An Exact Method for the Best Case in a Group Sequence: Application to a General Shop Problem

Guillaume Pinot Nasser Mebarki

IRCCyN — UMR CNRS 6597
Nantes, France
`firstname.lastname@irccyn.ec-nantes.fr`

INCOM 2009

Table of Contents

1. Introduction
2. Group Sequencing
3. Lower Bounds
4. The Exact Method
5. Experiments
6. Conclusion

Introduction

Group sequencing:

- is a scheduling method;
- describes a set of schedules;
- guarantees a minimal quality corresponding to the worst case.

A best-case evaluation of a group sequence could be interesting.

Group Sequencing

Group sequencing:

- provides sequential flexibility during the execution of the schedule;
- guarantees a minimal quality corresponding to the worst case.

To manage sequential flexibility, usage of “groups of permutable operations.”

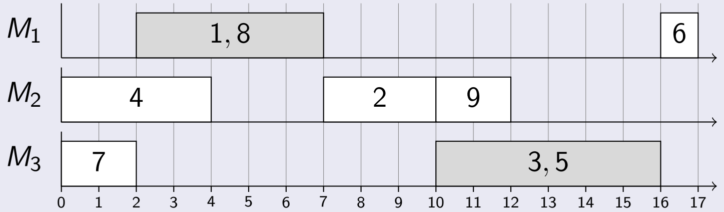
Example: a Job Shop Problem

i: the index of the operations, $\Gamma^-(i)$: the set of the predecessors of O_i, m_i: the resource needed by O_i, p_i: the processing time needed by O_i.

A Job Shop Problem

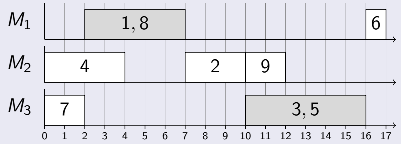
i	1	2	3	4	5	6	7	8	9
$\Gamma^-(i)$	$\varnothing$	$\{1\}$	$\{2\}$	$\varnothing$	$\{4\}$	$\{5\}$	$\varnothing$	$\{7\}$	$\{8\}$
m_i	M_1	M_2	M_3	M_2	M_3	M_1	M_3	M_1	M_2
p_i	3	3	3	4	3	1	2	2	2

A Solution to This Problem

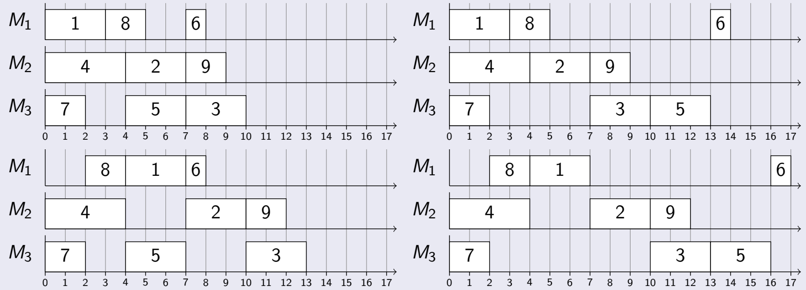


Execution of the Example

The Group Sequence



The Corresponding Semi-Active Schedules



Why is Group Sequencing Interesting?

Why is group sequencing interesting?

- predictive reactive method;
- flexibility on sequences;
- widely studied in the last twenty years: [Erschler and Roubellat, 1989, Wu et al., 1999, Artigues et al., 2005];
- no need to model the uncertainties;
- the method is able to absorb some uncertainties: [Wu et al., 1999, Esswein, 2003, Pinot et al., 2007];
- evaluation of the group sequence in the worst case in polynomial time for *minmax* regular objectives as $C_{\max}$ and $L_{\max}$.

The best-case evaluation of a group sequence could be usefull.

The Best Case Completion time of an Operation

$$\begin{cases} \theta_i = \max\left(r_i, \gamma_{g^-(i)}, \max\limits_{j\in\Gamma^-(i)} \chi_j\right) \\ \chi_i = \theta_i + p_i \\ \gamma_{g_{\ell,k}} = C_{\max} \text{ of } 1|r_i|C_{\max}, \forall O_i \in g_{\ell,k}, r_i = \theta_i \end{cases}$$

θ_i Best case lower bound for starting time of O_i

χ_i Best case lower bound for completion time O_i

$\gamma_{g_{\ell,k}}$ Lower bound for the completion time of $g_{\ell,k}$

It can be used to calculate a lower bound for any objective.

$$\mathrm{LB}(L_{\max}) = \max_{\forall O_i} L_i(\chi_i) = \max_{\forall O_i}(\chi_i - d_i)$$

Makespan Lower Bound

Classical job-shop lower bound: one-machine-problem relaxation [Carlier, 1982] on each machine.
The one-machine-problem relaxation require some tools:

- a head for each operations: θ_i;
- a tail for each operations: a reversed θ_i.

For group sequencing the relaxation is done on groups instead of machines (more subproblems, but smaller).
Solving the one-machine problems is done using the exact Carlier's algorithm [Carlier, 1982].

Presentation

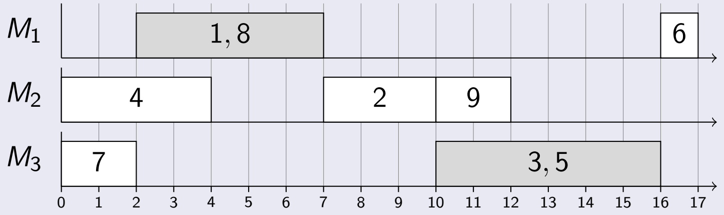
An exact method to find the optimal solution for any regular objective.
This method is a branch and bound algorithm:

- the branching procedure is based on active schedules;
- lower bound presented before.

Enumerating Active Schedules

Enumerating active schedules group by group (according to the precedence graph):

The Problem



A Valid Order

$$[\{O_4\},\{O_7\},\{O_1,O_8\},\{O_2\},\{O_9\},\{O_3,O_5\},\{O_6\}] \Rightarrow [\{O_1,O_8\},\{O_3,O_5\}]$$

Reducing the Search Space

The completion time of an operation interfere with the objective function:

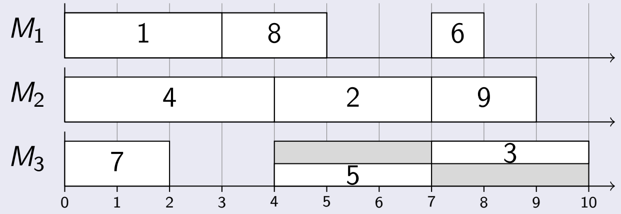
- the completion time, because the objective function is a function of the completion times;
- by interfering with the completion time of the other operations, because of precedence constraints or resource constraints.

A sufficient condition for the sequencing of an entire group without losing the optimal solution is:

- the sequencing does not degrade the objective function;
- the sequencing does not interfere on the earliest starting time of the operations with successor constraints and resource constraints.

Example

The group sequence: sequencing $\{1, 8\}$



The corresponding one machine problem

i	r_i	p_i	$\tilde{d}_i$
1	0	3	4
8	2	2	7

The solution $[1; 8]$:

- do not modify the starting time of the predecessors;
- do not modify the *makespan*.

⇒ this group can be sequenced without losing the optimal solution.

Searching Strategies

Exploring the search tree:

- Deep first:
  - Avantage: small amound of memory needed;
  - Drawback: a bad decision can be costy.
- Best bound first:
  - Avantage: no bad decision possible;
  - Drawback: lots of memory needed.

Selected solution: best bound first, then, when a given amound of nodes is stored, deep first.

Protocol

Instances : `la01` to `la40` from [Lawrence, 1984].
For each instances, we generate a group sequence with

- a known optimal makespan[Brucker et al., 1994];
- a very high flexibility [Esswein, 2003].

Different variants:

- Default :
  - the sufficient condition is used;
  - best-bound search is used until 1000 nodes are stored.
- Deep search: same as Default with deep search;
- No sufficient condition: same as Default without using the sufficient condition.

Results

Results of Default by size:

- Instances with 5 machines : < 1s;
- 10×10 et 15×10: < 1min (except la24: 14min);
- 30×10: < 4s;
- 20×10 and 15×15: 4 not solved in 24h on 10.

Comparison of Default with the other variants:

- Deep search:
  - in average 20 times slower;
  - faster on 4 instances of size 10×10;
- No sufficient condition:
  - in average 3 times slower;
  - never better;
  - 28 times slower on la17.

Conclusion

An exact method solving the best-case in a group sequence:

- for every regular objective;
- uses a lower bound based on the one-machine relaxation;
- enunerates active schedules;
- uses a dedicated method to reduce the search space.

An Exact Method for the Best Case in a Group Sequence: Application to a General Shop Problem

Guillaume Pinot Nasser Mebarki

IRCCyN — UMR CNRS 6597
Nantes, France
firstname.lastname@irccyn.ec-nantes.fr

INCOM 2009

Bibliography I

Artigues, C., Billaut, J.-C., and Esswein, C. (2005).
Maximization of solution flexibility for robust shop scheduling.
European Journal of Operational Research, 165(2):314–328.

Brucker, P., Jurisch, B., and Sievers, B. (1994).
A branch and bound algorithm for the job-shop scheduling problem.
Discrete Applied Mathematics, 49(1-3):107–127.

Carlier, J. (1982).
The one-machine sequencing problem.
European Journal of Operational Research, 11(1):42–47.

Bibliography II

Erschler, J. and Roubellat, F. (1989).
An approach for real time scheduling for activities with time and resource constraints.
In Slowinski, R. and Weglarz, J., editors, *Advances in project scheduling*. Elsevier.

Esswein, C. (2003).
Un apport de flexibilité séquentielle pour l'ordonnancement robuste.
Thèse de doctorat, Université François Rabelais Tours.

Lawrence, S. (1984).
Resource constrained project scheduling: an experimental investigation of heuristic scheduling techniques (supplement).
Technical report, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, Pennsylvania.

Bibliography III

- Pinot, G., Cardin, O., and Mebarki, N. (2007).
  A study on the group sequencing method in regards with transportation in an industrial FMS.
  In *Proceedings of the IEEE SMC 2007 International Conference*.
- Wu, S. D., Byeon, E.-S., and Storer, R. H. (1999).
  A graph-theoretic decomposition of the job shop scheduling problem to achieve scheduling robustness.
  Operations Research, 47(1):113–124.

Results for the Hard Instances After 24h of Computation

Size	Inst.	Opt.	LB	Nodes	UB	Nodes	Tot. Nodes
20 × 10	la27	1252*	1235	0	1279	5150695	9500000
20 × 10	la29	1202	1202	3836	1221	10343	10000000
20 × 10	la30	1355	1355	0	1359	2911199	12500000
15 × 15	la37	1397	1397	2	1412	7623146	9700000