

Projet de recherche

Maher MALLEM

Mots-clés : algorithmique, complexité paramétrée, ordonnancement, théorie des graphes.

Ce projet de recherche vise à renforcer et valoriser les ponts entre la recherche opérationnelle et la théorie des graphes, en particulier dans le cadre de l’ordonnancement et de la complexité paramétrée.

1 Introduction

Historiquement, la théorie des graphes et la recherche opérationnelle ont toutes deux été des fers de lance dans l’étude algorithmique et de complexité des problèmes informatiques. En témoigne, par exemple, la liste historique des vingt-et-un problèmes NP-complets de Karp [24]. Cependant, jusqu’à présent, la théorie de la complexité paramétrée s’est bien plus reposée sur la première que la seconde. Établie à la fin des années 1980 et visant à identifier plus finement les propriétés responsables de la difficulté des problèmes, cette théorie a très vite été popularisée dans le cadre de la théorie des graphes, et a depuis connu un essor important par ce biais [10]. À l’inverse, malgré quelques résultats ici et là (e.g., [3, 23]), son étude plus systématique en recherche opérationnelle, en ordonnancement par exemple, est bien plus récente [39, 40]. Il en résulte, entre ces deux communautés, un écart notable sur l’avancement des techniques algorithmiques et de bornes inférieures employées, ainsi que sur la popularisation de celles-ci.

Pourtant, à la lumière de leur succès en théorie des graphes, l’utilisation de telles techniques en recherche opérationnelle s’avère prometteuse. Cette dernière offre notamment une myriade de problèmes redoutables à résoudre et pouvant bénéficier d’une analyse plus fine de complexité. Par exemple, Nederlof et Swennenhuis ont récemment considéré une sous-branche de la complexité paramétrée, appelée *fine-grained complexity*, pour étudier pas moins d’une quarantaine de problèmes d’ordonnancement, avec comme paramètre le nombre minimum de tâches devant être ordonnancées [42]. L’année dernière, les mêmes auteurs ont pu user de cette expertise pour proposer le premier algorithme en temps sous-exponentiel [43] pour un problème central en ordonnancement ; problème dont on se demande s’il est dans P ou NP-difficile depuis les années 1970 [30], et pour lequel la précédente avancée algorithmique majeure de résolution exacte datait de 1963 [19, 20].

Suite à une mise en contexte centrée sur l’ordonnancement, ce projet détaille plusieurs cadres dans lesquels cette synergie entre recherche opérationnelle, théorie des graphes et complexité paramétrée pourra s’opérer. Bien que la plupart des exemples cités dans ce document concernent l’ordonnancement (de par mon parcours de recherche jusqu’à présent), la démarche pourra concerner tous les champs de la recherche opérationnelle.

2 Contexte

En guise d’illustration, nous situons les productions paramétrées récentes au sujet de l’ordonnancement. Ces dernières peuvent se décliner en deux approches générales : l’élaboration de schémas énumératifs efficaces basés sur des paramètres dits *structurels* (i.e., dégageant des régularités ou des contraintes permettant d’accélérer l’énumération), et la traduction en programmes entiers mixtes (MIP) avec peu de variables entières.

La première approche permet une proximité conceptuelle avec la théorie des graphes. En effet, en plus de ma thèse où cette approche était centrale [32], des paramètres usuels de graphes tels que le vertex cover number [2], le pathwidth [12] et la largeur de graphe orienté acyclique [49] sont à la base d’algorithmes paramétrés récents pour l’ordonnancement.

Quant à la seconde approche, elle s’avère particulièrement adaptée pour les instances dites à *haute multiplicité* - i.e., avec beaucoup de redondance dans les propriétés des éléments. Comme on peut s’y attendre, les paramètres utilisés visent à contraindre la multiplicité des propriétés des éléments, par exemple le nombre de dates butoirs distinctes pour un ensemble de tâches. Trois échelles de résultats

semblent se dessiner dans la littérature : la simple preuve d’existence d’algorithmes paramétrés à l’aide de traductions en MIP linéaires (MILP [31]) ou convexes (MICP [11], voir par exemple [21, 25]) ; à l’inverse, l’établissement de bornes inférieures et d’algorithmes de complexité (quasi-)coïncidente pour des problèmes précis [22] ; en entre-deux, la mise au point de méthodes de résolution optimisées pour des formes particulières de MILP appelées *N-fold* [27].

Enfin, bien que la seconde approche est à première vue plus éloignée de la théorie des graphes que la première, il est à noter que les résultats algorithmiques récents sur les programmes entiers *N-fold* reposent sur un paramètre de graphes [28]. En l’occurrence, il s’agit du minimum des treedepths des graphes de voisinage de la matrice de coefficients, l’un des variables entre elles, l’autre des contraintes entre elles, où deux variables (resp. contraintes) ont une arête entre elles si et seulement si elles ont un coefficient non nul dans une même contrainte (resp. une même variable a un coefficient non nul dans ces deux contraintes).

3 Sur le court terme

Les pistes mentionnées dans cette section visent à valoriser les résultats existants.

3.1 Implémentation

Bien qu’encore peu courante, l’implémentation des algorithmes paramétrés et leur confrontation à l’état de l’art sont primordiales à leur adoption par la communauté de la recherche opérationnelle. Pour l’exemple, de Weerdt et al. [12] avaient ciblé un problème d’ordonnancement avec un benchmark existant [7] pour implémenter leur algorithme paramétré et valider sa pertinence par rapport à l’état de l’art de l’époque, basé sur le *branch-and-price* [45]. D’ailleurs, au-delà de la quête de validation des approches, l’implémentation est une opportunité de questionnement et de redéfinition de celles-ci. En témoignent les difficultés rencontrées par Altmanova et al. [1] lors de l’implémentation de leur méthode de résolution des programmes entiers *N-fold*, et qui ont abouties à une avancée notable pour cette dernière, aussi bien en temps d’exécution effectif qu’en complexité asymptotique. Des méthodes hybrides pourront également être envisagées, par exemple en combinant règles de décomposition, heuristiques et/ou un modèle d’apprentissage automatique (voir par exemple [4]).

Cette démarche est entreprise dans mon post-doctorat actuel, où les algorithmes paramétrés proposés [38] sont en cours d’implémentation en C. À l’avenir, les challenges ROADEF et PACE pourront servir de terrains additionnels d’expérimentation.

3.2 Extension du champ d’applications

L’identification de problèmes de recherche opérationnelle qui pourraient bénéficier de la complexité paramétrée est à explorer. Probablement du fait de leur proximité avec la théorie des graphes, les graphes temporels ont déjà fait l’objet d’analyses de complexité paramétrée, par exemple pour la recherche de plus court chemin sans temps d’arrêt [5, 6]. À l’inverse, à ma connaissance, l’approche paramétrée a été encore peu explorée pour les problèmes de logistique (e.g., *lot-sizing*) ou de transport (e.g., tournées de véhicules). Pourtant, les méthodes paramétrées à base de MIP valent sans doute un détour.

Quant aux algorithmes paramétrés énumératifs, à l’image des travaux de de Weerdt et al. [12], il est probable que des modifications mineures permettent de les adapter à des critères d’optimisation plus sophistiqués que la minimisation du makespan (e.g., ma thèse [32]) ou le nombre pondéré de tâches en retard (e.g., [21, 25]). On pourra également s’inspirer du travail d’Hanen et Munier-Kordon [18], qui ont généralisé un algorithme d’énumération paramétré [41] afin de considérer un environnement de machines plus complet et plus à même de modéliser des problèmes réels.

3.3 Choix de paramètres

L’identification de paramètres pertinents pour les problèmes de recherche opérationnelle est loin d’être évidente. Par exemple, pour l’ordonnancement sur une machine, majorer le nombre de fenêtres de

temps concurrentes suffit à rendre le problème *fixed-parameter tractable* (voir la section 4.2. de ma thèse [32]). Aucune contrainte sur les durées des tâches n’est nécessaire, ce qui est surprenant au vu de leur centralité dans la preuve historique de NP-complétude du problème [16]. Ce résultat a même pu être amélioré lors de ma thèse [34], avec une nouvelle idée de paramètre issue du croisement entre des règles de dominance pré-existantes en ordonnancement [14] et un travail de classification des graphes d’intervalles [44].

Cependant, bien que toujours plus petit que le paramètre précédent, cela ne garantit pas une faible valeur de ce nouveau paramètre dans les instances réelles. Les choix de paramètres doivent donc être conditionnés par une analyse statistique de leur valeur, à l’image du travail de de Weerdt et al. [12], qui reposait sur le fait que la valeur de leur paramètre principal dépassait rarement 7 pour l’ordonnancement d’observations satellitaires.

3.4 Kernelisation

La kernelisation vise à proposer des algorithmes de compression en prétraitement, i.e., réduisant l’entrée à une instance équivalente de plus petite taille, idéalement majorée par une fonction du paramètre. Cette approche est complémentaire des algorithmes paramétrés de résolution exacte, et ce particulièrement dans le cadre de problèmes de recherche opérationnelle, pour réduire le nombre d’éléments avant le lancement de procédures coûteuses. On pourra s’inspirer de travaux existants sur le problème du sac à dos et ses variantes [15, 17], l’ordonnancement via programmes entiers N -fold [26], et l’ordonnancement sur une machine de tâches couplées [2], ou admettant des fenêtres de temps tel que dans l’une de mes soumissions récentes [36].

4 Sur le long terme

Les pistes mentionnées dans cette section visent à acquérir une compréhension plus fondamentale de la complexité des problèmes de recherche opérationnelle. Cet éclairage est voué à guider les éventuels choix à prendre dans le cadre des axes présentés dans la section précédente.

4.1 Frontières de complexité

Suite au travail effectué dans ma thèse pour l’établissement des frontières de complexité paramétrée de plusieurs problèmes d’ordonnancement [32], il s’agit de poursuivre cet effort pour les problèmes de recherche opérationnelle de manière générale. La complexité paramétrée étant une analyse bidimensionnelle (problème, paramètre) de complexité, en plus de déterminer les problèmes les plus généraux qui sont *fixed-parameter tractable* pour un paramètre donné, il s’agit d’établir les paramètres minimaux pour lesquels un problème donné est *fixed-parameter tractable*. Par exemple, une publication [33] et deux soumissions [35, 37] basées sur mes travaux de thèse approfondissent l’impact de l’inclusion de délais de précédence dans des problèmes d’ordonnancement avec fenêtres de temps. En plus de publications classiques, la diffusion de ces résultats pourra prendre la forme de cartes de paramètres, à l’image du travail fait en théorie des graphes pour les graphes non orientés [46, 48], ainsi que de contributions à la base de données du Scheduling Zoo [13].

4.2 Complexité *fine-grained*

Lorsqu’un problème paramétré est identifié comme *fixed-parameter tractable*, estimer la dépendance minimale possible en le paramètre devient un enjeu majeur, afin d’évaluer au mieux le potentiel applicatif de ce paramètre. Pour cela, en plus du réinvestissement de techniques algorithmiques avancées issues de la théorie des graphes, il s’agit d’établir des minorants de temps d’exécution reposant sur des hypothèses de complexité classiques, telles que l’*Exponential Time Hypothesis* (ETH) pour la tendance asymptotique, ou sa version forte SETH pour la minoration de la base des facteurs exponentiels en temps par exemple [10]. De telles démarches ont été récemment menées pour des problèmes d’ordonnancement, mais restent encore rares [8, 22, 42, 47]. En parallèle, il sera primordial de se maintenir à jour sur la pertinence et la plausibilité des hypothèses de complexité employées (voir par exemple [29]).

4.3 Méta-théorèmes

Enfin, dans la veine du théorème de Courcelle [9], l'établissement de méta-théorèmes de complexité pour les problèmes de recherche opérationnelle serait une avancée majeure pour élargir le champ d'actions des méthodes paramétrées ; notamment, dans le cadre de ce projet, que les algorithmes paramétrés pour l'ordonnancement puissent être réinvestis dans d'autres domaines de la recherche opérationnelle. Un défi de taille réside dans la définition d'une logique, à l'image de celle monadique du second ordre pour le théorème de Courcelle, qui engloberait un maximum de ces problèmes de recherche opérationnelle, en dépit de l'hétérogénéité de leurs définitions. Dans le cadre des problèmes à haute multiplicité, cette difficulté a été surmontée avec succès par le biais des programmes entiers N -fold (voir la section 1 de [27] pour un résumé de l'approche). Les retombées sont nombreuses et encore largement inexplorées en pratique, notamment en packing et en ordonnancement pour une large famille de critères d'optimisation (voir la section 5 de [27]). Ainsi, cet axe vise prioritairement l'établissement de résultats similaires pour des paramètres structuraux.

Références

- [1] K. Altmanová, D. Knop, and M. Koutecký. Evaluating and tuning n -fold integer programming. *ACM J. Exp. Algorithmics*, 24, July 2019.
- [2] S. Bessy and R. Giroudeau. Parameterized complexity of a coupled-task scheduling problem. *Journal of Scheduling*, 22(3) :305–313, June 2019.
- [3] H. L. Bodlaender and M. R. Fellows. $W[2]$ -hardness of precedence constrained k -processor scheduling. *Operations Research Letters*, 18(2) :93–97, 1995.
- [4] M. Bouška, P. Šůcha, A. Novák, and Z. Hanzálek. Deep learning-driven scheduling algorithm for a single machine problem minimizing the total tardiness. *European Journal of Operational Research*, 308(3) :990–1006, 2023.
- [5] A. Casteigts, A. Himmel, H. Molter, and P. Zschoche. Finding temporal paths under waiting time constraints. *Algorithmica*, 83(9) :2754–2802, 2021.
- [6] J. Cauvi and L. Viennot. Parameterized restless temporal path. In A. Jez and J. Otop, editors, *Fundamentals of Computation Theory*, pages 82–93, Cham, 2026. Springer Nature Switzerland.
- [7] B. Cesaret, C. Oğuz, and F. Sibel Salman. A tabu search algorithm for order acceptance and scheduling. *Computers and Operations Research*, 39(6) :1197–1205, 2012. Special Issue on Scheduling in Manufacturing Systems.
- [8] L. Chen, K. Jansen, and G. Zhang. On the optimality of exact and approximation algorithms for scheduling problems. *Journal of Computer and System Sciences*, 96 :1–32, 2018.
- [9] B. Courcelle. The monadic second-order logic of graphs. i. recognizable sets of finite graphs. *Information and Computation*, 85(1) :12–75, 1990.
- [10] M. Cygan, F. V. Fomin, Ł. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*, volume 5. Springer, 2015.
- [11] D. Dadush, C. Peikert, and S. Vempala. Enumerative lattice algorithms in any norm via m-ellipsoid coverings. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 580–589. IEEE, 2011.
- [12] M. de Weerd, R. Baart, and L. He. Single-machine scheduling with release times, deadlines, setup times, and rejection. *European Journal of Operational Research*, 291(2) :629–639, 2021.
- [13] C. Dürr. The Scheduling Zoo, 2023. URL : <https://github.com/xtodurr/schedulingzoo/wiki/The-Scheduling-Zoo-project>.
- [14] J. Erschler, G. Fontan, and C. Merce. Un nouveau concept de dominance pour l'ordonnancement de travaux sur une machine. *RAIRO-Operations Research*, 19(1) :15–26, 1985.
- [15] M. Etscheid, S. Kratsch, M. Mnich, and H. Röglin. Polynomial kernels for weighted problems. *Journal of Computer and System Sciences*, 84 :1–10, 2017.

- [16] M. R. Garey and D. S. Johnson. *Computers and Intractability*, volume 174. Freeman San Francisco, 1979.
- [17] F. Gurski, C. Rehs, and J. Rethmann. Knapsack problems : a parameterized point of view. *Theoretical Computer Science*, 775 :93–108, 2019.
- [18] C. Hanen and A. Munier Kordon. Fixed-parameter tractability of scheduling dependent typed tasks subject to release times and deadlines. *Journal of Scheduling*, pages 1–15, 2023.
- [19] M. Held and R. M. Karp. A dynamic programming approach to sequencing problems. *Journal of the Society for Industrial and Applied Mathematics*, 10(1) :196–210, 1962.
- [20] M. Held, R. M. Karp, and R. Shareshian. Assembly-line balancing—dynamic programming with precedence constraints. *Operations Research*, 11(3) :442–459, 1963.
- [21] D. Hermelin, S. Karhi, M. Pinedo, and D. Shabtay. New algorithms for minimizing the weighted number of tardy jobs on a single machine. *Annals of Operations Research*, 298(1) :271–287, 2021. Number : 1 Publisher : Springer.
- [22] K. Jansen, K. Kahler, and E. Zwanger. Exact and approximate high-multiplicity scheduling on identical machines. In I. Finocchi and L. Georgiadis, editors, *Algorithms and Complexity - 14th International Conference, CIAC 2025, Rome, Italy, June 10-12, 2025, Proceedings, Part I*, volume 15679 of *Lecture Notes in Computer Science*, pages 1–17. Springer, 2025.
- [23] K. Jansen, S. Kratsch, D. Marx, and I. Schlotter. Bin packing with fixed number of bins revisited. *Journal of Computer and System Sciences*, 79(1) :39–49, 2013.
- [24] R. M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103. Springer, 1972.
- [25] M. Kaul, M. Mnich, and H. Molter. Single-Machine Scheduling to Minimize the Number of Tardy Jobs with Release Dates. In E. Bonnet and P. Rzażewski, editors, *19th International Symposium on Parameterized and Exact Computation (IPEC 2024)*, volume 321 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 19 :1–19 :15, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [26] D. Knop and M. Koutecký. Scheduling kernels via configuration LP. In *30th Annual European Symposium on Algorithms (ESA 2022)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.
- [27] D. Knop, M. Koutecký, A. Levin, M. Mnich, and S. Onn. Multitype integer monoid optimization and applications. *arXiv :1909.07326 [cs, math]*, Sept. 2019. arXiv : 1909.07326.
- [28] M. Koutecký, A. Levin, and S. Onn. A Parameterized Strongly Polynomial Algorithm for Block Structured Integer Programs. In I. Chatzigiannakis, C. Kaklamanis, D. Marx, and D. Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, volume 107 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 85 :1–85 :14, Dagstuhl, Germany, 2018. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [29] M. Lampis. *The Primal Pathwidth SETH*, pages 1494–1564.
- [30] J. Lenstra and A. Rinnooy Kan. Complexity of scheduling under precedence constraints. *Oper. Res.*, 26(1) :22–35, 1978.
- [31] H. W. Lenstra Jr. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8(4) :538–548, 1983.
- [32] M. Mallem. *Parameterized Complexity and new Efficient Enumerative Schemes for RCPSP*. PhD thesis, Sorbonne Université, 2024.
- [33] M. Mallem, C. Hanen, and A. M. Kordon. Parameterized complexity of a parallel machine scheduling problem. In H. Dell and J. Nederlof, editors, *17th International Symposium on Parameterized and Exact Computation, IPEC 2022, Potsdam, Germany, September 7-9, 2022*, volume 249 of *LIPIcs*, pages 21 :1–21 :21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.
- [34] M. Mallem, C. Hanen, and A. M. Kordon. A new structural parameter on single machine scheduling with release dates and deadlines. In A. Basu, A. R. Mahjoub, and J. J. S. González, editors, *Combinatorial Optimization - 8th International Symposium, ISCO 2024, La Laguna, Tenerife*,

Spain, May 22-24, 2024, Revised Selected Papers, volume 14594 of *Lecture Notes in Computer Science*, pages 205–219. Springer, 2024.

- [35] M. Mallem, C. Hanen, and A. Munier-Kordon. Single machine scheduling with precedence constraints and bounded maximum delay value. *Journal on Combinatorial Optimization*, Submitted in August 2025.
- [36] M. Mallem, C. Hanen, and A. Munier-Kordon. Kernelization algorithms on single machine scheduling with time windows. *INFORMS Journal on Computing*, Submitted in November 2024.
- [37] M. Mallem, C. Hanen, and A. Munier-Kordon. Coupled-task scheduling with time windows, bounded pathwidth and bounded slack is para-np-complete. *Theoretical Computer Science*, Submitted in October 2025.
- [38] M. Mallem and F. Vivien. Notes on scheduling radio-astronomical observations. In *20th International Workshop on Project Management and Scheduling*, 2026. Submitted in December 2025.
- [39] M. Mnich and R. van Bevern. Parameterized complexity of machine scheduling : 15 open problems. *Computers & Operations Research*, 100 :254–261, Dec. 2018.
- [40] M. Mnich and A. Wiese. Scheduling and fixed-parameter tractability. *Mathematical Programming*, 154(1-2) :533–562, Dec. 2015.
- [41] A. Munier-Kordon. A fixed-parameter algorithm for scheduling unit dependent tasks on parallel machines with time windows. *Discrete Applied Mathematics*, 290 :1–6, 2021.
- [42] J. Nederlof and C. M. F. Swennenhuis. On the fine-grained parameterized complexity of partial scheduling to minimize the makespan. *Algorithmica*, 84(8) :2309–2334, 2022.
- [43] J. Nederlof, C. M. F. Swennenhuis, and K. Wegrzycki. A subexponential time algorithm for makespan scheduling of unit jobs with precedence constraints. In Y. Azar and D. Panigrahi, editors, *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2025, New Orleans, LA, USA, January 12-15, 2025*, pages 535–552. SIAM, 2025.
- [44] A. Proskurowski and J. A. Telle. Classes of graphs with restricted interval models. *Discrete Mathematics & Theoretical Computer Science*, 3, 1999.
- [45] Y. L. T. Silva, A. Subramanian, and A. A. Pessoa. Exact and heuristic algorithms for order acceptance and scheduling with sequence-dependent setup times. *Computers and Operations Research*, 90 :142–160, 2018.
- [46] M. Sorge and M. Weller. The graph parameter hierarchy. *Unpublished manuscript*, pages 1–10, 2019. URL : <https://manyu.pro/assets/parameter-hierarchy.pdf>.
- [47] C. M. F. Swennenhuis. Fine-grained parameterized complexity of scheduling and sequencing problems. 2022.
- [48] D. L. Tran. Expanding the graph parameter hierarchy. Bachelor’s thesis, Institute of Software, 2022.
- [49] R. van Bevern, R. Brederick, L. Bulteau, C. Komusiewicz, N. Talmon, and G. J. Woeginger. Precedence-constrained scheduling problems parameterized by partial order width. In Y. Kochevov, M. Khachay, V. Beresnev, E. Nurminski, and P. Pardalos, editors, *Discrete Optimization and Operations Research*, pages 105–120, Cham, 2016. Springer International Publishing.