

# Ordonnancement d'atelier avec lissage de consommation d'énergie dans une industrie chimique.

Quentin Christ<sup>1</sup>, Anne-Laure Ladier<sup>2</sup>, Bernard Penz<sup>2</sup>

<sup>1</sup> Grenoble INP Génie industriel  
46 avenue Félix Viallet, 38031 Grenoble cedex 1  
Quentin.Christ@grenoble-inp.org

<sup>2</sup> Univ. Grenoble Alpes, G-SCOP, F-38000 Grenoble, France  
CNRS, G-SCOP, F-38000 Grenoble, France  
{Anne-Laure.Ladier, Bernard.Penz}@g-scop.fr

**Mots-clés :** *Ordonnancement, contraintes de ressources, industrie chimique*

## 1 Introduction

Nous présentons ici l'adaptation d'un modèle d'ordonnancement d'atelier sous contraintes de ressources, pour répondre aux contraintes spécifiques de notre partenaire industriel dans l'industrie chimique.

L'essentiel de la production de l'entreprise considérée, que nous ne nommerons pas pour des raisons de confidentialité, provient d'un atelier fonctionnant 7j/7 et 24h/24. La fabrication s'effectue sur 26 machines, sous la supervision d'opérateurs très qualifiés. Une fois lancée, une machine ne peut être arrêtée avant la fin de la production, qui peut durer de 6 à 24 heures. Jusqu'à présent, le planning de démarrage des machines était fixe et reproduit quasiment à l'identique d'un jour à l'autre.

Le démarrage d'une machine est particulièrement coûteux en gaz, une ressource dont l'approvisionnement est récemment devenu incertain pour des causes extérieures à l'entreprise. Or le planning routinier fait démarrer toutes les machines en début de poste, ce qui engendre une consommation de gaz très importante en début de poste et beaucoup plus faible en fin de poste. Afin de pouvoir assurer une production maximale lorsque l'approvisionnement de gaz est limité, on cherche à établir un nouveau planning de démarrage des machines, qui lisse autant que possible cette consommation.

## 2 Approche de résolution

Il existe de nombreux modèles d'ordonnancement sous contrainte de ressources (voir par exemple Artigues *et al.* [1]). Nous avons choisi d'adapter le modèle de *resource-constrained project scheduling problem* en temps discret (RCPSP-DT) proposé par Pritsker *et al.* [2]. Les adaptations réalisées permettent de s'adapter aux spécificités de notre cadre industriel.

Dans le modèle RCPSP, les ressources sont intégrées sous forme de contraintes de limites à ne pas dépasser, contrairement à notre cas industriel où les ressources représentent aussi l'élément à optimiser. L'intégration de cette dimension ne se fait donc plus seulement sous forme de contrainte mais également via la fonction objectif. Afin de lisser la consommation en ressources, cette fonction intègre la moyenne des variations successives (en valeur absolue), ainsi que la variation de consommation maximale (en valeur absolue).

A l'inverse du modèle RCPSp, on ne cherche pas à minimiser la date de fin de projet (*makespan*) ; on souhaite au contraire que l'essentiel de la production soit effectuée dans l'horizon de planification, c'est-à-dire les 40 heures à venir. On utilise donc une contrainte souple, basée sur la consommation moyenne réalisée au-delà des 40 heures, pour s'assurer que l'essentiel de la production soit regroupée au sein de l'horizon d'étude.

La fonction objectif est donc une somme pondérée de quatre termes : la moyenne des variations en valeur absolue ; la variation maximale en valeur absolue ; la contrainte souple énoncée dans le paragraphe précédent ; un malus pénalisant le non-respect des priorités de production. Tous ces éléments ayant le même niveau d'importance, les pondérations sont ajustées afin de leur donner le même ordre de grandeur.

Enfin, nous avons ajouté au modèle de base une vingtaine de contraintes spécifiques à l'atelier considéré. On peut citer notamment la prise en compte de périodes d'indisponibilité des machines, le fait d'imposer le démarrage de certaines machines à la demande du manager, la prise en compte de priorités de production, du nombre d'ouvriers disponibles pour superviser la production, du débit limite disponible pour chacun des postes de l'horizon d'étude, de la différenciation des machines selon certains critères (temps de fonctionnement, type de matière produite) permettant de les affectées à des contraintes spécifiques, telles que des plages possibles de démarrage.

### 3 Résultats et perspectives

L'entreprise favorisait une solution utilisant Excel, déjà maîtrisé par les futurs utilisateurs. Le programme linéaire en nombres entiers a été implémenté en utilisant Open Solver 2.5 [3], un solveur PL/PLNE open-source pour Excel, basé sur le moteur d'optimisation COIN-OR. L'interface utilisateur est un document Excel, utilisée sur ordinateur équipé d'un processeur de 3,4 GHz et 8 Go de RAM.

L'outil d'aide à la décision est aujourd'hui utilisé quotidiennement par les managers de l'entreprise. Le temps d'exécution est de l'ordre de 40 secondes, comprenant environ 20 secondes de chargement du modèle, puis 20 secondes de résolution au bout desquelles le programme s'arrête automatiquement et retourne la meilleure solution trouvée. Nous avons montré expérimentalement qu'au bout de ce délai, l'objectif principal (lissage de la consommation en gaz) est atteint : l'écart-type de la consommation de gaz sur tous les pas de temps considérés diffère de celui de la solution optimale de moins d'1%.

Sur une période de tests de 2 semaines, nous avons montré que la variation de consommation de gaz (différence entre la consommation maximum et la consommation minimum sur une semaine) a été réduite de 34%.

Une perspective possible est de modifier le fait que le modèle impose le démarrage de certaines machines à la demande du manager. Cet aspect est formalisé sous forme d'une contrainte dure, ce qui peut rendre le modèle infaisable si le manager impose un ensemble de lancements impossible à satisfaire. Il est alors difficile de déterminer la tâche précise qui empêche la résolution du programme, malgré un code couleur dans l'interface graphique permettant de repérer les machines qui risquent d'arriver à saturation. Rendre souple la contrainte du respect du nombre de lancements permettrait de limiter le risque d'infaisabilité, facilitant aussi au manager la recherche des tâches problématiques.

### Références

- [1] C. Artigues, S. Demasse, and E. Néron, *Resource-Constrained Project Scheduling: Models, Algorithms, Extensions and Applications*, vol. 2010. 2010, p. 288.
- [2] A.Pritsker, L.Watters, and P.Wolfe, "Multi-project scheduling with limited resources: A zero-one programming approach", *Management Science*, 16: 93-108,1969.
- [3] <http://opensolver.org>.