

Caractérisation des solutions optimales du $1||L_{max}$

Mallory Taffonneau¹, Jean-Charles Billaut^{1,2}, Ronan Bocquillon¹

¹ EA 6300, LIFAT, Université de Tours, 37000 Tours

{(jean-charles.billaut,ronan.bocquillon)}@univ-tours.fr

² LAAS CNRS, Université de Toulouse, CNRS, Toulouse.

Mots-clés : *recherche opérationnelle, optimisation.*

1 Introduction

Il est bien connu que certains problèmes d'ordonnancement ont un nombre de solutions optimales potentiellement exponentiel. C'est le cas du problème $1||L_{max}$. On considère ce problème d'ordonnancement et des instances construites de la façon suivante. Pour une instance I quelconque, les travaux sont triés et renumérotés dans l'ordre des dates dues croissantes. Ensuite, on ajoute à chaque date due la valeur du L_{max} optimal. Chaque date due devient une date de fin au plus tard impérative. On obtient ainsi une instance I' et pour cette instance, on sait que la séquence $(1, 2, \dots, n)$ est réalisable.

Le graphe des permutations est un outil mathématique (une structure de treillis) qui permet de représenter l'ensemble des permutations de n éléments sous une forme relativement simple à comprendre. La séquence $S = \sigma ij\pi$ a pour descendantes les séquences de la forme $\sigma ji\pi$ pour tous les couples (i, j) consécutifs dans S et tels que $i < j$.

Le graphe des permutations pour $n = 4$ est donné figure 1.

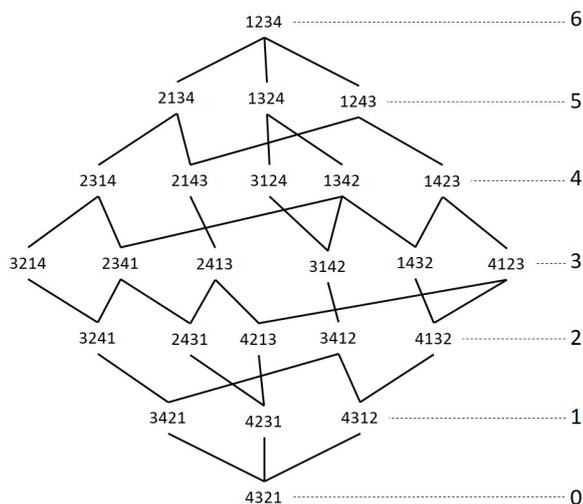


FIG. 1 – Permutoèdre pour $n = 4$

Un sommet quelconque de cette structure a des propriétés intéressantes, qui permettent de donner les caractéristiques de tous ses prédécesseurs. Si on considère que chaque sommet est une séquence, et si un sommet correspond à une séquence réalisable, alors on sait caractériser très facilement tous ses prédécesseurs, qui se trouvent être également des solutions réalisables.

Le problème consiste à trouver dans cette structure la séquence de niveau le plus bas possible. Ce problème a déjà fait l'objet de nombreuses études [1, 2, 3, 4].

2 L'ancien et le nouveau

Plusieurs méthodes de résolution exactes ont été proposées pour ce problème : trois modèles de programmation linéaire et un algorithme de branch-and-bound [4] avec une borne inférieure efficace en $O(1)$. Ces méthodes ont été testées et les résultats étaient largement en faveur du branch-and-bound pour certaines instances, mais plus mitigés pour d'autres.

Nous proposons une nouvelle méthode exacte, qui consiste à vérifier s'il existe une séquence réalisable à un niveau donné N du treillis, et à procéder ensuite par dichotomie (le nombre de niveaux dans le treillis est exactement $n(n-1)/2$) pour trouver le niveau le plus bas qui contient une séquence réalisable. Le problème se présente de la façon suivante avec les variables binaires $y_{i,j}$, où $y_{i,j} = 1$ si i précède j , 0 sinon :

Trouver les variables $y_{i,j}$ t.q. (1)

$$\sum_{i=1}^n \sum_{j=i+1}^n y_{i,j} = N \quad (2)$$

$$y_{j,k} + y_{k,l} + y_{i,j} \leq 2, \forall j \in \{1, 2, \dots, n\}, \forall k \in \{1, 2, \dots, n\}, \forall l \in \{1, 2, \dots, n\}, j \neq k \neq l \quad (3)$$

$$y_{i,j} + y_{j,i} = 1, \forall i \in \{1, 2, \dots, n\}, \forall j \in \{1, 2, \dots, n\}, \quad (4)$$

$$\sum_{i=1}^n p_i y_{i,j} + p_j \leq \tilde{d}_j, \forall j \in \{1, 2, \dots, n\} \quad (5)$$

Vérifier si une séquence réalisable existe à un niveau donné peut également se faire par l'utilisation d'un programme par contraintes. Plusieurs algorithmes sont proposés et testés.

3 Conclusion et perspectives

L'idée de chercher une solution à un niveau du treillis plutôt que dans le treillis complet nous permet d'aborder le problème sous un angle nouveau et devrait nous permettre d'avancer sur la question de la complexité du problème.

Références

- [1] J-C. Billaut, P. Lopez (2011). *Characterization of all rho-approximated sequences for some scheduling problems*, 16th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'2011), Toulouse, 5-9 septembre 2011.
- [2] J-C. Billaut, E. Hébrard, P. Lopez (2012). *Complete Characterization of Near-Optimal Sequences for the Two-Machine Flow Shop Scheduling Problem* Ninth International Conference on Integration of Artificial Intelligence and Operations Research Techniques in Constraint Programming (CPAIOR'12), Nantes, 28 mai – 1 juin 2012.
- [3] T.T.T. Ta, K. Ringard, J-C. Billaut (2017). *New objective functions based on jobs positions for single machine scheduling with deadlines*, 7th International Conference on Industrial Engineering and Systems Management (IESM 2017), Saarbrücken (Allemagne), Octobre 2017.
- [4] T.T.T. Ta (2018). *New single machine scheduling problems with deadlines for the characterization of optimal solutions*. Thèse de doctorat, Université de Tours, 2018.