A Hybrid Method for Planning and Scheduling

John Hooker
Carnegie Mellon University

CP 2004
Toronto, September
The Problem

• Allocate tasks to facilities.
• Schedule tasks assigned to each facility.
  • Subject to deadlines.
  • Facilities may run at different speeds and incur different costs.
• Cumulative scheduling
  • Several tasks may run simultaneously on a facility.
  • But total resource consumption must never exceed limit.
Approach

• In practice, problem is often solved by give-and-take.
  • If schedule doesn’t work, schedulers telephone planners and ask for a different allocation
  • Repeat until everyone can live with the solution.
• **Benders decomposition** is a mathematical formalization of this process.
  • Planning is the **master problem**.
  • Scheduling is the **subproblem**.
  • Telephone calls are **Benders cuts**.
• Use **logic-based** Benders.
  • Classical Benders requires that the subproblem be a linear or nonlinear programming problem.
Approach

• Decomposition permits hybrid solution:
  • Apply MILP to planning master problem.
    • MILP is generally better at resource allocation.
  • Apply CP to scheduling subproblem.
    • CP is generally better at scheduling.
Previous Work

1995 (JH & Yan) – Apply logic-based Benders to circuit verification.
  • Better than BDDs when circuit contains error.

  • Specialized Benders cuts must be designed for each problem class.
  • Branch-and-check proposed.

2001 (Jain & Grossmann) – Apply logic-based Benders to multiple-machine scheduling using CP/MILP.
  • Substantial speedup.
  • But… easy problem for Benders approach
2001 (Thorsteinsson) – Apply branch-and-check to CP/MILP.
   • 1-2 orders of magnitude speedup on multiple machine scheduling.

2003 (JH, Ottosson) – Apply logic-based Benders to IP (and SAT).

Today – Apply logic-based Benders to resource-constrained planning/scheduling problems.
   • Multiple facilities, cumulative scheduling on each facility.
   • Minimize cost, makespan, or total tardiness.

Also at this meeting (Cambazard et al.) – Logic-based Benders applied to real-time task allocation & scheduling
Logic-Based Benders Decomposition

\[
\begin{align*}
\text{min} & \quad f(x, y) \\
\text{subject to} & \quad C(x, y) \\
& \quad x \in D_x, y \in D_y
\end{align*}
\]

Basic idea: Search over values of \( x \) in master problem.

For each \( x = \bar{x} \) examined, solve subproblem for \( y \).

Master Problem

\[
\begin{align*}
\text{min} & \quad z \\
\text{subject to} & \quad z \geq B_{\bar{x}^k} (x), \text{ all } k \\
& \quad x \in D_x, y \in D_y
\end{align*}
\]

Subproblem

\[
\begin{align*}
\text{min} & \quad f(\bar{x}, y) \\
\text{subject to} & \quad C(\bar{x}, y) \\
& \quad y \in D_y
\end{align*}
\]

Solution of master problem

Benders cuts for all iterations \( k \)
Logic-Based Benders Decomposition

**Subproblem**

\[
\begin{align*}
\text{min} & \quad f(\bar{x}, y) \\
\text{subject to} & \quad C(\bar{x}, y) \\
& \quad y \in D_y
\end{align*}
\]

**Subproblem dual**

\[
\begin{align*}
\text{max} & \quad v \\
\text{s.t.} & \quad C(\bar{x}, y) \Rightarrow f(\bar{x}, y) \geq v \\
& \quad v \in R, \ P \in Q
\end{align*}
\]

Solution of subproblem **dual** is a proof that cost can be no less than the optimal cost \( B_{\bar{x}}(\bar{x}) \) when \( x = \bar{x} \)

We use the *same proof schema* to derive a valid lower bound \( B_{\bar{x}}(x) \) for any \( x \).

**Benders cut** \( z \geq B_{\bar{x}}(x) \) (a type of nogood) forces master problem to look at a value of \( x \) other than \( \bar{x} \) to get a lower cost.
Applying Benders to Planning & Scheduling

• **Decompose** problem into

  assignment + resource-constrained scheduling

  assign tasks to facilities schedule tasks on each facility

• Use logic-based Benders to link these.

• Solve: master problem with **MILP** -- good at resource allocation

  subproblem with **Constraint Programming** -- good at scheduling

• We will use Benders cuts that require no information from the CP solution process.
Notation

\( p_{ij} \) = processing time of task \( j \) on facility \( i \)
\( c_{ij} \) = resource consumption of task \( j \) on facility \( i \)
\( C_i \) = resources available on facility \( i \)

Total resource consumption \( \leq C_i \) at all times.
Objective functions

Minimize cost = \( \sum_{ij} g_{y_{j,j}} \)

- Fixed cost of assigning task \( j \) to facility \( y_j \)

Minimize makespan = \( \max_{ij} \{ t_{j,j} + p_{y_{j,j}} \} \)

- Start time of task \( j \)

Minimize tardiness = \( \sum_{ij} \left( t_{j,j} + p_{y_{j,j}} - d_{j,j} \right)^+ \)

- Due date for task \( j \)

\( \alpha^+ = \max \{0, \alpha\} \)
Minimize cost: MILP Model

\[
\begin{align*}
\text{min} & \quad \sum_{ijt} g_{ij} x_{ijt} \\
\text{subject to} & \quad \sum_{it} x_{ijt} = 1, \quad \text{all } j \\
& \quad \sum_{j} \sum_{t'} c_{ij} x_{ijt'} \leq C_i, \quad \text{all } i, t \\
& \quad t - p_{ij} < t' \leq t \\
& \quad x_{ijt} = 0, \quad \text{all } j, t \text{ with } d_j - p_{ij} < t \\
& \quad x_{ijt} = 0, \quad \text{all } j, t \text{ with } t > N - p_{ij} + 1 \\
& \quad x_{ijt} \in \{0,1\}
\end{align*}
\]

= 1 if task \( j \) starts at time point \( t \) on facility \( i \) \((t = 1,\ldots,N)\)

Task \( j \) starts at one time on one facility

Tasks underway at time \( t \) consume \( \leq C_i \) in resources

Tasks observe time windows
Minimize Cost: CP Model

\[
\begin{align*}
\text{min} & \quad \sum_{j} g_{y_j} \\
\text{subject to} & \quad \text{cumulative} \left( \begin{array}{c}
(t_j \mid y_j = i) \\
(p_{ij} \mid y_j = i) \\
(c_{ij} \mid y_j = i) \\
C_i
\end{array} \right), \quad \text{all } i \\
0 & \leq t_j \leq d_j - p_{y_j}, \quad \text{all } j
\end{align*}
\]

- \(y_j\) = facility assigned to task \(j\)
- start times of tasks assigned to facility \(i\)
- Observe time windows
- Observe resource limit on each facility
Minimize Cost: Logic-Based Benders

Master Problem: Assign tasks to facilities

\[
\begin{align*}
\text{min} & \quad \sum_{ij} g_{ij} x_{ij} \\
\text{subject to} & \quad \sum_i x_{ij} = 1, \quad \text{all } j \\
& \quad \sum_j p_{ij} c_{ij} x_{ij} \leq C_i d_k, \quad \text{all } i, \text{ all distinct } d_k \\
& \quad d_j \leq d_k \\
\end{align*}
\]

Benders cuts

Relaxation of subproblem:
“Area” \( d_{ij} r_{ij} \) of tasks due before \( d_k \) must fit before \( d_k \).
**Subproblem: Schedule tasks assigned to each facility**
Solve by constraint programming

\[
\begin{align*}
\text{cumulative} & \left\{ \begin{array}{l}
(t_j \mid \bar{x}_{ij} = 1) \\
(p_{ij} \mid \bar{x}_{ij} = 1) \\
(c_{ij} \mid \bar{x}_{ij} = 1) \\
C_i \\
0 \leq t_j \leq d_j
\end{array} \right\}, \quad \text{all } i
\end{align*}
\]

Let \( J_{ih} \) = set of tasks assigned to facility \( i \) in iteration \( h \).
If subproblem \( i \) is infeasible, solution of subproblem dual is a proof that not all tasks in \( J_{ih} \) can be assigned to facility \( i \).
This provides the basis for a simple Benders cut.
Master Problem with Benders Cuts
Solve by MILP

\[
\begin{align*}
\text{min} & \quad \sum_{ij} c_{ij} x_{ij} \\
\text{subject to} & \quad \sum_{i} x_{ij} = 1, \quad \text{all } j \\
& \quad \sum_{j} p_{ij} r_{ij} x_{ij} \leq C_i d_k, \quad \text{all } i, \text{all distinct } d_k \\
& \quad d_j \leq d_k \\
& \quad \sum_{j \in J_{ih}} (1 - x_{ij}) \geq 1, \quad \text{all } i, h \\
x_{ij} & \in \{0, 1\}
\end{align*}
\]
Minimize Makespan: Logic-Based Benders

**Master Problem: Assign tasks to facilities**

\[
\begin{align*}
\text{min} & \quad \sum_{ij} x_{ij} = 1, \quad \text{all } j \\
\text{subject to} & \quad m \geq \frac{1}{C_i} \sum_{j} p_{ij} c_{ij} x_{ij}, \quad \text{all } i
\end{align*}
\]

Benders cuts

Relaxation of subproblem: “Area” of tasks provides lower bound on makespan.
**Subproblem:** Schedule tasks assigned to each facility
Solve by constraint programming

\[
\begin{align*}
\min & \quad M \\
\text{subject to} & \quad \begin{cases}
M \geq t_j + d_{ij}, & \text{all } j \\
\text{cumulative} \begin{cases}
(t_j | x_{ij} = 1)
\end{cases}
\end{cases}, & \text{all } i \\
0 \leq t_j \leq d_j, & \text{all } j
\end{align*}
\]

Let \( J_{ih} \) = set of tasks assigned to machine \( i \) in iteration \( h \).

We get a Benders cut even when subproblem is feasible.
The Benders cut is based on:

**Lemma.** If we remove tasks 1, … s from a facility, the minimum makespan on that facility is reduced by at most

$$\sum_{j=1}^{s} p_{ij} + \max_{j \leq s} \{d_j\} - \min_{j \leq s} \{d_j\}$$

Assuming all deadlines $d_i$ are the same, we get the Benders cut

$$M \geq M^*_{hi} - \sum_{j \in J_{hi}} (1 - x_{ij}) p_{ij}$$

Min makespan on facility $i$ in last iteration
Why does this work? Add tasks 1,…,s sequentially at end of optimal schedule for other tasks…

**Case I:** resulting schedule meets deadline

\[ M^* \leq \hat{M} + \sum_{j=1}^{s} p_{ij} \Rightarrow \hat{M} \geq M^* - \sum_{j=1}^{s} p_{ij} \]

\[ \text{Deadline for all tasks} \]

\[ \text{Feasible makespan for all tasks} \]
Case II: resulting schedule exceeds deadline

\[ M^* \leq d \text{ and } \hat{M} + \sum_{j=1}^{s} p_{ij} > d \Rightarrow \hat{M} \geq M^* - \sum_{j=1}^{s} p_{ij} \]
**Master Problem: Assign tasks to facilities**
Solve by MILP

\[
\begin{align*}
\text{min} & \quad M \\
\text{subject to} & \quad \sum_i x_{ij} = 1, \quad \text{all } j \\
& \quad M \geq \frac{1}{C_i} \sum_j p_{ij} c_{ij} x_{ij}, \quad \text{all } i \\
& \quad M \geq M^*_{hi} - \sum_{j \in J_{ik}} (1 - x_{ij}) p_{ij}, \quad \text{all } i, h \\
x_{ij} & \in \{0,1\}
\end{align*}
\]

- Relaxation
- Benders cuts
- Makespan on facility \( i \) in iteration \( h \)
Minimize Tardiness: Logic-Based Benders

**Master Problem: Assign tasks to facilities**

\[
\begin{align*}
\text{min} & \quad T \\
\text{s.t.} & \quad \sum_{i} x_{ij} = 1, \quad \text{all } j \\
\end{align*}
\]

relaxation of subproblem

Benders cuts
Relaxation of subproblem

**Lemma.** Consider a min tardiness problem that schedules tasks 1, ..., \( n \) on facility \( i \), where \( d_1 \leq \ldots \leq d_n \). The min tardiness \( T^* \) is bounded below by

\[
L = \sum_{k=1}^{n} L_k
\]

where

\[
L_k = \left( \frac{1}{C_i} \sum_{j=1}^{k} p_i\pi_i(j)c_i\pi_i(j) - d_k \right)^+
\]

and \( \pi \) is a permutation of 1, ..., \( n \) such that

\[
p_{\pi_i(1)c_\pi_i(1)} \leq \cdots \leq p_{\pi_i(n)c_\pi_i(n)}
\]
Idea of proof

For a permutation $\sigma$ of $1, \ldots, n$ let $L(\sigma) = \sum_{k=1}^{n} L_k(\sigma)$

where $L_k(\sigma) = \left( \frac{1}{C_i} \sum_{j=1}^{k} p_{i\pi_i(j)} c_{i\pi_i(j)} - d_{\sigma(k)} \right)^+$

Let $\sigma_0(1), \ldots, \sigma_0(n)$ be order of jobs in any optimal solution, so that $t_{\sigma_0(1)} \leq \cdots \leq t_{\sigma_0(n)}$ and min tardiness is $T^*$

Consider bubble sort on $\sigma_0(1), \ldots, \sigma_0(n)$ to obtain $1, \ldots, n$. Let $\sigma_0, \ldots, \sigma_s$ be resulting sequence of permutations, so that $\sigma_s, \sigma_{s+1}$ differ by a swap and $\sigma_s(j) = j$. 
Now we have

\[ T^* \geq L(\sigma_0) \geq \cdots \geq L(\sigma_s) \geq L(\sigma_{s+1}) \geq \cdots \geq L(\sigma_S) = L \]

Since

\[ t_k \geq \frac{1}{C_i} \sum_{j=1}^{k} p_i \pi_i(j) c_i \pi_i(j) \]

\[ L(\sigma_s) = \sum_{j=1}^{k-1} L_j(\sigma_s) + L_k(\sigma_s) + L_{k+1}(\sigma_s) + \sum_{j=k+2}^{n} L_j(\sigma_s) \]

\[ L(\sigma_{s+1}) = \sum_{j=1}^{k-1} L_j(\sigma_s) + L_k(\sigma_{s+1}) + L_{k+1}(\sigma_{s+1}) + \sum_{j=k+2}^{n} L_j(\sigma_s) \]

So

\[ L(\sigma_s) - L(\sigma_{s+1}) = L_k(\sigma_s) + L_{k+1}(\sigma_s) - L_k(\sigma_{s+1}) - L_{k+1}(\sigma_{s+1}) \]

\[ = (a - A)^+ + (A - b)^+ - (a - b)^+ - (A - B)^+ \geq 0 \]

Since \( A \geq a, \ B \geq b \)
From the lemma, we can write the relaxation

\[ T \geq \sum_{i} \sum_{k=1}^{n} L'_{ik} x_{ik} \]

where \[ L'_{ik} \geq \frac{1}{C_i} \sum_{j=1}^{k} p_i \pi_i(j) c_i \pi_i(j) x_i \pi_i(j) - d_k \]

To linearize this, we write \[ T \geq \sum_{i} \sum_{k=1}^{n} L_{ik} \]

and \[ L_{ik} \geq \frac{1}{C_i} \sum_{j=1}^{k} p_i \pi_i(j) c_i \pi_i(j) x_i \pi_i(j) - d_k - (1 - x_{ik}) M_{ik} \]

where \[ M_{ik} = \frac{1}{C_i} \sum_{j=1}^{k} p_i \pi_i(j) c_i \pi_i(j) - d_k \]
Benders cuts

To extract some “dual” information, re-solve the scheduling subproblem a few times with some tasks removed.

Let \( J^0_{hi} = \{ \text{tasks that can be individually removed without reducing min makespan} \} \)

\( \Delta_{hi} = \text{reduction in min makespan if all tasks in } J^0_{hi} \text{ are removed simultaneously} \)

This yields Benders cuts:

\[
T \geq T^*_{hi} - \Delta_{hi} - U \sum_{j \in J_{hi} \setminus J^0_{hi}} (1 - x_{ij}), \quad \text{all } i, h
\]

\[
T \geq T^*_{hi} - U \sum_{j \in J_{hi} \setminus J^0_{hi}} (1 - x_{ij}), \quad \text{all } i, h
\]
Computational Results

• Random problems on 2, 3, 4 facilities.

• Facilities run at different speeds.

• All release times = 0.
  
  • Min cost and makespan problems: all tasks have same deadline.
  
  • Tardiness problems: random due date parameters set so that a few tasks tend to be late.

• No precedence or other side constraints.
  
  • Makes problem harder.

• Implement with OPL Studio
  
  • CPLEX for MILP.
  
  • ILOG Scheduler for CP. Use AssignAlternatives & SetTimes.
Min cost, 2 facilities

Computation time in seconds
Average of 5 instances shown

<table>
<thead>
<tr>
<th>Jobs</th>
<th>MILP*</th>
<th>CP</th>
<th>Benders</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1.9</td>
<td>0.14</td>
<td>0.09</td>
</tr>
<tr>
<td>12</td>
<td>199</td>
<td>2.2</td>
<td>0.06</td>
</tr>
<tr>
<td>14</td>
<td>1441</td>
<td>79</td>
<td>0.04</td>
</tr>
<tr>
<td>16</td>
<td>3605+</td>
<td>1511</td>
<td>1.1</td>
</tr>
<tr>
<td>18</td>
<td>7200+</td>
<td></td>
<td>7.0</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>85</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td>1674+</td>
<td></td>
</tr>
</tbody>
</table>

+ At least one problem in the 5 exceeded 7200 sec (2 hours)
Min cost, 3 facilities

Computation time in seconds
Average of 5 instances shown

<table>
<thead>
<tr>
<th>Tasks</th>
<th>MILP*</th>
<th>CP</th>
<th>Benders</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.9</td>
<td>0.13</td>
<td>0.37</td>
</tr>
<tr>
<td>12</td>
<td>797</td>
<td>2.6</td>
<td>0.55</td>
</tr>
<tr>
<td>14</td>
<td>114</td>
<td>35</td>
<td>0.34</td>
</tr>
<tr>
<td>16</td>
<td>678*</td>
<td>1929</td>
<td>4.5</td>
</tr>
<tr>
<td>18</td>
<td>7200+</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>2.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>53</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*CPLEX ran out of memory on 1 or more problems.
+ At least one problem in the 5 exceeded 7200 sec (2 hours)
Min cost, 4 facilities

Computation time in seconds
Average of 5 instances shown

<table>
<thead>
<tr>
<th>Jobs</th>
<th>MILP*</th>
<th>CP</th>
<th>Benders</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>2.0</td>
<td>0.10</td>
<td>0.6</td>
</tr>
<tr>
<td>12</td>
<td>7.2</td>
<td>1.4</td>
<td>4.0</td>
</tr>
<tr>
<td>14</td>
<td>158</td>
<td>72</td>
<td>2.8</td>
</tr>
<tr>
<td>16</td>
<td>906*</td>
<td>344</td>
<td>0.8</td>
</tr>
<tr>
<td>18</td>
<td>6343+</td>
<td></td>
<td>5.2</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td>2.6</td>
</tr>
<tr>
<td>22</td>
<td></td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td>114</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td>76</td>
<td></td>
</tr>
</tbody>
</table>

*CPLEX ran out of memory on 1 or more problems.

+ At least one problem in the 5 exceeded 7200 sec (2 hours)
Min makespan, 2 facilities
Average of 5 instances shown

<table>
<thead>
<tr>
<th>Jobs</th>
<th>MILP</th>
<th>CP</th>
<th>Benders</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>3.4</td>
<td>0.8</td>
<td>0.08</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>4.0</td>
<td>0.39</td>
</tr>
<tr>
<td>14</td>
<td>2572+</td>
<td>299</td>
<td>7.8</td>
</tr>
<tr>
<td>16</td>
<td>5974+</td>
<td>3737</td>
<td>30</td>
</tr>
<tr>
<td>18</td>
<td>7200+</td>
<td>461</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>2656</td>
<td></td>
</tr>
</tbody>
</table>

+ At least one problem in the 5 exceeded 7200 sec (2 hours)
Min makespan, 3 facilities
Average of 5 instances shown

<table>
<thead>
<tr>
<th>Jobs</th>
<th>MILP</th>
<th>CP</th>
<th>Benders</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>3.9</td>
<td>0.9</td>
<td>0.06</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>7.5</td>
<td>0.3</td>
</tr>
<tr>
<td>14</td>
<td>524</td>
<td>981</td>
<td>0.7</td>
</tr>
<tr>
<td>16</td>
<td>1716+</td>
<td>4414</td>
<td>6.5</td>
</tr>
<tr>
<td>18</td>
<td>4619+</td>
<td>7200+</td>
<td>13.3</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td>3084+</td>
<td></td>
</tr>
</tbody>
</table>

+ At least one problem in the 5 exceeded 7200 sec (2 hours)
Min makespan, 4 facilities

Average of 5 instances shown

<table>
<thead>
<tr>
<th>Jobs</th>
<th>MILP</th>
<th>CP</th>
<th>Benders</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1.0</td>
<td>0.07</td>
<td>0.09</td>
</tr>
<tr>
<td>12</td>
<td>5.0</td>
<td>1.9</td>
<td>0.09</td>
</tr>
<tr>
<td>14</td>
<td>24</td>
<td>524</td>
<td>0.8</td>
</tr>
<tr>
<td>16</td>
<td>35</td>
<td>3898</td>
<td>0.9</td>
</tr>
<tr>
<td>18</td>
<td>3931+</td>
<td>7200+</td>
<td>14</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td>472</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td>1131</td>
<td></td>
</tr>
</tbody>
</table>

+ At least one problem in the 5 exceeded 7200 sec (2 hours)
Scaling up the Benders Method

Average of 5 instances shown

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Facilities</th>
<th>Min cost (sec)</th>
<th>Min makespan (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>2</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td>15</td>
<td>3</td>
<td>0.7</td>
<td>1.6</td>
</tr>
<tr>
<td>20</td>
<td>4</td>
<td>50</td>
<td>13</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2.9</td>
<td>213</td>
</tr>
<tr>
<td>30</td>
<td>6</td>
<td>4.8</td>
<td>3373+</td>
</tr>
<tr>
<td>35</td>
<td>7</td>
<td>128</td>
<td>6404+</td>
</tr>
<tr>
<td>40</td>
<td>8</td>
<td>1792+</td>
<td>7200+</td>
</tr>
</tbody>
</table>

+ At least one problem in the 5 exceeded 7200 sec (2 hours)
Bounds Provided by Benders
Min makespan problems unsolved after 2 hours

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Facilities</th>
<th>Best solution value</th>
<th>Lower bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>6</td>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>35</td>
<td>7</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>35</td>
<td>7</td>
<td>15</td>
<td>13</td>
</tr>
<tr>
<td>40</td>
<td>8</td>
<td>14</td>
<td>11</td>
</tr>
<tr>
<td>40</td>
<td>8</td>
<td>15</td>
<td>12</td>
</tr>
<tr>
<td>40</td>
<td>8</td>
<td>16</td>
<td>13</td>
</tr>
<tr>
<td>40</td>
<td>8</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>40</td>
<td>8</td>
<td>13</td>
<td>11</td>
</tr>
</tbody>
</table>
### Min tardiness, 3 facilities

#### Smaller problems

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Time (sec)</th>
<th>Min tardiness</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CP</td>
<td>MILP</td>
</tr>
<tr>
<td>10</td>
<td>13</td>
<td>4.7</td>
</tr>
<tr>
<td></td>
<td>1.1</td>
<td>6.4</td>
</tr>
<tr>
<td></td>
<td>1.4</td>
<td>6.4</td>
</tr>
<tr>
<td></td>
<td>4.6</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>8.1</td>
<td>33</td>
</tr>
<tr>
<td>12</td>
<td>4.7</td>
<td>0.7</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>0.6</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>0.7</td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>317</td>
<td>25</td>
</tr>
<tr>
<td>14</td>
<td>838</td>
<td>7.0</td>
</tr>
<tr>
<td></td>
<td>7159</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>1783</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>&gt;7200</td>
<td>73</td>
</tr>
<tr>
<td></td>
<td>&gt;7200</td>
<td>&gt;7200</td>
</tr>
<tr>
<td>16</td>
<td>&gt;7200</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>&gt;7200</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>&gt;7200</td>
<td>52</td>
</tr>
<tr>
<td></td>
<td>&gt;7200</td>
<td>1105</td>
</tr>
<tr>
<td></td>
<td>&gt;7200</td>
<td>3424</td>
</tr>
</tbody>
</table>
Min tardiness, 3 facilities

Larger problems

On all problems:
average time ratio MILP/Benders = 20
Future Research

• Implement branch-and-check for Benders problem.
• Exploit dual information from the subproblem solution process (e.g. edge finding).
• Explore other problem classes.
  • Integrated long- and short-term scheduling
  • Vehicle routing
  • SAT (subproblem is renamable Horn)
  • Stochastic IP