other cliques.

$\Rightarrow$ 4 maximal cliques of 3 vertices:

- $\{1, 2, 3\}$ due to (i)
- $\{5, 12, 13\}$ due to (ii) and (iii)
- $\{7, 12, 13\}$ due to (ii) and (iii)
- $\{9, 12, 13\}$ due to (ii) and (iii)
Clique

A clique of $G_{NO}$ is a subset of the set of operations $W' \subset W$ such that any two operations in $W'$ must not overlap.

A maximal clique is a clique that is not a subset of any other cliques.

$\Rightarrow$ 4 maximal cliques of 3 vertices

- $\{1, 2, 3\}$ due to (i)
- $\{5, 12, 13\}$ due to (ii) and (iii)
- $\{7, 12, 13\}$ due to (ii) and (iii)
- $\{9, 12, 13\}$ due to (ii) and (iii)
Cliqued

Graph cliques

- A **clique** of $G_{NO}$ is a subset of the set of operations $W' \subset W$ such that any two operations in $W'$ must not overlap.
- A **maximal clique** is a clique that is not a subset of any other cliques.

$\Rightarrow$ 4 maximal cliques of 3 vertices

- $\{1, 2, 3\}$ due to (i)
- $\{5, 12, 13\}$ due to (ii) and (iii)
- $\{7, 12, 13\}$ due to (ii) and (iii)
- $\{9, 12, 13\}$ due to (ii) and (iii)
Strengthened reformulations

Strengthened non-overlapping constraints

- **Assignment constraint using maximal cliques**
  \[
  \sum_{v \in W'} Z_{iv} \leq 1 \quad i \in T, \ W' \in \text{maxclique}(G_{NO})
  \]

- **Non-overlapping constraint using maximal cliques**
  \[
  \sum_{v \in W'} E_{i_1v} \leq \sum_{v \in W'} S_{i_2v} + H \cdot \left[1 - \sum_{v \in W'} Z_{i_2v}\right] \quad i_1, i_2 \in T, \ i_1 < i_2, \ W' \in \text{maxclique}(G_{NO})
  \]
Strengthened reformulations

Strengthened non-overlapping constraints

- Assignment constraint using maximal cliques
  \[ \sum_{v \in W'} Z_{iv} \leq 1 \quad i \in T, W' \in \text{maxclique}(G_{NO}) \]

- Non-overlapping constraint using maximal cliques
  \[ \sum_{v \in W'} E_{i_1v} \leq \sum_{v \in W'} S_{i_2v} + H \cdot \left[ 1 - \sum_{v \in W'} Z_{i_2v} \right] \quad i_1, i_2 \in T, i_1 < i_2, W' \in \text{maxclique}(G_{NO}) \]

⇒ More compact and tighter formulation!

\[ \begin{align*}
E_{i_11} &\leq S_{i_21} + H \cdot (1 - Z_{i_21}) \\
E_{i_11} &\leq S_{i_22} + H \cdot (1 - Z_{i_22}) \\
E_{i_11} &\leq S_{i_23} + H \cdot (1 - Z_{i_23}) \\
E_{i_12} &\leq S_{i_21} + H \cdot (1 - Z_{i_21}) \\
E_{i_12} &\leq S_{i_22} + H \cdot (1 - Z_{i_22}) \\
E_{i_12} &\leq S_{i_23} + H \cdot (1 - Z_{i_23}) \\
E_{i_13} &\leq S_{i_21} + H \cdot (1 - Z_{i_21}) \\
E_{i_13} &\leq S_{i_22} + H \cdot (1 - Z_{i_22}) \\
E_{i_13} &\leq S_{i_23} + H \cdot (1 - Z_{i_23}) \\
\end{align*} \]

\[ \Rightarrow \sum_{v=1}^{3} E_{i_1v} \leq \sum_{v=1}^{3} S_{i_2v} + H \cdot (1 - \sum_{v=1}^{3} Z_{i_2v}) \]

For maximal clique \{1, 2, 3\}
Computational results

- Problem 2 from Lee et al. (1996) solved using the MOS representation
- Only the MILP relaxation is solved (nonlinear constraints are dropped)
- MILP solver: GAMS/CPLEX 12 (cut generation deactivated)
- LP relaxation is identical in all cases

<table>
<thead>
<tr>
<th></th>
<th>Original Formulation</th>
<th>Strengthened Assignment csts</th>
<th>Strengthened Non-overlap. csts</th>
<th>Full Strengthening</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU time</td>
<td>95s</td>
<td>78s</td>
<td>88s</td>
<td>57s</td>
</tr>
<tr>
<td>Nb of nodes</td>
<td>2704</td>
<td>1652</td>
<td>2540</td>
<td>1708</td>
</tr>
</tbody>
</table>

- MILP solver: GAMS/CPLEX 12 (cut generation activated)

<table>
<thead>
<tr>
<th></th>
<th>Original Formulation</th>
<th>Strengthened Assignment csts</th>
<th>Strengthened Non-overlap. csts</th>
<th>Full Strengthening</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU time</td>
<td>156s</td>
<td>143s</td>
<td>113s</td>
<td>34s</td>
</tr>
<tr>
<td>Nb of nodes</td>
<td>2624</td>
<td>1945</td>
<td>2209</td>
<td>570</td>
</tr>
</tbody>
</table>

⇒ Strengthened formulations using maximal cliques are solved faster!
Bicliques

Graph bicliques

▸ A biclique of $G_{NO}$ is a pair of sets of operations $(W_1; W_2) \in W^2$ such that for any pair of operations $(v_1; v_2) \in W_1 \times W_2$, $v_1$ and $v_2$ must not overlap

▸ A maximal biclique is a biclique that is not contained in any other bicliques

⇒ 4 maximal bicliques of 3 vertices

- $(\{4\}; \{1, 4, 11\})$ due to (ii)
- $(\{6\}; \{2, 6, 11\})$ due to (ii)
- $(\{8\}; \{2, 8, 14\})$ due to (ii)
- $(\{10\}; \{3, 10, 14\})$ due to (ii)
Bicliques

Graph bicliques

- A biclique of $G_{NO}$ is a pair of sets of operations $(W_1; W_2) \in W^2$ such that for any pair of operations $(v_1; v_2) \in W_1 \times W_2$, $v_1$ and $v_2$ must not overlap
- A maximal biclique is a biclique that is not contained in any other bicliques

$\Rightarrow$ 5 maximal bicliques of 4 vertices

- $(\{9\}; \{3, 9, 12, 13\})$ due to (ii)
- + 2 similar based on 5 and 7
- $(\{11\}; \{4, 6, 11, 12\})$ due to (ii) and (iv)
- + 1 similar based on 14
Bicliques

Graph bicliques

- A biclique of $G_{NO}$ is a pair of sets of operations $(W_1; W_2) \in W^2$ such that for any pair of operations $(v_1; v_2) \in W_1 \times W_2$, $v_1$ and $v_2$ must not overlap.
- A maximal biclique is a biclique that is not contained in any other bicliques.

$\Rightarrow$ 3 maximal bicliques of 5 vertices

- $\{\{1\}; \{1, 2, 3, 4, 5\}\}$ due to (i) and (ii)
- $\{\{3\}; \{1, 2, 3, 9, 10\}\}$ due to (i) and (ii)
- $\{\{5, 7, 9, 12, 13\}; \{12, 13\}\}$ due to (ii)
Bicliques

Graph bicliques

- A biclique of $G_{NO}$ is a pair of sets of operations $(W_1; W_2) \in W^2$ such that for any pair of operations $(v_1; v_2) \in W_1 \times W_2$, $v_1$ and $v_2$ must not overlap.
- A maximal biclique is a biclique that is not contained in any other bicliques.

⇒ 3 maximal bicliques of 6 vertices

- $(\{2\}; \{1, 2, 3, 6, 7, 8\})$ due to (i) and (ii)
- $(\{12\}; \{5, 7, 9, 11, 12, 13\})$
  - due to (ii),(iii) and (iv)
- + 1 similar based on 13
Strengthened reformulations

Strengthened non-overlapping constraints

- Non-overlapping constraint using maximal bicliques

\[
\sum_{v_1 \in W_1} E_{i_1 v_1} \leq \sum_{v_2 \in W_2} S_{i_2 v_2} + H \cdot \left[ 1 - \sum_{v_2 \in W_2} Z_{i_2 v_2} \right]
\]

\[
\sum_{v_2 \in W_2} E_{i_1 v_2} \leq \sum_{v_1 \in W_1} S_{i_2 v_1} + H \cdot \left[ 1 - \sum_{v_1 \in W_1} Z_{i_2 v_1} \right]
\]

\(i_1, i_2 \in T, i_1 < i_2, (W_1; W_2) \in \text{maxbiclique}(G_{NO})\)
Strengthened reformulations

Strengthened non-overlapping constraints

▶ Non-overlapping constraint using maximal bicliques

\[
\sum_{v_1 \in W_1} E_{i_1 v_1} \leq \sum_{v_2 \in W_2} S_{i_2 v_2} + H \cdot \left[ 1 - \sum_{v_2 \in W_2} Z_{i_2 v_2} \right]
\]

\[
\sum_{v_2 \in W_2} E_{i_1 v_2} \leq \sum_{v_1 \in W_1} S_{i_2 v_1} + H \cdot \left[ 1 - \sum_{v_1 \in W_1} Z_{i_2 v_1} \right]
\]

\[i_1, i_2 \in T, i_1 < i_2, (W_1; W_2) \in \text{maxbiclique}(G_{NO})\]

⇒ More compact and tighter formulation!

\[
\begin{align*}
E_{i_1} &\leq S_{i_2 4} + H \cdot (1 - Z_{i_2 4}) \\
E_{i_4} &\leq S_{i_2 1} + H \cdot (1 - Z_{i_2 1}) \\
E_{i_4} &\leq S_{i_2 4} + H \cdot (1 - Z_{i_2 4}) \\
E_{i_4} &\leq S_{i_2 11} + H \cdot (1 - Z_{i_2 11}) \\
E_{i_11} &\leq S_{i_2 4} + H \cdot (1 - Z_{i_2 4})
\end{align*}
\]

⇒ \[
\begin{align*}
E_{i_4} &\leq \sum_{v=1,4,11} S_{i_2 v} + H \cdot (1 - \sum_{v=1,4,11} Z_{i_2 v}) \\
\sum_{v=1,4,11} E_{i_2 v} &\leq S_{i_2 4} + H \cdot (1 - Z_{i_2 4})
\end{align*}
\]

For maximal biclique \((\{4\}; \{1, 4, 11\})\)
Cliques and bicliques selection strategies

- Cliques can be used alone in the SOS time representation
- Bicliques can also help strengthening the model
- However, many redundant constraints would be generated by simultaneously using cliques and bicliques
- We propose three cliques/bicliques selection strategies:
  - Selection a: select all maximal cliques
  - Strategie b: select cliques and bicliques from constraint definitions
  - Selection c: improve selection b by making cliques and bicliques maximal and removing unnecessary elements

<table>
<thead>
<tr>
<th>Constraint</th>
<th>No.</th>
<th>Selection a</th>
<th>Selection b</th>
<th>Selection c</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) 1</td>
<td>1</td>
<td>{1, 2, 3}</td>
<td>{1, 2, 3}</td>
<td>implied by 2, 3, 4</td>
</tr>
<tr>
<td>2</td>
<td>{1, 4}, {1, 5}</td>
<td>{1}; {4, 5}</td>
<td>{1}; {1, 2, 3, 4, 5}</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>{2, 6}, {2, 7}, {2, 8}</td>
<td>{2}; {6, 7, 8}</td>
<td>{2}; {1, 2, 3, 6, 7, 8}</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>{3, 9}, {3, 10}</td>
<td>{3}; {9, 10}</td>
<td>{3}; {1, 2, 3, 9, 10}</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>{4, 11}, {6, 11}</td>
<td>{4, 6}; {11}</td>
<td>{4, 6, 11, 12}; {11}</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>{5, 12, 13}, {7, 12, 13}, {9, 12, 13}</td>
<td>{5, 7, 9}; {12, 13}</td>
<td>{5, 7, 9, 12, 13}; {12, 13}</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>{8, 14}, {10, 14}</td>
<td>{8, 10}; {14}</td>
<td>{8, 10, 13, 14}; {14}</td>
<td></td>
</tr>
<tr>
<td>(iii) 8</td>
<td>implied by 6</td>
<td>{12, 13}</td>
<td>implied by 6</td>
<td></td>
</tr>
<tr>
<td>(iv) 9</td>
<td>{11, 12}</td>
<td>{11, 12}</td>
<td>implied by 5</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>{13, 14}</td>
<td>{13, 14}</td>
<td>implied by 7</td>
<td></td>
</tr>
</tbody>
</table>

No. of non-overlap. csts

15  16  12
Computational results

- Problems 1 to 4 are solved using the SOS representation
- Only the MILP relaxation is solved (nonlinear constraints are dropped)
- MILP solver: GAMS/CPLEX 12 (cut generation deactivated)
  - Selection \textit{a}: select all maximal cliques
  - Selection \textit{c}: smart selection of maximal cliques and bicliques

<table>
<thead>
<tr>
<th>Pb</th>
<th>Selection</th>
<th>( n )</th>
<th>LP</th>
<th>MILP</th>
<th>Nb of nodes</th>
<th>CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>\textit{a}</td>
<td>13</td>
<td>80.000</td>
<td>79.750</td>
<td>18</td>
<td>5.88s</td>
</tr>
<tr>
<td></td>
<td>\textit{c}</td>
<td>13</td>
<td>80.000</td>
<td>79.750</td>
<td>21</td>
<td>4.92s</td>
</tr>
<tr>
<td>2</td>
<td>\textit{a}</td>
<td>21</td>
<td>103.000</td>
<td>101.175</td>
<td>36</td>
<td>120.42s</td>
</tr>
<tr>
<td></td>
<td>\textit{c}</td>
<td>21</td>
<td>103.000</td>
<td>101.175</td>
<td>25</td>
<td>60.50s</td>
</tr>
<tr>
<td>3</td>
<td>\textit{a}</td>
<td>21</td>
<td>100.000</td>
<td>87.400</td>
<td>28</td>
<td>191.47s</td>
</tr>
<tr>
<td></td>
<td>\textit{c}</td>
<td>21</td>
<td>100.000</td>
<td>87.400</td>
<td>31</td>
<td>64.46s</td>
</tr>
<tr>
<td>4</td>
<td>\textit{a}</td>
<td>26</td>
<td>132.585</td>
<td>132.548</td>
<td>16</td>
<td>606.86s</td>
</tr>
<tr>
<td></td>
<td>\textit{c}</td>
<td>26</td>
<td>132.585</td>
<td>132.548</td>
<td>32</td>
<td>308.43s</td>
</tr>
</tbody>
</table>

\( \Rightarrow \) Maximal bicliques are more helpful than maximal cliques in SOS models
Conclusions

- Automatic strengthening of several scheduling formulations based on a global graph representation of the non-overlapping constraints
- The extraction of useful graph entities such as cliques and bicliques can be done automatically by well-known algorithmic tools:
  - very efficient for small and medium scale graphs
  - not polynomial
- The strengthened formulation is more compact and has a tighter LP relaxation:
  - 40% decrease in CPU time on problem 2 for MOS model (w/o cuts)
  - 78% decrease in CPU time on problem 2 for MOS model (w/ cuts)
  - 40% average decrease in CPU time for SOS models