Solving an integrated Job-Shop problem with human resource constraints

PMS'10 - Tours (France)

O. Guyon[1.2], P. Lemaire[3], É. Pinson[2] and D. Rivreau[2]

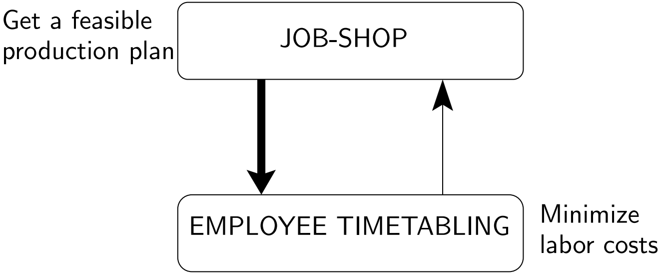
[1] École des Mines de Saint-Étienne
[2] LISA - Institut de Mathématiques Appliquées d'Angers
[3] Institut Polytechnique de Grenoble

April 26-28, 2010

Table of contents

Plan

Nature of the problem (1/3)



Nature of the problem (2/3) - Employee Timetabling problem

Time horizon

- Timetabling horizon $H = \delta \cdot \pi$ where:
  - δ number of shifts
  - π duration time of a shift

Nota: it can modelize, for example, a three-shift system

Employee Timetabling Problem for a set E of μ employees

- A_e set of machines employee e masters
- $\mathcal{T}_e$ set of shifts where employee e is available

Nature of the problem (3/3) - Job-Shop problem

Job-Shop: Schedule a set J of n jobs on m machines

- $\forall j \in J$ $\{O_{ji}\}_{i=1..m}$ chain of operations of job j
  - machine $m_{ji} \in \{1 \ldots m\}$
  - processing time p_{ji}
    $\hookrightarrow$ *Notation* ρ_{jk}: processing time of j on machine k
  - can not be interrupted
  - requires a qualified employee to use machine m_{ji}

Feasible production plan

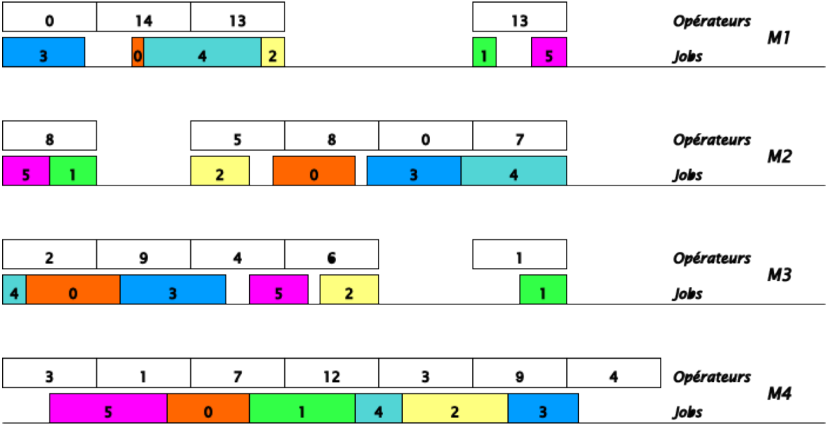
A schedule for which all operations are completed before a given scheduling completion time Cmax $\leq$ H.

Nature of the problem (3/3)

Objective

Assigning at minimum cost employees to both machines and shifts in order to be able to provide a *feasible production plan*

Example (6 jobs - 4 machines - 15 employees)



Motivation: Extension of work

Continuation of a work

- Guyon, Lemaire, Pinson and Rivreau.
  European Journal of Operational Research (March 2010)
- Integrated employee timetabling and production scheduling problem
- *Methods*:
  $\hookrightarrow$ Decomposition and cut generation process
- $\rightsquigarrow$ Simplified production scheduling problem
- Motivation for this new study:
  $\hookrightarrow$ Is the decomposition and cut generation process also efficient with a harder production scheduling problem (Job-Shop)?

Motivation: Case treated in literature

Case treated in literature

- Artigues, Gendreau, Rousseau and Vergnaud [AGRV09].
  Computers and Operations Research (2009)
- Aim: To experiment hybrid CP-ILP methods on an integrated job-shop scheduling and employee timetabling problem
- *Methods*:
  ↪ CP with a global additional constraint corresponding to the LP-relaxation of the employee timetabling problem
- *Our study* : specific case of mapping activities - machines
  ↪ 8 of the 11 instances of [AGRV09] can be used

Plan

1. Introduction
   - Problem
   - Time-indexed ILP formulation
   - Solution methods
2. 0-1 Branch and Bound
   - Outline
   - Initial cuts
   - Characteristics
3. Conclusion
   - Experimental results
   - Concluding remarks

ILP model (1/4)

Binary decision variables

- $x_{eks} = 1$ iif employee e is assigned to the pair (machine k; shift s)
- $y_{ikt} = 1$ iif job i starts its processing on the machine k at instant t

ILP model (2/4)

Objective function

$$[P] \quad \min \Theta = \sum_{e \in E} \sum_{k \in A_e} \sum_{s \in T_e} c_{eks} \cdot x_{eks}$$

Employee Timetabling Problem specific constraints

$$\begin{array}{rll}
\sum_{k \notin A_e} \sum_{s=0}^{\sigma} x_{eks} = 0 & e = 1, \dots, \mu & \\
\sum_{k \in A_e} \sum_{s \notin \mathcal{T}_e} x_{eks} = 0 & e = 1, \dots, \mu & \\
\sum_{k \in A_e} (x_{eks} + x_{ek(s+1)} + x_{ek(s+2)}) \leq 1 & e = 1, \dots, \mu & s = 0, \dots, \sigma - 3 \\
x_{eks} \in \{0, 1\} & e = 1, \dots, \mu & k = 1, \dots, m \\
 & s = 0, \dots, \sigma - 1 &
\end{array}$$

ILP model (3/4)

Job-Shop scheduling problem specific constraints

$$\sum_{t=0}^{d_{ik}-\rho_{ik}} t \cdot y_{ikt} + \rho_{ik} \leq C_{max} \qquad i = 1, \ldots, n \quad k = m_{im}$$

$$\sum_{t=r_{ik}}^{d_{ik}-\rho_{ik}} y_{ikt} = 1 \qquad i = 1, \ldots, n \quad k = 1, \ldots, m$$

$$\sum_{t=0}^{r_{ik}} y_{ikt} + \sum_{t=d_{ik}-\rho_{ik}+1}^{C_{max}} y_{ikt} = 0 \qquad i = 1, \ldots, n \quad k = 1, \ldots, m$$

$$\sum_{u=r_{ik}+\rho_{ik}}^{t} y_{ilu} - \sum_{u=r_{ik}}^{t-\rho_{ik}} y_{iku} \leq 0 \qquad i = 1, \ldots, n \quad j = 1, \ldots, m-1$$

$$k = m_{ij} \quad l = m_{i(j+1)}$$

$$t = \rho_{ik} + p_{ik}, \ldots, d_{il} - \rho_{il}$$

$$\sum_{i=1}^{n} \sum_{u=\max(r_{ik}, t-\rho_{ik}+1)}^{\min(d_{ik}-\rho_{ik}, t)} y_{iku} \leq 1 \qquad k = 1, \ldots, m \quad t = 0, \ldots, C_{max}$$

$$y_{ikt} \in \{0, 1\} \qquad i = 1, \ldots, n \quad k = 1, \ldots, m$$

$$t = 0, \ldots, C_{max}$$

ILP model (4/4)

Coupling constraints

$$\sum_{e \in E} x_{eks} - \sum_{i=1}^{n} \sum_{u=\max(r_{ik}, t-\rho_{ik}+1)}^{\min(d_{ik}-\rho_{ik}, t)} y_{iku} \geq 0 \quad k = 1, \ldots, m \quad t = 0, \ldots, C_{max}$$

$$s = \lfloor t/\pi \rfloor$$

Plan

Solution methods

Three exact methods

- MIP
- Decomposition and cut generation process
- 0-1 Branch and Bound based on the work (or not) for each pair (machine, shift)

Plan

Outline

Underlying ideas

- Splitting of $[P]$ (by relaxing coupling constraints)
  - $[Job - Shop]$ (non-coupling constraints; with variables y_{jkt})
  - $[Employee]$ (non-coupling constraints; with variables x_{eks})
- Fixing an assignment $\bar{z}$ of worked and unworked pairs (machine, shift)
- Checking the feasibility of $\bar{z}$ at two levels:
  - $[Job - Shop]$ which is solved with a dedicated Job-Shop solver
  - $[Employee]$ which is solved with an ILP solver
- To avoid an exhaustive search:
  $\hookrightarrow$ $\bar{z}$ is generated through a 0-1 Branch and Bound coupled with a generation of
  - 3 sets of initial cuts
  - feasibility cuts (generated all along the process)

Plan

Probing initial cuts

What?

Test the obligation of work for each pair (machine k, shift s)

How?

- create a fictive job j_f that has to be processed on k over s for a duration of π
- solve [Job-Shop] with a dedicated Job-Shop solver
- if [Job-Shop] is unfeasible ($\Leftrightarrow$ does not respect $Cmax$) :
  - j_f ($\Leftrightarrow$ absence of work on (k, s)) can not be scheduled
  - an employee must be assigned to (k, s)

Result

- set of pairs (k, s) fixed to work
- cuts for $[Employee] \Rightarrow \sum_{e=1}^{\mu} x_{eks} = 1 \ \forall \ (k, s)$ fixed to work

Capacitated cuts

Capacitated cut for machine k

Find a close lower bound b_k of the minimal number of worked shifts on k

How? b_k is the maximum of 3 valid lower bounds

1. direct: number of pairs (k, s) fixed by probing
2. direct: $\left\lceil \frac{\sum_{j=1}^{n} \rho_{jk}}{\pi} \right\rceil$
3. Fix (by probing) the maximum number of schedulable fictive jobs

Result: Cuts for [*Employee*]

$$\sum_{e=1}^{\mu} \sum_{s \in \mathcal{T}_e} x_{eks} \geq b_k \quad k = 1, \ldots, m$$

Non-overlapping cuts

Non-overlapping cuts for each pair (machine k, shift s)

To ensure each feasible solution to get at most one employee assigned to (k, s)

Result: Cuts for [*Employee*]

$$\sum_{e \in E | (k \in A_e) \wedge (s \in T_e)} x_{eks} \leq 1 \quad k = 1, \ldots, m \quad s = 0, \ldots, \delta - 1$$

Plan

Characteristics (1/4)

Binary separation scheme

- left child: impose a given pair (machine $\bar{k}$, shift $\bar{s}$) to be not worked
- right child: impose the same given pair $(\bar{k}, \bar{s})$ to be worked

Exploration strategy

depth-first

Branching pair (machine $\bar{k}$, shift $\bar{s}$)

- $\bar{k}$: machine with a maximum gap between the number of shifts fixed (or which can still be fixed) to work and the lower bound LB_k of the minimum number of worked shifts on $\bar{k}$
- $\bar{s}$: latest shift such that $(\bar{k}, \bar{s})$ has not been fixed yet

Characteristics (2/4)

Definition

- *Relaxed assignment*
  Assignment of pairs (k, s) such that each fixed pair (k, s) is fixed to its value and all the other ones are free
- *Strict assignment*
  Assignment of pairs (k, s) such that each fixed pair (k, s) is fixed to its value and all the other ones are imposed to be *unworked*

Evaluation

LP-relaxation of [*Employee*] for the *relaxed assignment*

Characteristics (3/4)

Implications rules

If the decision is: *the pair* $(\bar{k}, \bar{s})$ *must be unworked*
We try to fix pairs (machine, shift) by:

- using probing techniques for each non-fixed pair $(\bar{k}, s)$
- checking the respect of the lower bound $b_{\bar{k}}$ of the minimal number of worked shifts on $\bar{k}$

Elimination rules

A branching node can be pruned if one of these two conditions is fufilled:

1. [*Employee*] with *relaxed assignment* is unfeasible (evaluation) or has a cost greater than the best current known
2. *Relaxed assignment* is unfeasible for [*Job* – *Shop*]

Characteristics (4/4)

If the current node has not been pruned

- We get the optimum $(\bar{x}^*, \bar{\theta}^*)$ for $[Employee]$ that respects the *strict assignment*
  - if $\bar{x}^*$ exists, we check if the *strict assignment* is feasible for $[Job - Shop]$
    - if it is, UB is updated: $UB \leftarrow \bar{\theta}^*$
- If UB is not updated: we add a feasibility cut to $[Employee]$

Feasibility cut for $[Employee]$

- permits to eliminate solutions similar to $\bar{x}^*$
- impose the work on at least one of the unworked pairs (k, s) of the *strict assignment*
- Cut : $\sum_{(k,s)\text{not fixed}} \sum_{e=1}^{\mu} x_{eks} \geq 1$

Plan

1. Introduction
   - Problem
   - Time-indexed ILP formulation
   - Solution methods
2. 0-1 Branch and Bound
   - Outline
   - Initial cuts
   - Characteristics
3. Conclusion
   - Experimental results
   - Concluding remarks

Test bed

8 instances of [AGRV09]

Instance	n	m	μ	μ_{extra}	δ	π
ejs	6	4	25	10	8	8
$ejs8X8$	8	8	40	20	8	8
$ejs10X10$	10	10	50	20	10	10

Tools

- Computer programming language:
  Java except for [AGRV09] (C++)
- MIP solver: Ilog Cplex 12.1
- CP solvers (for [AGRV09]): Ilog Solver 6.7; Ilog Scheduler 6.7
- Job-Shop solver:
  Branch & Bound (*Carlier, Péridy, Pinson, Rivreau*)
- Processor: Intel Core 2 Quad Q6600 @ 2,40 GHz - 3 GB RAM
- CPU time limit: 300 seconds

Results: initial cuts

Impact of the initial cuts

	LP relaxation of $[P]$		LP' relaxation of $[P]$ with initial cuts			
Instance	LP/Θ^*	Time	LP'/Θ^*	#cuts	Preprocess time	Total time
ejs4	92.4%	0.2s	100.0%	43	0.0s	0.1s
ejs9	87.3%	0.3s	94.6%	67	1.5s	0.3s
ejs10	97.1%	0.5s	100.0%	50	0.4s	0.2s
ejs8 $\times$ 8_1	60.1%	1.6s	86.2%	94	9.5s	10.8s
ejs8 $\times$ 8_2	68.0%	2.5s	91.6%	103	7.9s	9.6s
ejs8 $\times$ 8_3	55.1%	1.6s	91.8%	93	4.6s	6.0s
ejs10 $\times$ 10_1	55.7%	11.4s	93.1%	158	17.4s	24.4s
ejs10 $\times$ 10_3	69.1%	18.1s	88.9%	164	218.8s	233.2s

Results: exact methods

Results of the exact methods

		MIP		[AGRV09]		0-1 Branch and Bound			
Instance	Θ^*	Θ	Time	Θ	Time	Θ	Preprocessing time	Total time	#initial cuts + #cuts
ejs4	23	23	0.2s	23	0.9s	23	0.0s	0.0s	43 + 0
ejs9	24	24	11.4s	24	87.9s	24	1.5s	5.0s	67 + 44
ejs10	23	23	1.0s	23	3.5s	23	0.4s	0.6s	50 + 2
ejs8 $\times$ 8_1	78	84	TL	78	64.8s	78	9.5s	37.8s	94 + 66
ejs8 $\times$ 8_2	96	96	TL	96	98.9s	96	7.9s	43.3s	103 + 108
ejs8 $\times$ 8_3	83	83	TL	83	29.7s	83	4.5s	9.0s	93 + 15
ejs10 $\times$ 10_1	124	-1	TL	137	TL	124	17.5s	33.0s	158 + 31
ejs10 $\times$ 10_3	95	-1	TL	150	TL	102	218.8s	TL	164 + 251

Plan

Concluding remarks

- Strong initial cuts (especially probing cuts)
- Interesting decomposition approach
- 0-1 Branch and Bound really competitive