Arbres de décision robustes pour l'ordonnancement proactif/réactif sous incertitude

Tom Portoleau[1,2], Christian Artigues[1], Romain Guillaume[2]

[1] LAAS-CNRS, Université de Toulouse, CNRS, France
{tom.portoleau,christian.artigues}@laas.fr

[2] IRIT-CNRS, Université de Toulouse, France romain.guillaume@irit.fr

Mots-clés : *ordonnancement sous incertitude, proactif-réactif, arbre de décision*

1 Introduction

L'incertitude dans les problèmes d'ordonnancement est étudiée depuis maintenant plusieurs années, et plusieurs familles de méthodes ont vu le jour pour y faire face. Les méthodes pro-actives par exemple visent à élaborer hors ligne des ordonnancements censés être de bonne qualité en moyenne, lorsque des distributions de probabilités sont connues, ou dans un scénario pire dans le cadre de l'optimisation robuste cas sinon. Cependant, un défaut majeur des méthodes pro-actives est qu'elles produisent en général des solutions trop conservatrices, en effet des solutions trop robustes ont tendances à être de moins bonne qualité en pratique. C'est particulièrement le cas pour des problèmes présentant des incertitudes grandes et fréquentes. Dans ces cas, les méthodes réactives sont plus appropriées car elles ne mettent pas l'accent sur l'ordonnancement initial, mais plutôt sur la manière de le modifier en ligne. Elles se basent généralement sur des règles de priorités, leur donnant une politique de réordonnancement pour s'adapter au scénario en cours. En revanche, contrairement aux méthodes proactives, elles ne fournissent aucune garantie sur la qualité de la solution. Dans le but de produire des solutions équilibrées entre robustesse et performance, des méthodes hybrides ont été mises au point. Dans [1], les auteurs proposent une méthode proactive-réactive qui cherche à calculer une politique optimale, qui dans leur cas correspond à une solution initiale robuste et un ensemble de réactions possibles. Dans [2], les auteurs proposent une approche "Just In Case", qui consiste à calculer un ordonnancement contingent qui maximise la probabilité que le planning se termine sans perturbation, dans un contexte lié à la robotique. Nous nous intéressons à l'adaptation des concepts de planning contingents aux problèmes d'ordonnancement robuste. Nous considérons des problèmes d'ordonnancement dans lesquels plusieurs paramètres sont connus à chaque itération du problème, et d'autres incertains selon les scénarios. Nous proposons une méthode proactive-réactive dans laquelle nous supposons que le décideur a accès à de la connaissance à propos des paramètres incertains, qui lui permettrait de réduire l'incertitude et de changer l'ordonnancement. En pratique, l'accès à une information peut être coûteuse, et la modification du planning peut être pénalisante. En utilisant au mieux les informations disponibles à chaque moment de décision, nous construisons (hors ligne) un arbre de décision qui est utilisable à chaque répétition du problème. Nous avons choisi d'appliquer notre approche sur le problème $1||L_{max}$ dans lequel on suppose que les due dates sont incertaines.

2 Modèle

On se place dans un problème d'ordonnancement générique avec un fonction objectif f à minimiser. Nous modélisons les incertitudes par des intervalles, sans aucune hypothèse sur la distribution de probabilité : une donnée incertaine x prend sa valeur dans $X = [x_{min}, x_{max}]$. Si

$(x_i)_{i \in I}$ est une famille de données incertaines, on note $\Omega = \prod_{i \in I} X_i$ l'ensemble des réalisations possibles pour cette famille, et on appelle un scénario $\omega \in \Omega$ une réalisation donnée. On note $f(\omega, s)$ la valeur objectif d'une solution s dans le scenario ω. Étant donné un paramètre incertain x et un instant t, une information sur x au moment t est une valeur $k_x^t \in X$ et un opérateur dans $\{\leq, \geq\}$. Par exemple une information peut être de la forme $x \leq k_x^t$. Dans le cadre du problème $1||L_{max}$ notre hypothèse sur l'information est $k_{d_i}^t = \min(t + p_i, 2t)$ si $\min(t + p_i, 2t) \in D_i$, où D_i est le domaine d'incertitude sur la date de livraison d_i. Nous appelons un arbre de décision robuste un arbre dont les arcs sont étiquetés avec des solutions partielles, et un noeud n avec un ensemble de scénario Ω^n et une partition de ce sous ensemble notée P^n. L'arbre doit vérifier les propriétés suivantes. (i) Un noeud n a exactement $|P^n|$ fils. (ii) Si on note $(n_j)_{j \leq |P^n|}$ les fils de n, alors pour tout $j \leq |P_n|$, $\Omega^{n_j} \in P^n$ et $\bigcup_{j \leq |P_n|} \Omega^{n_j} = \Omega^n$. (iii) Pour tout chemin $(n_0, ..., n_m)$ où n_0 est la racine T et n_m une feuille, l'ordonnancement obtenu en concaténant toutes les solutions partielles sur le chemin est réalisable. (iv) Soient n et n' deux noeuds de T. La solution partielle s' sur l'arc (n, n') est robuste : $s' = \text{argmin}_{s \in S} \max_{\omega \in \Omega^{n'}} f(\omega, s)$ où S est l'ensemble des solutions partielles admissibles. Notre méthode a pour but de construire un tel arbre, ce qui implique de calculer la meilleure partition de l'ensemble des scénarios à chaque noeud. Pour comparer deux partitions, nous introduisons un critère basé sur la valeur $\min\max$ de chacune des parties de la partition. D'après la définition que l'on a donné, une information permet, à un instant t, de couper en deux l'ensemble des scénarios (ceux pour lesquels $x \leq k_x^t$ et ceux pour lesquels $x > k_x^t$). Plus généralement m informations permettent de séparer un ensemble de scénarios en 2^m sous-ensembles. Nous introduisons le Robust Partitioning Problem (RPP). Il prend en entrée un ensemble de scénarios Ω, un ensemble d'information K^t et deux entiers Q et L. Une solution à ce problème est une partition P telle que $|P| \leq L$, les sous-ensembles constituant P doivent être des unions d'hyperrectangles dont les côtés doivent coïncider avec les informations de K^t, les boites constituant les éléments de P ne peuvent utiliser qu'au plus Q informations, et P doit minimiser $RS(P)$ pour l'ordre lexicographique.

3 Algorithmes, résultats expérimentaux et conclusion

Étant donnée une liste de moment de décision $(t_j)_{j \in J}$, nous construisons l'arbre de sorte que chaque étage corresponde à un moment de décision : la racine est le moment $t_0 = 0$, pour laquelle on construit une solution robuste partielle jusqu'au prochain noeud à t_1. En ce noeud on résout le RPP avec l'ensemble des informations disponibles à ce moment, puis pour chaque sous-ensemble constituant la partition solution, on calcule la solution robuste partielle jusqu'au prochain noeud de décision, créant un nouveau noeud à l'étage suivant, et on réitère, jusqu'au dernier moment de décision. Les paramètres Q et L sont donnés par le décideur et permettent de contrôler la taille de l'arbre et son temps de calcul. Pour résoudre le RPP, nous présentons un algorithme exact de complexité exponentielle basé sur une recherche exhaustive. Nous démontrons qu'une partie de notre algorithme peut être résolue en temps polynomial si f vérifie une certaine propriété. Nous avons mené des expérimentations afin d'évaluer la robustesse et la stabilités des solutions proposées par l'arbre, et également étudié la qualité des informations sélectionnées en comparant ces résultats avec d'autres algorithmes proactifs-réactifs. Les résultats sont très encourageants et montrent que cet arbre de décision robuste pourrait être utilisé dans un cadre industriel sur des problèmes plus complexes.

Références

[1] Morteza Davari and Erik Demeulemeester. Important classes of reactions for the proactive and reactive resource-constrained project scheduling problem. *Annals of Operations Research*, 274(1-2) :187–210, 2019.

[2] Mark Drummond, John Bresina, and Keith Swanson. Just-in-case scheduling. In *AAAI*, volume 94, pages 1098–1104, 1994.