Nantes Université

École Doctorale MaSTIC

Département d'Informatique
UFR Sciences et Techniques

Quelques contributions à la résolution de problèmes d'optimisation en présence d'incertitude, de modularité ou de contrôlabilité

THÈSE

présentée et soutenue publiquement le 25 mars 2025

pour l'obtention d'une

Habilitation à Diriger des Recherches de l'Université de Nantes

(Spécialité Informatique)

par

Evgeny Gurevsky

Composition du jury

Présidente :	Nathalie Bostel	Professeur, Université de Nantes
Rapporteurs :	Christian Artigues	Directeur de Recherche, LAAS, CNRS, Toulouse
	Jean-Charles Billaut	Professeur, Université de Tours
	Safia Kedad-Sidhoum	Professeur, CNAM, Paris
Examinateurs :	Olga Battaïa	Professeur, Kedge Business School, Talence
	Nadjib Brahimi	Professeur, Rennes School of Business
	André Rossi	Professeur, Université Paris-Dauphine
Garant :	Alexandre Dolgui	Professeur, IMT Atlantique, Nantes

Laboratoire des Sciences du Numérique de Nantes (UMR CNRS 6004)

LS2N
Laboratoire des Sciences du Numérique de Nantes

Remerciements

Tout d'abord, je souhaite exprimer ma profonde gratitude à Mr. Alexandre Dolgui, le garant de cette HDR et Professeur à l'IMT Atlantique (Nantes), pour ses précieux conseils et son soutien constant tout au long de la préparation de ce manuscrit.

Je remercie également très chaleureusement l'ensemble des membres du jury qui m'ont fait l'honneur d'évaluer ce travail. J'adresse en particulier mes sincères remerciements à Mme. Nathalie Bostel, Professeur à l'Université de Nantes, pour avoir accepté de présider le jury de cette HDR. Ensuite, je tiens à exprimer toute ma reconnaissance à Mme. Safia Kedad-Sidhoum, Professeur au CNAM (Paris), à Mr. Christian Artigues, Directeur de Recherche CNRS au LAAS (Toulouse), et à Mr. Jean-Charles Billaut, Professeur à l'Université de Tours, pour leur lecture attentive du manuscrit et leur engagement en tant que rapporteurs. Je remercie également Mme. Olga Battaïa, Professeur à Kedge Business School (Talence), Mr. Nadjib Brahimi, Professeur à Rennes School of Business, et Mr. André Rossi, Professeur à l'Université Paris-Dauphine, pour l'honneur qu'ils m'ont fait en acceptant d'examiner mon travail.

Enfin, je tiens à adresser un grand merci à l'ensemble de mes collègues du laboratoire LS2N (UMR CNRS 6004) et plus particulièrement de l'équipe MODELIS, ainsi qu'à tous les membres du Département d'Informatique de l'UFR Sciences et Techniques. Leur amitié, leur appui et leur disponibilité au fil des années – depuis mon recrutement à l'Université de Nantes il y a maintenant douze ans – ont été pour moi d'une valeur inestimable.

Je dédie ce manuscrit
d'HDR à ma famille.

Table des matières

Introduction générale **7**

1 Équilibrage robuste de lignes de production avec tâches de durée incertaine **9**

1.1 Introduction . . . 9
1.2 Notations et définitions de base . . . 12
1.3 Calcul des rayons de stabilité . . . 14
1.4 Formulation du problème . . . 15
1.5 Pré-traitement . . . 17
1.5.1 Intervalles d'affectation et blocs vides . . . 17
1.5.2 Heuristique . . . 19
1.5.3 Approche globale de pré-traitement . . . 19
1.6 Résultats numériques . . . 20
1.7 Conclusion et perspectives . . . 22

2 Équilibrage de lignes de production reconfigurables avec équipement modulaire **23**

2.1 Introduction . . . 23
2.2 Revue de la littérature . . . 25
2.3 Formulation PLNB compacte . . . 28
2.4 Reformulation du modèle PLNB compact . . . 30
2.4.1 Algorithme de génération de modules . . . 32
2.4.2 Génération de contraintes . . . 33
2.5 Résultats numériques . . . 33
2.5.1 Comparaison des formulations . . . 34
2.6 Conclusion et perspectives . . . 36

3 Problèmes de gestion du risque avec données contrôlables **39**

3.1 Introduction . . . 39
3.2 Réduction à une série de problèmes plus simples . . . 41
3.3 Quelques cas polynomiaux . . . 42
3.3.1 Problème Min-Risk-Span . . . 42
3.3.2 Problème Min-Risk-Path . . . 43
3.4 Problèmes reverses . . . 43
3.5 Conclusion et perspectives . . . 45

Conclusion générale et perspectives **47**

A Curriculum vitæ **51**

A.1 Éducation 51
A.2 Parcours académique 51
A.3 Thèmes de recherche et production scientifique 52
A.4 Encadrement doctoral et scientifique 62
A.5 Participation à des projets de recherche 64
A.6 Animation scientifique 66
A.7 Activités d'enseignement 68
A.8 Responsabilités collectives et d'intérêt général 69

Bibliographie **71**

Introduction générale

Ce mémoire porte sur mes activités de recherche ainsi que sur l'encadrement scientifique que j'ai pu assurer depuis mon recrutement à l'Université de Nantes en 2012. Au lieu de chercher à réaliser une synthèse à caractère chronologique de mes travaux au cours des 12 dernières années, j'ai fait le choix de mettre en avant les contributions les plus représentatives de mes directions scientifiques.

Ainsi, ce mémoire se compose de la présente introduction générale, de trois chapitres de contributions scientifiques, d'une conclusion générale et de perspectives, et enfin d'une annexe présentant mon CV.

Le chapitre 1 est dédié à la description de l'un des résultats obtenus dans le cadre de la thèse d'Aleksandr Pirogov (2016 - 2019), le tout premier doctorant que j'ai pu co-encadrer avec Alexandre Dolgui (IMT Atlantique, Nantes) et André Rossi (Université Paris-Dauphine). Ce résultat concerne une approche de robustesse alternative appliquée à l'équilibrage de lignes de transfert avec des tâches de durée incertaine, que nous popularisons depuis la parution de l'article Rossi et al. (2016), également co-écrit avec Olga Battaïa (Kedge Business School, Talence), Alexandre Dolgui et moi-même. L'approche de robustesse étudiée repose sur le concept de rayon de stabilité, défini en détail dans le chapitre 1, qui a l'avantage de ne pas nécessiter d'historique de variabilité de la durée des tâches pour être mis en œuvre. L'idée est donc de trouver la configuration de ligne réalisable la plus résistante aux aléas, c'est-à-dire celle dont la valeur du rayon de stabilité est la plus élevée possible. Il s'agit en fait d'un nouveau problème d'optimisation combinatoire NP-difficile, introduit dans Rossi et al. (2016) pour le cas de lignes d'assemblage, qui a ouvert de nombreuses perspectives algorithmiques et pratiques. Ainsi, avec Aleksandr Pirogov, nous avons tout d'abord étendu les résultats obtenus dans Rossi et al. (2016) pour le cas de lignes de transfert en développant une formulation de programmation linéaire en nombres mixtes (PLNM) efficace pour chaque norme l_1 et l_∞ dans lesquelles le rayon de stabilité peut être calculé et interprété. De plus, plusieurs techniques puissantes de pré-traitement ont également été proposées, permettant ainsi de réduire considérablement l'espace de recherche et de résoudre plus efficacement le problème soulevé pour les deux normes, sur la base de la génération de coupes spécifiques fondées sur le principe d'optimalité. À la fin de ce chapitre, d'autres résultats de la thèse d'Aleksandr Pirogov sont aussi mentionnés. En outre, une contribution importante issue de la collaboration avec Xavier Schepler, à l'époque post-doctrorant, et que j'ai eu l'occasion de co-superviser de 2016 à 2017 dans le cadre d'un projet similaire, est également évoquée. Le chapitre se termine par un bilan pour cette première direction de recherche et donne quelques perspectives qui sont déjà à l'étude dans le cadre d'une nouvelle collaboration initiée avec Rui S. Shibasaki (Université de Picardie Jules Verne, Amiens).

Le chapitre 2, quant à lui, se focalise sur l'une des contributions réalisées dans le cadre de la thèse d'Abdelkrim R. Yelles-Chaouche (2018 - 2021), le deuxième doctorant que j'ai eu l'occasion de co-encadrer avec Nadjib Brahimi (Rennes School of Business) et Alexandre Dolgui. Cette contribution traite de l'équilibrage de lignes de production reconfigurables, où les tâches sont exécutées successivement au sein de l'équipement modulaire (ci-après modules) qui constitue les machines de la ligne. Pour une famille de produits donnée, l'objectif est donc de déterminer une configuration de ligne réalisable pour chaque produit, tout en minimisant le nombre total de modules différents requis à la composition de ces configurations. Ainsi, pour pouvoir traiter efficacement ce problème complexe, avec Abdelkrim R. Yelles-Chaouche, nous l'avons décomposé en deux sous-problèmes. Le premier consiste à générer intelligemment l'ensemble des modules réalisables à partir desquels la ligne peut être constituée. Le seconde est un programme linéaire en nombres binaires (PLNB) basé sur cet ensemble de modules pré-générés. Afin de résoudre exactement ce PLNB, une technique permettant de générer des contraintes dites «lentes» a également été mise en œuvre. De plus, pour aborder les instances de taille moyenne/grande de ce problème de manière plus ra-

pide et plus efficace, une méthode heuristique basée sur ce PLNB ainsi que sur une technique de filtrage des modules a aussi été proposée. Ce chapitre se conclut par quelques perspectives sur cette contribution et d'autres résultats de la thèse d'Abdelkrim R. Yelles-Chaouche.

En ce qui concerne le chapitre 3, il est le résultat d'une collaboration fructueuse, débutée en 2015, avec Dmitry Kopelevich (UIIP, NASB, Minsk, Biélorussie), Sergey Kovalev (INSEEC Business School, Lyon) et Mikhail Y. Kovalyov (UIIP, NASB, Minsk, Biélorussie), suite à la première visite de courte durée de Mikhail Y. Kovalyov à l'Université de Nantes dans le cadre du programme «Mission Chercheurs Invités», piloté par l'UFR Sciences et Techniques. Il s'agit de l'étude de la complexité algorithmique pour une nouvelle classe de problèmes d'optimisation combinatoire avec des données contrôlables et des fonctions de risque. D'une manière générale, ces problèmes, introduits dans les travaux de Chen et al. (2009a,b), peuvent être vus comme ceux où les variables de décision bornées représentent une certaine valeur de consommation de ressources limitées (comme, par exemple, le budget, le temps, l'espace, etc.), impliquant ainsi le niveau de risque de défaillance ou bien d'accident qu'on vise à minimiser. Ces problèmes trouvent des applications directes intéressantes dans des domaines tels que le routage ou la conception de réseaux fiables. Le chapitre 3 est consacré à une variante de ces problèmes, à savoir celle qui cherche à minimiser le risque total, notée ci-après Min-Risk. Au niveau des résultats, une approche de réduction originale a été développée, nous permettant de proposer des algorithmes polynomiaux significativement plus rapides et plus efficaces (en termes d'ordres de grandeur) que dans Chen et al. (2009a,b) non seulement pour les deux applications de routage et de réseau, mais aussi pour une classe plus large de fonctions de risque, à savoir les fonctions concaves non-croissantes. Les problèmes dits «reverses», notés Rev-Min-Risk, ont également été abordés. Pour ces problèmes, la contrainte de ressource et la fonction de risque sont interverties. Pour eux aussi, nous avons réussi à proposer des algorithmes polynomiaux plus efficaces et plus rapides que dans Álvarez-Miranda et al. (2011). Le chapitre 3 s'achève sur une conclusion pour cette troisième direction de recherche, en mentionnant également certains résultats obtenus pour une autre variante du problème, appelée Min-Max-Risk, qui cherche à minimiser le risque maximum. Enfin, un certain nombre de pistes de contributions futures sont exposées dans le contexte de la poursuite de cette collaboration.

Finalement, ce mémoire se termine par un bilan des contributions scientifiques réalisées au cours de ces 12 dernières années et par un projet de recherche pour les années à venir. Outre les directions de recherche présentées dans les trois premiers chapitres qui ont toutes, à mon avis, un degré de maturité suffisant avec de nombreuses perspectives, les travaux en cours de trois autres thèses de doctorat, que j'ai actuellement le plaisir de co-encadrer, sont également brièvement décrits.

Chapitre 1

Équilibrage robuste de lignes de production avec tâches de durée incertaine

Sommaire

1.1 Introduction **9**
1.2 Notations et définitions de base **12**
1.3 Calcul des rayons de stabilité **14**
1.4 Formulation du problème **15**
1.5 Pré-traitement **17**
1.5.1 Intervalles d'affectation et blocs vides 17
1.5.2 Heuristique 19
1.5.3 Approche globale de pré-traitement 19
1.6 Résultats numériques **20**
1.7 Conclusion et perspectives **22**

1.1 Introduction

Une ligne d'assemblage simple est un système de fabrication typique orienté vers le flux (voir, *e.g.*, Scholl, 1999), qui est utilisé pour fabriquer de grandes quantités d'un seul type de produit. Il peut être vu comme un ensemble de postes de travail ordonnés linéairement et reliés par un convoyeur à bande déplaçant les unités de produit. Au cours de la fabrication, les unités passent par les postes de travail dans l'ordre de leur disposition. Ainsi, elles sont introduites séquentiellement au début de la ligne, transférées d'un poste de travail à l'autre et sortent à la fin de la ligne. Les postes de travail opèrent simultanément. À chacun d'entre eux, son propre ensemble de tâches est exécuté de manière répétitive sur les unités successives. De plus, le fonctionnement des lignes d'assemblage simples possède également les caractéristiques naturelles suivantes. Une seule unité peut être traitée simultanément par le poste de travail et un seul poste de travail à la fois peut manipuler l'unité. Les tâches de chaque poste de travail sont exécutées séquentiellement, une par une, sans fractionnement. Il n'y a ni stock tampon ni poste de travail parallèle et, par conséquent, le transfert de toutes les unités situées sur la ligne est effectué de manière synchronisée, c'est-à-dire que toutes les unités sont déplacées simultanément de leur poste de travail actuel au poste suivant.

La conception de ces lignes est un problème majeur, car elle implique généralement des investissements importants. Cette étape comprend plusieurs problèmes à résoudre, dont l'un est appelé *problème d'équilibrage*. En général, il s'agit de répartir l'ensemble des tâches d'assemblage nécessaires entre les postes de travail de manière optimale en fonction d'un objectif de production donné. La plupart du temps, des

restrictions supplémentaires sont également prises en considération. Par exemple, certaines tâches ne sont généralement pas exécutées de manière arbitraire, mais sont soumises à des contraintes de précédence. La représentation de ces contraintes se fait souvent par un graphe acyclique orienté, où l'ensemble des nœuds correspond à l'ensemble des tâches et les arcs introduisent un ordre partiel sur celles-ci. Ainsi, un arc (i, j) signifie que la tâche j ne peut pas commencer avant que la tâche i ne soit terminée. Le cadencement du déplacement des unités demande que le temps de travail total (ou la charge) de chaque poste de travail ne soit pas supérieur à une certaine valeur fixée, qui détermine le taux de production de la ligne. Cette valeur est appelée *temps de cycle*[1] et la contrainte correspondante est appelée *contrainte de temps de cycle*. Enfin, les limitations de l'espace disponible pour l'assemblage peuvent être naturellement traduites en contraintes sur le nombre maximal de postes de travail à installer.

En ce qui concerne les objectifs utilisés, les problèmes d'équilibrage de lignes d'assemblage simples, connus sous l'abréviation SALBP[2] dans la littérature anglo-saxone, sont généralement classés selon les types suivants (voir, *e.g.*, Scholl, 1999; Battaïa and Dolgui, 2013) : minimiser le nombre de postes de travail utilisés pour un temps de cycle fixe (SALBP-1) ; minimiser le temps de cycle avec un nombre donné de postes de travail (SALBP-2) ; et si ni le nombre de postes de travail ni le temps de cycle de la ligne ne sont fixes, maximiser l'efficacité de la ligne (SALBP-E). Ce dernier problème vise à déterminer une configuration de ligne qui minimise l'expression suivante : le nombre de postes de travail utilisés multiplié par le temps de travail sur le poste le plus chargé. Pour ces problèmes, connus pour être NP-difficiles (voir Scholl, 1999, chapitre 2.2.1.5), un grand nombre de méthodes exactes et heuristiques ont été développées. Elles sont présentées en détail dans Rekiek et al. (2002), Scholl and Becker (2006) et Battaïa and Dolgui (2013).

Malgré toute l'attention accordée au SALBP, sa formulation classique reste assez générale et ne reflète pas toujours les situations réelles particulières de la fabrication. Souvent, des hypothèses plus spécifiques doivent être prises en compte. Ainsi, par exemple, l'un des sujets importants à considérer est la variabilité de la durée des tâches. En effet, comme mentionné dans Battaïa and Dolgui (2013), les durées des tâches ne sont souvent pas connues avec exactitude au stade de la conception préliminaire de la ligne et seules leurs valeurs nominales (ou estimées) sont utilisées. Cela est dû à des facteurs pratiques, comme ceux décrits ci-dessous :

- pour les lignes d'assemblage manuel, les performances des opérateurs, qui exécutent les tâches, dépendent de leur rythme de travail, de leur niveau de compétence, de leur fatigue et de leur motivation ;
- les spécifications du produit ainsi que les caractéristiques du poste de travail peuvent être modifiées au cours du cycle de vie de la ligne. Cela peut être dû à la demande d'un client ou à la mise à jour du marché des matériaux ;
- divers retards et micro-arrêts lors de l'exécution des tâches peuvent survenir.

Chacun de ces événements peut se produire à n'importe quel moment de l'exploitation de la ligne et peut entraîner une interruption coûteuse de la ligne si le temps de cycle est dépassé. En conséquence, pour construire une configuration de ligne robuste pour une utilisation à long terme, la variabilité de la durée des tâches doit être anticipée au stade de l'équilibrage de la ligne. Dans ce qui suit, nous présentons une vue d'ensemble des approches existantes consacrées à ces aspects.

Le choix d'une approche appropriée pour tenir compte la durée variable des tâches dépend fortement des informations disponibles concernant son incertitude. Ainsi, parmi les méthodes utilisées dans la littérature, nous pouvons distinguer les suivantes : *stochastique*, *floue* et *robuste*.

Pour l'*approche stochastique*, les durées des tâches sont généralement représentées comme des variables aléatoires indépendantes avec des distributions de probabilité connues. Par conséquent, pour regrouper les tâches en postes de travail, une technique particulière supervisant la contrainte de temps de cycle doit être utilisée. Parmi les travaux qui traitent de ce sujet, ceux qui appliquent la méthode dite *méthode des contraintes probabilistes* sont souvent cités. Cette approche consiste à assigner les tâches de manière à ce que la probabilité de respecter le temps de cycle soit supérieure, pour chaque poste de travail, à

1. Dans notre étude, nous supposons que le temps de cycle peut être supérieur ou égal à la charge réelle du poste de travail le plus occupé.
2. SALBP – Simple Assembly Line Balancing Problem (en anglais).

une valeur donnée appelée *niveau de confiance*. Par exemple, cette méthode a été appliquée dans Urban and Chiang (2006) et Ağpak and Gökçen (2007) pour l'équilibrage d'une ligne d'assemblage en U et dans Özcan (2010) pour la conception d'une ligne d'assemblage à double face. Dans ces articles, les auteurs utilisent une formulation de programmation linéaire en nombres entiers (PLNE) du problème correspondant qui intègre la contrainte de temps de cycle probabiliste. Sur la base des informations exprimant les durées des tâches, ils introduisent de nouvelles variables supplémentaires et utilisent diverses techniques de linéarisation afin d'obtenir à nouveau une formulation PLNE déterministe équivalente au problème probabiliste initial.

En ce qui concerne l'*approche floue*, les valeurs potentielles de la durée des tâches sont représentées sous la forme d'un ensemble flou dont la fonction d'appartenance décrit leur distribution possible. Comme dans le cas stochastique, il est nécessaire de contrôler la contrainte de temps de cycle pour affecter ces tâches aux postes de travail. Pour ce faire, il convient d'introduire une arithmétique floue et une méthode appropriée pour comparer ces ensembles flous. Une application relative aux tâches avec des durées floues a été présentée dans Tsujimura et al. (1995) et Gen et al. (1996) pour SALBP-1, dans Hop (2006) pour l'équilibrage d'une ligne à modèles mixtes et dans Zacharia and Nearchou (2012) pour une variante bi-objectif de SALBP-2.

Il convient toutefois de noter que l'utilisation de ces deux approches dans la pratique pourrait être un défi difficile à relever. En effet, les connaissances disponibles sur les données d'entrée ne sont pas toujours suffisantes pour dériver des distributions de probabilités ou de possibilités adéquates pour toutes les durées des tâches, en particulier si la conception de la ligne d'assemblage est planifiée pour la première fois. L'*approche robuste* (voir, *e.g.*, Kouvelis and Yu, 1997) est souvent plus pertinente dans de telles situations, car elle suppose que seul un ensemble discret de scénarios et/ou d'intervalles de réalisations potentielles de la durée des tâches est connu sans aucune distribution ou même qu'un ensemble de tâches dont la durée est susceptible de varier est fourni comme une donnée d'entrée du problème.

L'utilisation de la représentation *discrète* ou *intervalle* des scénarios modifie généralement l'objectif d'un problème considéré. En effet, une solution optimale trouvée pour un scénario peut perdre son optimalité et même sa faisabilité pour un autre scénario. Pour contourner cette situation, un critère appelé *robustesse absolue* ou *min-max* peut être appliqué. Largement utilisé dans l'optimisation robuste, il vise à rechercher une solution qui reste faisable pour tous les scénarios et qui présente les meilleures performances pour le plus mauvais d'entre eux. Par exemple, ce critère a été étudié dans Gurevsky et al. (2013b) pour SALBP-1 avec des durées des tâches sous forme d'intervalles. Pour trouver la solution mentionnée ci-dessus, les auteurs ont développé un algorithme de type «branch and bound». Une approche similaire a été appliquée dans Dolgui and Kovalev (2012) pour SALBP-2, mais avec un ensemble discret de scénarios. Dans ce dernier travail, la complexité de calcul de la recherche de la solution robuste a été présentée pour différents types de contraintes de précédence.

Le cas où seul un *ensemble de tâches incertaines* (avec des durées variables) peut être identifié sans aucune information supplémentaire est moins instructif, mais probablement le plus fréquent dans la pratique. En raison du manque d'informations, les méthodes utilisées pour les approches mentionnées précédemment ne sont pas applicables. Pour évaluer une solution dans une telle situation, Sotskov et al. (2005, 2006), en étudiant SALBP-1 et SALBP-2, ont suggéré un indicateur spécifique, appelé *rayon de stabilité*. Étant donné une solution, il est calculé comme l'amplitude maximale des déviations de la durée des tâches incertaines par rapport à leurs valeurs nominales pour lesquelles la faisabilité (ou l'optimalité) de la solution est garantie. Les auteurs montrent que cet indicateur peut servir de mesure de robustesse appropriée. En effet, plus sa valeur est élevée pour une solution étudiée, plus la robustesse de la configuration de ligne correspondante est grande. De plus, il a été prouvé que le calcul du rayon de stabilité au sens de la faisabilité est un problème polynomial pour toute solution admissible de SALBP-1. Ces résultats positifs ont inspiré plusieurs autres études. Ainsi, dans Gurevsky et al. (2012), les résultats obtenus pour SALBP-1 et SALBP-2 ont été généralisés pour SALBP-E. Une version robuste pour un problème plus complexe d'équilibrage de ligne d'assemblage soumis à l'incertitude de la durée des tâches a également été proposée dans Gurevsky et al. (2013a). Pour ce problème, le rayon de stabilité au sens de la faisabilité a été considéré comme le deuxième objectif à maximiser. Pour trouver un compromis entre sa valeur et le coût de la ligne, une approche heuristique de type ϵ-contraintes a été développée pour rechercher une approximation de l'ensemble de Pareto pour un tel problème d'optimisation bi-objectif.

C'est alors, en collaboration avec Olga Battaïa, Alexandre Dolgui et André Rossi, que nous avons

entrepris de populariser davantage notre approche alternative de la robustesse, en nous concentrant cette fois non pas sur le calcul du rayon de stabilité pour une configuration de ligne donnée, mais plutôt sur la recherche de la configuration la plus robuste avec un nombre fixe de postes de travail, c'est-à-dire celle qui présente la valeur de rayon de stabilité la plus élevée possible. Ainsi, pour le cas de lignes d'assemblage, nous avons réussi à développer dans Rossi et al. (2016) une formulation de programmation linéaire en nombres mixtes (PLNM) efficace et de nombreuses bornes supérieures serrées pour chaque norme l_1 et l_∞ dans lesquelles le rayon de stabilité peut être calculé et interprété. C'est ensuite dans le cadre de la thèse d'Aleksandr Pirogov, à laquelle ce chapitre est dédié, que nous avons décidé d'étendre les résultats obtenus dans Rossi et al. (2016) pour le cas des lignes de transfert où les tâches s'exécutent en parallèle au sein de blocs qui sont, à leur tour, activés séquentiellement dans les machines.

Le reste de ce chapitre est organisé comme suit. La section 1.2 présente les notations, définitions et propriétés de base du problème ainsi que le concept de l'approche robuste proposée et appliquée aux lignes de transfert. La section 1.3 est consacrée aux aspects algorithmiques du calcul du rayon de stabilité considéré pour deux normes. La complexité du problème et son modèle PLNM pour la norme l_∞ sont exposés et discutés dans la section 1.4. Les techniques de pré-traitement pour le modèle PLNM sont décrites dans la section 1.5. La section 1.6 montre une série d'expériences numériques et examine les résultats de calcul obtenus. À la fin de ce chapitre, nous fournissons une conclusion et présentons quelques perspectives de recherche ouvertes par ces travaux.

1.2 Notations et définitions de base

On considère qu'il existe un ensemble non vide de *tâches incertaines* $\widetilde{V}$ ($\widetilde{V} \subseteq V$) dont la durée peut s'écarter de sa valeur nominale sans aucune information supplémentaire. Toutes les autres tâches de $V \setminus \widetilde{V}$ sont nommées *certaines*. Les notations de base détaillées sont données ci-dessous :

- V est l'ensemble des tâches nécessaires, *i.e.*, $\{1, 2, \ldots, n\}$;
- $\widetilde{V}$ est l'ensemble des tâches incertaines, où $\widetilde{V} \subseteq V$;
- W est l'ensemble des machines disponibles, *i.e.*, $\{1, 2, \ldots, m\}$;
- G est un graphe acyclique orienté (V, A) représentant les contraintes de précédence, où A est l'ensemble des arcs ;
- t_j est la durée nominale non négative de la tâche j ;
- t est un vecteur exprimant les durées nominales des tâches, *i.e.*, $(t_1, t_2, \ldots, t_n)$;
- $F(t)$ est l'ensemble de toutes les solutions réalisables par rapport à un vecteur donné t ;
- Ξ est l'ensemble des vecteurs, chacun d'entre eux représentant les écarts non négatifs possibles des durées pour les tâches incertaines, *i.e.*, $\{\xi \in \mathbb{R}^n_+ \mid \xi_j = 0,\ j \in V \setminus \widetilde{V}\}$;
- T est le temps de cycle ;
- $r_{\max}$ est le nombre maximal de tâches qui peuvent être assignées à un bloc ;
- $b_{\max}$ est le nombre maximal de blocs pouvant être alloués à une machine ;
- U est l'ensemble de tous les blocs disponibles, *i.e.*, $\{1, 2, \ldots, mb_{\max}\}$;
- $U(p)$ est l'ensemble des blocs disponibles de la machine p, *i.e.*, $\{(p-1) \cdot b_{\max} + 1, \ldots, p \cdot b_{\max}\}$;
- V_k est l'ensemble des tâches assignées au bloc k ;
- $\widetilde{V}_k$ est l'ensemble de toutes les tâches incertaines assignées au bloc k, *i.e.*,$V_k \cap \widetilde{V}$;
- τ_k est le temps de traitement nominal du bloc k, *i.e.*, $\max_{j \in V_k} t_j$;
- $\widetilde{U}(p)$ est l'ensemble des blocs incertains de la machine p, chacun d'entre eux ayant au moins une tâche incertaine assignée ;
- $\widetilde{W}$ est l'ensemble des machines incertaines, où chacune possède au moins un bloc incertain.

Pour évaluer la robustesse d'une solution réalisable, nous utilisons le concept de *rayon de stabilité* dont la définition formelle nécessite quelques notations supplémentaires. Ainsi, le rayon de stabilité d'une solution réalisable $s \in F(t)$ peut être défini comme suit (voir, *e.g.*, Sotskov et al., 2006) :

$$\rho(s,t) = \max\{\epsilon \geq 0 \mid \forall \xi \in B(\epsilon) \quad (s \in F(t+\xi))\}, \tag{1.1}$$

où $B(\epsilon) = \{\xi \in \Xi \mid \|\xi\| \leq \epsilon\}$. En d'autres termes, $\rho(s,t)$ est déterminé comme la valeur du rayon de la plus grande boule fermée $B(\cdot)$, appelée *boule de stabilité*, représentant les écarts [3] de la durée nominale des tâches incertaines, pour laquelle s reste réalisable. Tout élément ξ de $B(\cdot)$ est évalué sur la base d'une norme donnée $\|\cdot\|$ définissant la distance entre les vecteurs t et $t+\xi$ (ou l'amplitude des écarts par rapport à t).

Dans la thèse d'Aleksandr Pirogov, deux normes l_1 ($\|\cdot\|_1$) et l_∞ ($\|\cdot\|_\infty$) ont été étudiées en détail, où par définition $\|\xi\|_1 = \sum_{j\in\widetilde{V}} \xi_j$ et $\|\xi\|_\infty = \max_{j\in\widetilde{V}} \xi_j$. En conséquence, les notations ρ_1, $B_1(\cdot)$ et ρ_∞, $B_\infty(\cdot)$ ont été utilisées pour l_1 et l_∞, respectivement.

Pour une solution donnée, le rayon de stabilité dans la norme l_1 est le montant total de l'augmentation de la durée que cette solution peut supporter tout en restant faisable, quelle que soit la manière dont cette durée supplémentaire est répartie entre les tâches incertaines. Cependant, sa valeur numérique est souvent limitée parce qu'elle couvre des situations qui peuvent être considérées comme improbables, comme lorsque toute l'augmentation de la durée est concentrée sur une seule tâche incertaine. Cette situation peut être considérée comme irréaliste, en particulier lorsque la valeur du rayon de stabilité est plusieurs fois supérieure à la durée minimale des tâches incertaines. On peut considérer qu'il est peu probable que l'augmentation de la durée d'une tâche incertaine courte atteigne plusieurs fois sa valeur nominale. Dans certains contextes, cependant, une telle situation peut ne pas être considérée comme totalement irréaliste. Une tâche de forage, par exemple, ne devrait pas durer plus de quelques secondes. Cependant, si le foret se casse pendant l'opération, l'ouverture du mandrin, l'installation d'un nouveau foret et la reprise de la tâche peuvent retarder la durée prévue d'une très grande partie du temps.

Le rayon de stabilité dans la norme l_∞ correspond à une hypothèse différente sur la façon dont les écarts peuvent affecter la durée des tâches incertaines. Il mesure l'augmentation maximale de la durée que chaque tâche incertaine peut supporter (indépendamment des autres tâches incertaines) sans compromettre la faisabilité de la solution. Elle est appropriée lorsque l'on suppose que toutes les tâches incertaines sont soumises au même risque d'augmentation de la durée.

Voyons un exemple illustratif de l'interprétation du rayon de stabilité dans les normes l_∞ et l_1. Le problème suivant est considéré : $n = 7$, $m = 1$, $\widetilde{V} = \{2, 5\}$, $t = (5, 4, 2, 3, 2.5, 2, 2.5)$, $T = 11.5$, $r_{\max} = 3$. Il n'y a pas de contrainte de précédence. L'affectation réalisable est illustrée par la Figure 1.1 : la charge totale de la machine est inférieure au temps de cycle. Les temps de traitement des blocs sont $\tau_1 = 5$, $\tau_2 = 3$ et $\tau_3 = 2.5$ par définition. Sur la base de la Figure 1.2, il est facile de voir que la plus grande augmentation (en traits hachurés) de la durée, qui peut être supportée simultanément par toute tâche incertaine, est égale à 1.25. Cette valeur définit donc le rayon de stabilité dans la norme l_∞. En ce qui concerne les Figures 1.3 et 1.4, nous pouvons remarquer que la plus grande augmentation de la durée, qui peut être supportée individuellement par une tâche incertaine, ne peut être supérieure à 1.5 unités de temps. Cette dernière valeur détermine le rayon de stabilité dans la norme l_1.

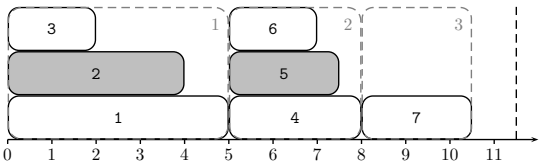

FIGURE 1.1 – Exemple d'affectation de 7 tâches à 3 blocs

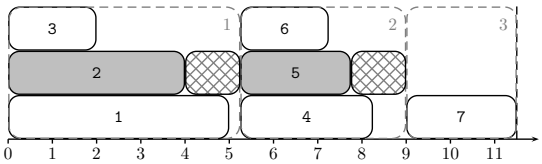

FIGURE 1.2 – La durée des tâches 2 et 5 est augmentée de 1.25 simultanément

3. Comme toute diminution de la durée des tâches ne peut compromettre la faisabilité de la solution, il suffit de ne considérer que les écarts non négatifs de la durée des tâches dans ce travail, *i.e.*, $\xi_j \in \mathbb{R}_+$ pour tout $j \in \widetilde{V}$.

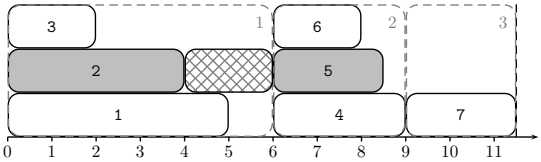

FIGURE 1.3 – La durée de la tâche 2 est augmentée de 2

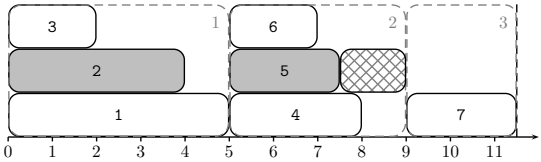

FIGURE 1.4 – La durée de la tâche 5 est augmentée de 1.5

Avant de poursuivre l'examen du rayon de stabilité, nous allons introduire un paramètre important, appelé *durée de réserve*. Il est déterminé uniquement pour les blocs incertains et calculé comme la différence entre le temps de traitement du bloc et la durée de sa tâche incertaine la plus longue, *i.e.*, $\tau_k - \max_{j \in \widetilde{V}_k} t_j$, $k \in \widetilde{U}(p)$, $p \in W$. Par conséquent, la durée de réserve peut être positive ou égale à zéro, voir respectivement la Figure 1.5 et la Figure 1.6. Ci-après, la durée de réserve d'un bloc incertain k de la machine p est désignée par $\Delta_k^{(p)}$ et la durée de réserve minimale de la machine $p \in \widetilde{W}$ par $\Delta_{\min}^{(p)} = \min_{k \in \widetilde{U}(p)} \Delta_k^{(p)}$.

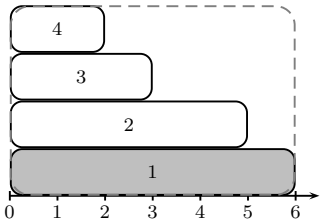

FIGURE 1.5 – Bloc avec durée de réserve nulle

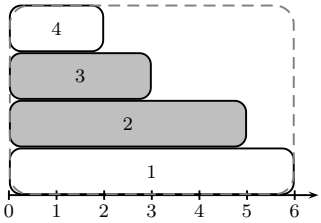

FIGURE 1.6 – Bloc avec durée de réserve égale à 1

1.3 Calcul des rayons de stabilité

Dans cette section, nous présentons comment calculer le rayon de stabilité pour une solution réalisable donnée dans deux normes, l_1 et l_∞. Les aspects de complexité algorithmique sont également discutés. Un exemple illustratif est également fourni. Les preuves des théorèmes sont données en détail dans Rossi et al. (2016) et Pirogov et al. (2021).

Théorème 1.1 (Pirogov et al. (2021)). *Le rayon de stabilité ρ_1 pour une solution réalisable donnée est calculé comme suit*

$$\rho_1 = \min_{p \in \widetilde{W}} \left\{ T - \sum_{k \in U(p)} \tau_k + \Delta_{\min}^{(p)} \right\}. \tag{1.2}$$

Il n'est pas difficile de voir que le calcul de ρ_1 peut être fait en temps $O(n)$ par une analyse séquentielle de tous les blocs de chaque machine. En d'autres termes, chaque tâche n'est examinée qu'une seule fois.

Dans le cas de $r_{\max} = 1$, la configuration de la ligne de transfert est transformée en celle de la ligne d'assemblage simple, puisqu'au plus une tâche peut être allouée à un bloc. Par conséquent, la durée de réserve de tous les blocs incertains devient égale à zéro, *i.e.*, $\Delta_k^{(p)} = 0$, $k \in \widetilde{U}(p)$, $p \in \widetilde{W}$. En conséquence, le Théorème 1.1 donne directement

Conséquence 1.1 (Rossi et al. (2016)). *Le rayon de stabilité ρ_1 pour une configuration donnée d'une ligne d'assemblage simple est calculé comme* $\min_{p \in \widetilde{W}} \left\{ T - \sum_{k \in U(p)} \tau_k \right\}$.

Pour toute machine incertaine p, introduisons la fonction suivante, qui sera utile dans la suite :

$$\theta(p, q) = \frac{T - \sum_{k \in U(p)} \tau_k + \sum_{i=1}^{q} \Delta_{\pi_i}^{(p)}}{q},$$

où $\Delta^{(p)}_{\pi_1} \leq \Delta^{(p)}_{\pi_2} \leq \ldots \leq \Delta^{(p)}_{\pi_{|\widetilde{U}(p)|}}$ est l'ordre non décroissant de la durée de réserve pour tous les blocs incertains de la machine $p \in \widetilde{W}$ et $q \in \{1, \ldots, |\widetilde{U}(p)|\}$.

Théorème 1.2 (Pirogov et al. (2021)). *Le rayon de stabilité ρ_∞ pour une solution réalisable donnée est calculé comme suit*

$$\rho_\infty = \min_{p \in \widetilde{W}} \min_{q=1,\ldots,|\widetilde{U}(p)|} \theta(p, q). \tag{1.3}$$

Il n'est pas difficile de voir que le calcul de ρ_∞ peut être effectué en temps $O(n + mb_{\max} \log b_{\max})$ par une analyse séquentielle de tous les blocs et en ordonnant ensuite les durées de réserve de chaque machine incertaine.

Sur la base des mêmes arguments, développés ci-dessus pour $r_{\max} = 1$ et le cas de la norme l_1, le Théorème 1.2 implique le résultat déjà connu pour les lignes d'assemblage dites «simples» :

Conséquence 1.2 (Sotskov et al. (2006)). *Le rayon de stabilité ρ_∞ pour une configuration donnée d'une ligne d'assemblage simple est calculé comme suit* $\min_{p \in \widetilde{W}} \frac{T - \sum_{k \in U(p)} \tau_k}{|\widetilde{U}(p)|}$.

Présentons un petit exemple illustratif d'une machine dont la charge effective est de 19 avec cinq blocs incertains et le temps de cycle $T = 21$ (voir Figure 1.8). Les cinq durées de réserve ordonnées et les valeurs correspondantes de $\theta(\cdot, q)$ sont indiquées dans la Table 1.1. La Figure 1.7 présente le graphique de $\theta(\cdot, q)$ avec un minimum global $\theta(\cdot, 3) = 1.67$. Suivant le Théorème 1.2, le rayon de stabilité ρ_∞ de la machine étudiée est égal à $\theta(\cdot, 3)$. Nous remarquons également que $\Delta^{(\cdot)}_3 \leq \rho_\infty \leq \Delta^{(\cdot)}_4$ est valide (voir Figure 1.9).

q	1	2	3	4	5
$\Delta^{(\cdot)}_q$	0.5	1.0	1.5	2.0	2.0
$\theta(\cdot, q)$	2.5	1.75	1.67	1.75	1.8

TABLE 1.1 – Valeurs de $\Delta^{(\cdot)}_q$ et $\theta(\cdot, q)$.

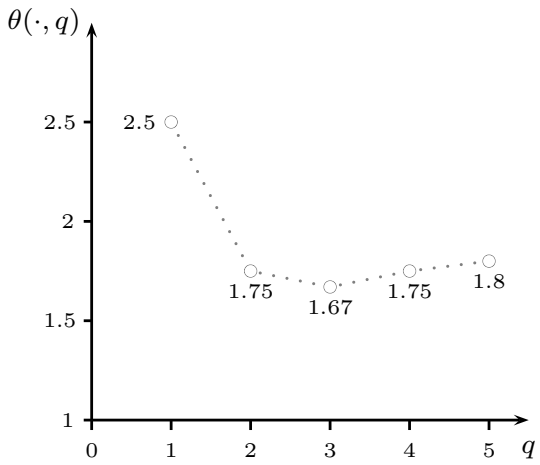

FIGURE 1.7 – Graphique de $\theta(\cdot, q)$

1.4 Formulation du problème

Dans cette section, nous nous focalisons sur le problème d'optimisation qui vise à déterminer une configuration de ligne ayant la plus grande valeur de rayon de stabilité. Nous notons ce problème étudié P et distinguons ses deux versions P_1 et P_∞ respectivement pour les normes l_1 et l_∞. Le problème P est

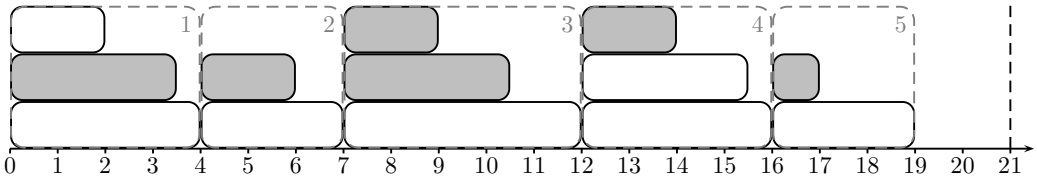

FIGURE 1.8 – Une machine avec cinq blocs incertains

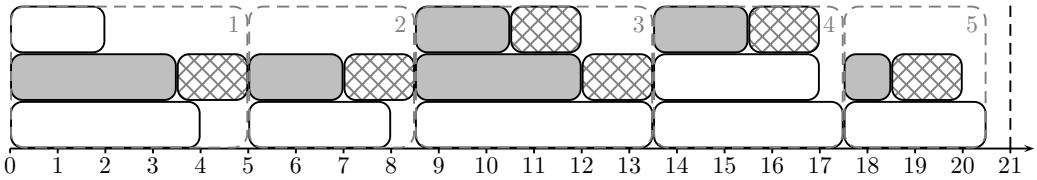

FIGURE 1.9 – La charge de la machine est perturbée par $\Delta_3^{(\cdot)} = 1.5$ pour chaque tâche incertaine.

fortement NP-difficile. En effet, il suffit de considérer une instance de P, où le nombre maximal de tâches par bloc est limité à 1, *i.e.*, $r_{\max} = 1$. Alors, P se réduit au même problème que celui étudié dans Rossi et al. (2016), qui a été prouvé être NP-difficile au sens fort.

Nous présentons ci-dessous une formulations PLNM pour P_∞. La formulation PLNM similaire pour P_1 peut être trouvée dans Pirogov et al. (2021). Pour cette formulation, la valeur de $b_{\max}$ est fixée à $\max\left\{k \mid \sum_{i=1}^{k} t_{\pi_i} \leq T\right\}$, où $(\pi_1, \pi_2, \ldots, \pi_n)$ est une permutation de V par rapport à l'ordre non décroissant de leurs durées. Il est facile de voir que cette dernière expression peut être calculée en temps $O(n \log n)$.

P_∞ est formulé au moyen d'un programme linéaire en nombres mixtes (PLNM) sur les variables de décision suivantes : ρ_∞ est la valeur du rayon de stabilité à maximiser ; x_{jk} est une variable binaire qui vaut un si et seulement si la tâche j est allouée au bloc k ; y_k est égal à 1 si le bloc k n'est pas vide et à 0 dans le cas contraire ; ξ_{jk} est un écart de la durée de la tâche j dans le bloc k ; $\tau_k \geq 0$ détermine le temps de traitement du bloc k. L'idée principale de la formulation PLNM pour P_∞ est que la durée de toutes les tâches incertaines peut être augmentée de ρ_∞ sans compromettre la faisabilité de la solution.

$$\max \quad \rho_\infty, \quad \text{s.c.}$$

$$\sum_{k \in U} x_{jk} = 1, \quad \forall j \in V \tag{1.4}$$

$$\sum_{j \in V} x_{jk} \leq r_{\max}, \quad \forall k \in U \tag{1.5}$$

$$x_{jk} \leq y_k, \quad \forall k \in U, \quad \forall j \in V \tag{1.6}$$

$$y_k \leq \sum_{j \in V} x_{jk}, \quad \forall k \in U \tag{1.7}$$

$$y_{k+1} \leq y_k, \quad \forall p \in W, \quad \forall k \in U(p) \setminus \{pb_{\max}\} \tag{1.8}$$

$$\xi_{jk} \leq T \cdot x_{jk}, \quad \forall k \in U, \quad \forall j \in \widetilde{V} \tag{1.9}$$

$$t_j \cdot x_{jk} + \xi_{jk} \leq \tau_k, \quad \forall k \in U, \quad \forall j \in \widetilde{V} \tag{1.10}$$

$$t_j \cdot x_{jk} \leq \tau_k, \quad \forall k \in U, \quad \forall j \in V \setminus \widetilde{V} \tag{1.11}$$

$$\rho_\infty = \sum_{k \in U} \xi_{jk}, \quad \forall j \in \widetilde{V} \tag{1.12}$$

$$\sum_{k \in U(p)} \tau_k \leq T, \quad \forall p \in W \tag{1.13}$$

$$\sum_{q=k}^{|U|} x_{iq} \leq \sum_{q=k+1}^{|U|} x_{jq}, \quad \forall (i,j) \in A, \quad \forall k \in U \tag{1.14}$$

$$\rho_\infty \geq 0,$$

$$\xi_{jk} \geq 0, \; x_{jk} \in \{0,1\}, \quad \forall j \in V, \quad \forall k \in U$$

$$y_k \in \{0,1\}, \quad \forall k \in U$$

Les contraintes (1.4) garantissent que chaque tâche est allouée à exactement un bloc. Les inégalités (1.5) montrent que chaque bloc contient au plus $r_{\max}$ tâches. Tout bloc ayant au moins une tâche assignée est considéré comme non vide, sinon il est vide, conformément aux contraintes (1.6) et (1.7). La contrainte (1.8) garantit que le bloc $k+1$ doit être vide si le bloc k est vide. En effet, pour briser la symétrie, ces contraintes imposent que lorsqu'un bloc est vide, tous les blocs suivants de la machine considérée soient également fermés. L'interdiction d'intercaler des blocs vides entre des blocs non vides réduit le nombre de solutions équivalentes au problème et a un impact positif sur le temps de résolution. Les inégalités (1.9) précisent que l'écart de la durée d'une tâche incertaine ne peut être supérieur au temps de cycle. Le temps de traitement d'un bloc n'est pas inférieur à la durée d'une tâche donnée ou à la durée déviée d'une tâche incertaine qui lui est attribuée, comme le prévoient (1.10) et (1.11). Les contraintes (1.4) et (1.9) impliquent que pour une tâche incertaine donnée j, seule une variable des ξ_{jk}, $k \in U$, a une valeur non nulle. Ceci, associé à (1.12), implique que l'écart de la durée pour toute tâche incertaine est fixé à ρ_∞. Quant aux contraintes (1.13), elles stipulent que la charge d'une machine ne doit pas dépasser le temps de cycle. Les contraintes de précédence sont exprimées par les inégalités (1.14).

1.5 Pré-traitement

Dans cette section, nous présentons quelques techniques utiles pour réduire l'espace de recherche de la formulation PLNM pour P_∞. Ainsi, la sous-section 1.5.1 est consacrée à la présentation de certaines méthodes visant à réduire ce que l'on appelle l'intervalle d'affectation pour chaque tâche et à trouver des blocs inutilisés (ou a priori vides). Ensuite, dans la sous-section 1.5.2, une heuristique est proposée pour générer des solutions réalisables comme points de départ de la formulation PLNM. Enfin, une approche globale de pré-traitement est présentée dans la sous-section 1.5.3.

1.5.1 Intervalles d'affectation et blocs vides

Nous développons ci-dessous un algorithme important, qui renvoie une borne inférieure sur le nombre de blocs nécessaires à l'affectation d'un ensemble donné de tâches. Cela peut être fait non seulement de manière triviale comme un rapport entre la cardinalité de cet ensemble et $r_{\max}$, mais aussi par rapport au temps de cycle. L'idée est d'abord de former les blocs de tâches à partir de cet ensemble de manière à ce que la somme des temps de traitement des blocs obtenus soit aussi petite que possible, puis de trouver une borne inférieure sur le nombre de machines nécessaires pour allouer tous ces blocs. Ainsi, le nombre de blocs recherchés est égal à l'indice du premier bloc de la dernière machine nécessaire. Cet algorithme est présenté formellement ci-dessous, où D est considéré comme un ensemble donné de tâches.

Algorithme $\alpha(\cdot)$

1. Soit $(\sigma_1, \sigma_2, \ldots, \sigma_{|D|})$ une permutation de D par rapport à l'ordre non croissant de leurs durées.
2. Former $\omega = \left\lceil \frac{|D|}{r_{\max}} \right\rceil$ blocs $B_1, B_2, \ldots, B_\omega$, dont chacun possède exactement $r_{\max}$ tâches sauf éventuellement le dernier, de sorte que la tâche σ_j se trouve dans le bloc $B_{\lceil \frac{j}{r_{\max}} \rceil}$, $j \in D$. Cela permet de déterminer le temps de traitement de chaque bloc obtenu : $\tau_k = \max_{i \in B_k} t_i$, $k = 1, 2, \ldots, \omega$.
3. La borne inférieure sur le nombre de machines nécessaires pour traiter tous les blocs obtenus est égale à $\lambda = \frac{1}{T} \cdot \sum_{k=1}^{\omega} \tau_k$. À partir de là, la borne inférieure retournée sur le nombre de blocs égale $b_{\max} \cdot \lceil \lambda - 1 \rceil + 1$.

Grâce à l'étape 1, l'algorithme $\alpha(D)$ peut être mis en œuvre pour s'exécuter en temps $O(|D| \log |D|)$.

Lors de la recherche d'une affectation réalisable de toutes les tâches données aux blocs et des blocs aux machines, des contraintes de précédence doivent être satisfaites. Par exemple, pour chaque tâche $j \in V$, l'ensemble de tous ses prédécesseurs, désigné par $P(j)$, doit être assigné avant j. De même, l'ensemble des successeurs de j, désigné par $S(j)$, doit être affecté après j. Ainsi, pour toute tâche j, nous introduisons ce que l'on appelle l'intervalle d'affectation $Q_j = [l_j, u_j]$, où l_j (resp. u_j) est l'indice le plus petit (resp. le plus élevé) du bloc capable d'accueillir la tâche j. Pour une résolution efficace du problème P_∞, ces intervalles doivent être aussi serrés que possible. Afin d'atteindre cet objectif, nous fournissons ci-dessous cinq règles de réduction différentes pour réduire Q_j :

- La **règle 1** est basée sur la connaissance de la quantité de tous les prédécesseurs et de tous les successeurs pour une tâche donnée dans le graphe de précédence : $l_j^{(1)} = \left\lceil \frac{|P(j)|}{r_{\max}} \right\rceil + 1$ et $u_j^{(1)} = mb_{\max} + 1 - \left(\left\lceil \frac{|S(j)|}{r_{\max}} \right\rceil + 1 \right)$ pour tout $j \in V$.
- La **règle 2** consiste à appliquer l'algorithme $\alpha(\cdot)$ sur les ensembles de prédécesseurs et de successeurs de chaque tâche : $l_j^{(2)} = \alpha(P(j)) + 1$ et $u_j^{(2)} = mb_{\max} + 1 - (\alpha(S(j)) + 1)$ pour tout $j \in V$.
- La **règle 3** est similaire à la règle 2, mais l'ensemble des tâches de l'algorithme $\alpha(\cdot)$ comprend également la tâche j elle-même : $l_j^{(3)} = \alpha(P(j) \cup \{j\})$ et $u_j^{(3)} = mb_{\max} + 1 - \alpha(S(j) \cup \{j\})$.
- La **règle 4** analyse les intervalles d'affectation de tous les prédécesseurs et successeurs directs pour toutes les tâches comme suit : $l_j^{(4)} = \max_{(i,j) \in A} l_i + 1$ et $u_j^{(4)} = \min_{(j,i) \in A} u_i - 1$ pour tout $j \in V$.
- La **règle 5** prend également en compte (par rapport à la règle 4) la cardinalité des ensembles de prédécesseurs directs $P^*(j)$ et de successeurs directs $S^*(j)$ de j : $l_j^{(5)} = \min_{(i,j) \in A} l_i + \left\lceil \frac{|P^*(j)|}{r_{\max}} \right\rceil$ et $u_j^{(5)} = \max_{(j,i) \in A} u_i - \left\lceil \frac{|S^*(j)|}{r_{\max}} \right\rceil$.

Les intervalles d'affectation réduits servent à détecter un ensemble de blocs inutilisés (ou a priori vides), puis à tirer parti de cet élément d'information afin de réduire encore davantage ces intervalles. L'algorithme $\beta(\cdot)$ présente un algorithme de recherche de blocs inutilisés pour une machine donnée p.

Algorithme $\beta(\cdot)$

1. Calculer l'ensemble $V(p) = \{j \in V \mid Q_j \cap U(p) \neq \emptyset\}$ des tâches affectables à la machine p.
2. Sur la base de $V(p)$, déterminer le nombre maximal de blocs $b_{\max}^{(p)}$ pour la machine p en déterminant $\max \left\{ k \mid \sum_{i=1}^{k} t_{\pi_i} \leq T \right\}$, où $(\pi_1, \pi_2, \ldots, \pi_{|V(p)|})$ est une permutation de $V(p)$ selon l'ordre non décroissant de leurs durées. Ainsi, l'ensemble $\widehat{U}(p) = \{(p-1)b_{\max} + 1 + b_{\max}^{(p)}, \ldots, pb_{\max}\}$ de blocs inutilisés (ou a priori vides) est introduit pour la machine p.
3. Pour chaque tâche $j \in V(p)$, mettre à jour son intervalle Q_j comme suit. Si $l_j \in \widehat{U}(p)$, fixer $l_j = pb_{\max} + 1$. Si $u_j \in \widehat{U}(p)$, alors $u_j = (p-1)b_{\max} + b_{\max}^{(p)}$.

Grâce à l'étape 2, le temps d'exécution de l'algorithme $\beta(\cdot)$ n'est pas supérieur à $O(n \log n)$.

L'ensemble du processus de recherche des blocs inutilisés et de réduction des intervalles d'affectation est décrit dans l'approche `Reduction`, où $[1, mb_{\max}]$ est l'intervalle d'affectation initial pour chaque tâche.

Approche `Reduction`

1. Pour chaque tâche $j \in V$, calculer $Q_j^{(1)}$, $Q_j^{(2)}$ et $Q_j^{(3)}$ en appliquant respectivement les règles 1, 2 et 3. Mettre à jour les intervalles Q_j comme suit : $l_j := \max\{l_j, l_j^{(1)}, l_j^{(2)}, l_j^{(3)}\}$ et $u_j := \min\{u_j, u_j^{(1)}, u_j^{(2)}, u_j^{(3)}\}$.
2. Pour chaque tâche $j \in V$, calculer $Q_j^{(4)}$ et $Q_j^{(5)}$ en appliquant respectivement les règles 4 et 5. Mettre à jour les intervalles Q_j comme suit : $l_j := \max\{l_j, l_j^{(4)}, l_j^{(5)}\}$ et $u_j := \min\{u_j, u_j^{(4)}, u_j^{(5)}\}$.

3. Pour chaque machine $p \in W$, appliquer l'algorithme $\beta(p)$ et mettre à jour les intervalles Q_j conformément aux ensembles construits $\widehat{U}(p)$ de blocs inutilisés.
4. S'il existe une tâche $j \in V$ telle que Q_j a été réduit lors des étapes 2 ou 3, alors répéter l'étape 2. Sinon, arrêter.

Le temps d'exécution de l'approche `Reduction` ne dépasse pas $O(n^2 m b_{\max} G_{\max})$, où

$$G_{\max} = \max\{\max_{j \in V} |P^*(j)|, \max_{j \in V} |S^*(j)|\}.$$

1.5.2 Heuristique

Sur la base des idées présentées dans Gurevsky et al. (2013a), nous décrivons dans cette sous-section une méthodologie heuristique générale applicable au problème P_∞. Ensuite, sa version à démarrage multiple est également fournie.

L'heuristique, notée $\mathcal{H}(\rho)$, vise à construire une solution réalisable dont le rayon de stabilité est strictement supérieur à une valeur donnée ρ. Son principe de base est d'affecter autant de tâches que possible au bloc courant de la machine courante. Au début, une solution partiellement construite ne contient qu'une seule machine avec un bloc vide. L'heuristique affecte des tâches à ce bloc jusqu'à ce qu'aucune tâche ne puisse être ajoutée en raison des contraintes existantes. Ensuite, un nouveau bloc vide (ou une nouvelle machine avec un bloc vide) est ouvert et devient courant. Ce processus se poursuit jusqu'à ce que toutes les tâches soient affectées et qu'une solution réalisable soit obtenue ou qu'il n'y ait pas de possibilité de construire une solution réalisable.

L'heuristique utilise ce que l'on appelle la liste des tâches candidates $\mathcal{CL}(k, p)$, qui contient toutes les tâches pouvant être affectées au bloc courant k de la machine courante p. Cette liste est construite de la manière suivante : l'ensemble des tâches non encore affectées est analysé et la tâche j est ajoutée à $\mathcal{CL}(k, p)$ si toutes les conditions suivantes sont remplies :

1. j n'a pas de prédécesseur ou tous les prédécesseurs de j sont déjà affectés,
2. l'affectation de j ne viole pas la contrainte de temps de cycle,
3. l'affectation de j ne dépasse pas le nombre maximal de tâches par bloc $r_{\max}$,
4. l'affectation de j assure que la solution partielle a le rayon de stabilité supérieur à ρ.

Une version multi-start de $\mathcal{H}(\rho)$, notée $\mathcal{M}(\rho)$, apparaît naturellement. En effet, dans cette nouvelle version, chaque tentative de $\mathcal{H}(\rho)$ cherche à construire une nouvelle solution réalisable, dont le rayon de stabilité est plus grand que celui de la meilleure solution connue. Pour la condition d'arrêt, il suffit juste de limiter le nombre de tentatives sans amélioration à une valeur fixe, notée `limit`.

1.5.3 Approche globale de pré-traitement

Basée sur la stratégie heuristique et les règles de réduction, cette sous-section fournit une approche globale des améliorations pour la formulation P_∞. L'une des idées principales de cette approche se trouve dans la déclaration suivante.

Déclaration 1.1. *Soit (s^0, ρ^0) une paire composée respectivement d'une solution réalisable et de son rayon de stabilité pour une instance $\mathcal{I}$ du problème P_∞. Alors s^0 est également une solution réalisable pour l'instance $\mathcal{I}'$, qui est obtenue à partir de $\mathcal{I}$ en augmentant la durée de toutes les tâches incertaines de ρ^0. En outre, tous les intervalles d'affectation et les blocs inutilisés de $\mathcal{I}'$ sont également valables pour $\mathcal{I}$.*

En d'autres termes, la Déclaration 1.1 indique que l'analyse de $\mathcal{I}'$ peut fournir des intervalles d'affectation plus serrés pour $\mathcal{I}$. Le fonctionnement global de l'approche de pré-traitement est présenté plus bas. Ici, l'heuristique multi-start $\mathcal{M}$ commence à chercher des solutions admissibles en utilisant n'importe quelle valeur strictement négative (par exemple -1 comme ci-dessous) comme borne inférieure du rayon de stabilité.

Approche `Glob-Proc`

1. Soit (s^*, ρ^*) respectivement la meilleure solution réalisable et sa valeur de rayon de stabilité après application de $\mathcal{M}(-1)$.
2. Pour chaque tâche incertaine $j \in \widetilde{V}$, fixer $t_j := t_j + \rho^*$.
3. Appliquer l'approche `Reduction` pour construire les ensembles de blocs inutilisés $\widehat{U}(p)$, $p \in W$, et réduire les intervalles Q_j, $j \in V$.
4. Pour chaque tâche incertaine $j \in \widetilde{V}$, fixer $t_j := t_j - \rho^*$.
5. Considérer s^* comme une solution réalisable de départ pour la formulation PLNM correspondante et ajouter les coupes suivantes basées sur l'optimalité :

$$x_{jk} = 0, \quad \forall j \in V, \quad \forall k \notin Q_j, \tag{1.15}$$

$$y_k = 0, \quad \forall p \in W, \quad \forall k \in \widehat{U}(p). \tag{1.16}$$

Ici, les contraintes (1.15) modélisent l'intervalle d'affectation pour toute tâche $j \in V$ et les contraintes (1.16) décrivent tous les blocs inutilisés. Ces contraintes n'éliminent aucune solution optimale, mais excluent certaines solutions réalisables. C'est pourquoi on les appelle «optimality-based cuts» (coupes fondées sur l'optimalité) dans la littérature anglo-saxonne. Elles peuvent être ajoutées au modèle PLNM chaque fois qu'une nouvelle meilleure solution entière réalisable est trouvée par le solveur. Pour ce faire, on peut utiliser la routine `LazyConstraints`, qui est normalement proposée par des solveurs comme CPLEX ou GUROBI.

1.6 Résultats numériques

Trois séries S1, S2 et S3 d'instances de référence bien connues[4] ont été utilisées pour tester notre modèle PLNM et nos approches de pré-traitement. Toutes les instances d'une même série partagent le même temps de cycle, le même nombre de tâches et de machines ainsi que le même ordre de densité (OS) pour le graphe de précédence correspondant. Le OS d'un graphe de précédence est défini comme le nombre d'arcs dans sa fermeture transitive divisé par $n \cdot (n-1)/2$. Chaque série contient 50 instances, dont les caractéristiques susmentionnées figurent dans la Table 1.2. En outre, nous supposons que le nombre de tâches incertaines $|\widetilde{V}|$ est de $\{\lceil \frac{n}{2} \rceil, \lceil \frac{3n}{4} \rceil, n\}$, où $\widetilde{V}$ est généré en prenant les $|\widetilde{V}|$ premiers éléments d'une permutation aléatoire de $\{1, \ldots, n\}$, associée à chaque instance. Enfin, le nombre maximal de tâches par bloc $r_{\max}$ varie dans $\{2, 3\}$. Les tests ont été effectués sur un Apple MacBook Pro, équipé d'un Intel Core i5 cadencé à 2.7 GHz et disposant de 8 Go de RAM. GUROBI 9.0 a été utilisé comme solveur MIP pour traiter la formulation PLNM du problème P_∞. Le temps de calcul a été limité à 600 secondes par instance et le paramètre `limit` a été fixé à $100n$.

Série	T	n	m	OS
S1	70	25	5	0.50
S2	70	25	5	0.15
S3	70	50	10	0.90

Table 1.2 – Caractéristiques des séries S1, S2 et S3

Les résultats numériques pour la formulation PLNM sont divisés en deux parties : Table 1.3 pour $r_{\max} = 2$ et Table 1.4 pour $r_{\max} = 3$. Les deux premières colonnes de chaque table correspondent respectivement à la série et au nombre de tâches incertaines. Les trois colonnes suivantes (resp. dernières colonnes) décrivent le comportement de la formulation PLNM sans pré-traitement (resp. avec pré-traitement). Elles indiquent successivement le nombre de solutions optimales trouvées, l'écart moyen `GAP` et le temps moyen `CPU` pour chaque catégorie d'instances.

4. http://pagesperso.ls2n.fr/~gurevsky-e/data/R-TLBP.zip

TABLE 1.3 – Résultats pour P_∞ sans et avec pré-traitement pour $r_{\max} = 2$

Série	$\lvert\widetilde{V}\rvert$	P_∞ sans pré-traitement			P_∞ avec pré-traitement		
		#OPT	Moy. GAP, (%)	Moy. CPU, (s.)	#OPT	Moy. GAP, (%)	Moy. CPU, (s.)
S1	$\lceil\frac{n}{2}\rceil$	12	93.3	533.20	35	19.6	238.91
	$\lceil\frac{3n}{4}\rceil$	4	206.9	583.30	28	25.2	337.82
	n	0	339.1	600.00	41	11.1	176.46
S2	$\lceil\frac{n}{2}\rceil$	0	140.5	600.00	2	120.1	585.38
	$\lceil\frac{3n}{4}\rceil$	0	264.7	600.00	1	200.8	591.29
	n	0	422.1	600.00	3	197.5	568.70
S3	$\lceil\frac{n}{2}\rceil$	0	$\ominus$	600.00	47	2.2	97.02
	$\lceil\frac{3n}{4}\rceil$	0	$\ominus$	600.00	49	2.0	43.77
	n	0	$\ominus$	600.00	50	$\oplus$	12.85

($\oplus$) Toutes les solutions optimales ont été trouvées dans le délai de 600 secondes.
($\ominus$) Aucune solution réalisable n'a été trouvée dans le délai de 600 secondes.

Pour la Table 1.3, nous constatons que le problème P_∞ sans pré-traitement est très difficile à résoudre. Le meilleur résultat est obtenu pour le cas de 50% de tâches incertaines dans la série S1, pour laquelle seulement 12 solutions optimales sur 50 ont été trouvées. Dans la série S2, seules des solutions réalisables sont obtenues, et l'écart moyen GAP fourni par le solveur est assez important. Dans la série S3, aucune solution réalisable n'est trouvée, ce qui est indiqué par $\ominus$ dans la colonne de GAP. En ce qui concerne la résolution de P_∞ avec pré-traitement, nous pouvons remarquer une vrai amélioration des résultats pour les séries S1 et S3 où 69.33% et 97.33% des solutions optimales ont été respectivement trouvées. Seule la série S2 reste inaccessible même avec un pré-traitement, car seulement 6 des 150 instances ont été résolues de manière optimale. Cela peut s'expliquer par la piètre efficacité des techniques de pré-traitement pour les instances ayant une faible densité de ces graphes de précédence.

TABLE 1.4 – Résultats pour P_∞ sans et avec pré-traitement pour $r_{\max} = 3$

Série	$\lvert\widetilde{V}\rvert$	P_∞ sans pré-traitement			P_∞ avec pré-traitement		
		#OPT	Moy. GAP, (%)	Moy. CPU, (s.)	#OPT	Moy. GAP, (%)	Moy. CPU, (s.)
S1	$\lceil\frac{n}{2}\rceil$	48	1.8	224.04	50	$\oplus$	10.72
	$\lceil\frac{3n}{4}\rceil$	42	14.1	325.01	50	$\oplus$	11.09
	n	42	21.2	293.88	50	$\oplus$	0.33
S2	$\lceil\frac{n}{2}\rceil$	13	20.4	514.95	28	10.2	324.80
	$\lceil\frac{3n}{4}\rceil$	5	94.2	564.61	17	68.9	426.47
	n	18	69.6	466.89	46	0.3	71.28
S3	$\lceil\frac{n}{2}\rceil$	0	$\ominus$	600.00	48	6.4	104.84
	$\lceil\frac{3n}{4}\rceil$	0	$\ominus$	600.00	49	2.0	48.32
	n	0	$\ominus$	600.00	50	$\oplus$	9.92

($\oplus$) Toutes les solutions optimales ont été trouvées dans le délai de 600 secondes.
($\ominus$) Aucune solution réalisable n'a été trouvée dans le délai de 600 secondes.

En ce qui concerne le problème P_∞ avec $r_{\max} = 3$ (voir la Table 1.4), nous observons que 88% des instances ont été résolues à l'optimalité sans utiliser de pré-traitement pour la série S1. Pour cette même série, grâce au pré-traitement, 100% des instances ont été traitées de manière optimale, avec un temps moyen CPU ne dépassant pas 12 secondes par instance. La série S2 s'est avérée plus facile à résoudre dans le cas de $r_{\max} = 3$ que $r_{\max} = 2$, même si seulement 36 (resp. 91) des 150 instances ont été traitées à l'optimum sans pré-traitement (resp. avec pré-traitement). Quant à la série S3, pour P_∞ sans pré-traitement, nous avons toujours la même situation que pour $r_{\max} = 2$, à savoir qu'aucune solution

réalisable n'a été trouvée. Cependant, avec le pré-traitement, 98% des instances ont été résolues de manière optimale avec un temps moyen `CPU` ne dépassant pas 105 secondes par instance.

Toutes ces observations nous amènent à conclure que la formulation PLNM de base (sans pré-traitement) n'est pas compétitive, même pour les instances de petite et moyenne taille. Dans le même temps, les techniques de pré-traitement apportent une réelle amélioration lors de la résolution d'instances ayant une densité assez élevée de ces graphes de précédence.

1.7 Conclusion et perspectives

Ce chapitre traite de l'équilibrage robuste des lignes de transfert. Il s'agit de trouver une configuration de ligne avec le plus grand rayon de stabilité sous réserve d'un nombre restreint de machines, d'un temps de cycle fixe, de contraintes de précédence et de la variabilité de la durée des tâches. Le rayon de stabilité est évalué dans les normes l_1 et l_∞. Pour chaque norme, le problème correspondant, désigné respectivement par P_1 et P_∞, a été prouvé comme étant fortement NP-difficile. Une formulation PLNM a été proposée pour P_∞. Une heuristique, cinq règles de réduction pour les intervalles d'affectation et des techniques de pré-traitement ont également été développées afin d'améliorer le modèle PLNM. Les résultats numériques montrent que les renforcements proposés améliorent considérablement la recherche de solutions optimales. Une des pistes de recherche futures pourrait consister à étudier des règles de réduction plus puissantes pour les intervalles d'affectation, sur la base de la reformulation du problème étudié en un problème d'ordonnancement.

Les résultats de ce chapitre sont issus de la thèse de doctorat d'Aleksandr Pirogov. Ils sont également publiés dans le journal EJOR [5] (voir Pirogov et al., 2021). De plus, dans le cadre de la même thèse, nous avons également développé un autre indicateur de robustesse, appelé *facteur de stabilité*, qui contrôle la variation relative de la durée des tâches plutôt que la variation absolue comme c'était le cas pour le concept de rayon de stabilité. L'étude de ce nouvel indicateur pour le cas de lignes d'assemblage a été publiée dans le journal DAM [6] (voir Gurevsky et al., 2022b) où le développement d'une formulation PLNM optimisée et de nombreuses bornes supérieures très utiles ont été proposés. Il convient également de noter que les résultats de cette thèse ont été présentés lors de 4 conférences internationales/nationales et 1 séminaire invité (voir l'annexe A pour plus de détails). Après la thèse et un post-doc de deux ans à l'Université de Lancaster en Angleterre, Aleksandr, quant à lui, occupe actuellement le poste de «Data Scientist» chez Tesco à Londres.

L'étude de nombreux indicateurs de robustesse dans le contexte de l'équilibrage de lignes de production nous a logiquement conduit à explorer une nouvelle variante du problème robuste de bin-packing qui en découle. Celle-ci consiste à minimiser le nombre de boîtes de capacité identique à utiliser, mais qui prévoit toutefois un espace libre pré-défini dans chaque boîte dès lors qu'au moins un objet de taille variable y est placé. Pour résoudre efficacement cette variante de bin-packing, nous avons développé la méthode de type «branch-and-price» appropriée en collaboration avec Xavier Schepler. Une des particularités de la méthode développée est la nécessité de résoudre deux sous-problèmes de pricing différents afin de retrouver des colonnes de coût réduit négatif pour ensuite les insérer dans le problème maître. Ce travail a été publié dans le journal EJOR (voir Schepler et al., 2022). En ce qui concerne Xavier Schepler, il a collaboré avec nous de 2016 à 2017 en tant que post-doctorant à l'Université d'Angers dans le projet «ADLACI», financé par le programme régional RFI AtlanSTIC 2020.

Dans le cadre de la même direction de recherche, André Rossi et moi-même avons récemment établi une nouvelle collaboration très enrichissante et dynamique avec Rui S. Shibasaki (Université de Picardie Jules Verne, Amiens). Ensemble, nous avons publié un article dans le journal EJOR (voir Shibasaki et al., 2024) qui se focalise sur l'application de la décomposition de Dantzig-Wolfe pour trouver une borne supérieure de bonne qualité sur le rayon de stabilité pour le problème introduit dans Rossi et al. (2016). Pour le même problème, nous avons également commencé à étudier l'efficacité et les limites de la décomposition de Benders. Sans oublier un travail en cours sur l'extension des résultats obtenus dans Gurevsky et al. (2022b), mais cette fois pour les lignes de transfert.

5. EJOR – European Journal of Operational Research, édité par le groupe Elsevier B.V.
6. DAM – Discrete Applied Mathematics, édité par le groupe Elsevier B.V.

Chapitre 2

Équilibrage de lignes de production reconfigurables avec équipement modulaire

Sommaire

2.1 Introduction **23**
2.2 Revue de la littérature **25**
2.3 Formulation PLNB compacte **28**
2.4 Reformulation du modèle PLNB compact **30**
2.4.1 Algorithme de génération de modules 32
2.4.2 Génération de contraintes 33
2.5 Résultats numériques **33**
2.5.1 Comparaison des formulations 34
2.6 Conclusion et perspectives **36**

2.1 Introduction

La modularité est considérée comme l'une des principales caractéristiques des systèmes de production reconfigurables (RMS [7]) (voir, *e.g.*, Koren et al., 1999). Le terme *modularité* est employé pour décrire l'utilisation d'unités indépendantes, normalisées et interchangeables, appelées modules, afin d'exécuter une variété de fonctions (voir, *e.g.*, Huang and Kusiak, 1998). Ainsi, une structure modulaire est un concept dans lequel les fonctionnalités sont divisées en entités discrètes, cohérentes et autonomes. Elle permet aux RMS de s'adapter aux changements de manière rapide, efficace et rentable. Cela est rendu possible par l'utilisation de machines reconfigurables qui peuvent être facilement modifiées en réarrangeant, remplaçant ou ajoutant des modules, qui subdivisent effectivement le système en composants modulaires simplifiés (voir, *e.g.*, Baldwin and Clark, 2006). La modularité permet également de s'adapter avec souplesse à des environnements dynamiques, car chaque module peut être facilement modifié ou remplacé sans affecter le reste du système (voir, *e.g.*, Shaik et al., 2015). De plus, un système de fabrication modulaire permet également de mieux faire face à des événements inattendus tels que des pannes de machine, puisque la défaillance d'un module n'a pas nécessairement d'incidence sur les autres. En outre, les systèmes modulaires sont plus rentables à développer et à modifier que les systèmes non modulaires, car les modules individuels peuvent être personnalisés ou mis à niveau sans qu'il soit nécessaire de modifier l'ensemble du système.

7. RMS – Reconfigurable Manufacturing Systems (en anglais).

Malgré ses nombreux avantages, la conception d'un système de production modulaire reste un problème très complexe, puisqu'il s'agit généralement de déterminer les modules nécessaires à la production d'un produit, chaque module effectuant des tâches spécifiques. En outre, les modules doivent être affectés à des machines compatibles avec leur logiciel et leur matériel. Lors de la conception d'un système de production modulaire, il faut trouver un compromis entre le nombre de fonctions/tâches que les modules peuvent accomplir et leur coût. Les modules plus grands sont capables d'exécuter plus de fonctions et nécessitent moins d'interfaces, ce qui permet de réduire le nombre de reconfigurations nécessaires, mais ils peuvent être coûteux à concevoir, à développer et à entretenir. En revanche, les modules plus petits sont souvent moins coûteux, mais impliquent davantage d'interfaces, ce qui augmente la complexité du système et le nombre de reconfigurations nécessaires.

L'objectif de ce travail est d'examiner le problème de la conception combinatoire et de l'équilibrage d'une ligne modulaire capable de fabriquer plusieurs produits dans le contexte de RMS. La ligne considérée est composée de machines modulaires, où chaque module est constitué d'une pièce d'équipement physique capable d'effectuer un nombre limité de tâches de production. Cette conception modulaire facilite la reconfiguration de la ligne pour l'adapter à différents produits ou répondre à l'évolution de la demande (voir, *e.g.*, Koren et al., 1999). Nous nous concentrons ici uniquement sur le premier cas mentionné ci-dessus et n'abordons pas le second, à savoir la réponse à l'évolution de la demande.

Plus précisément, la ligne étudiée est composée d'un nombre fixe de machines modulaires disposées de manière linéaire et reliées par un système de manutention tel qu'un convoyeur. Chaque machine dispose d'un nombre limité d'emplacements dans lesquels les modules peuvent être placés. Par souci de simplicité, on suppose que les machines sont compatibles avec tous les modules. Les produits sont traités de manière séquentielle par les modules lorsqu'ils passent d'une machine à l'autre. Chaque module est capable d'effectuer un nombre limité de tâches, également de manière séquentielle, en suivant un ordre prédéfini qui n'affecte pas les contraintes de précédence entre les tâches. Ces dernières sont généralement exprimées par un graphe acyclique orienté, dont les nœuds (resp. les arcs) représentent les tâches (resp. les relations de précédence entre les tâches). Le temps de travail d'une machine est déterminé par la somme des durées de toutes les tâches effectuées par les modules de cette machine. Ce temps ne doit pas dépasser une valeur seuil donnée, appelée aussi *temps de cycle*. Enfin, la ligne étudiée peut traiter plusieurs types de produits par lots. Les relations de précédence des tâches et le temps de cycle de chaque type de produit peuvent être différents. Pour accommoder chaque type de produit, une configuration spécifique de la ligne est nécessaire, ce qui implique de reconfigurer les machines en ajoutant, déplaçant ou supprimant des modules.

L'objectif est alors de créer une configuration de ligne réalisable pour chaque type de produit, avec le nombre minimum de modules différents nécessaires. Il s'agit d'identifier les tâches que chaque module est capable d'accomplir et d'organiser ces modules en différentes configurations pour produire les divers produits. Les modules doivent être positionnés dans les machines disponibles de manière à répondre aux exigences de production. Pour résoudre ce problème d'optimisation, une formulation compacte de programmation linéaire en nombres binaires (PLNB) est d'abord proposée. Ensuite, une reformulation plus efficace est développée, basée sur une approche de décomposition.

La Figure 2.1 fournit un exemple du problème décrit ci-dessus. Elle montre deux graphes de précédence correspondant à deux produits. On peut remarquer que les produits partagent le même ensemble de tâches à exécuter. Cependant, les graphes de précédence et les durées des tâches sont différents. Le temps de cycle de chaque produit est également différent : 95 unités de temps pour le produit 1 et 75 unités de temps pour le produit 2.

Si nous supposons qu'aucun module ne peut effectuer plus de deux tâches et qu'une machine peut contenir au maximum deux modules, alors une composition optimale, représentée dans la Figure 2.2, nécessite six modules m_i, $i \in \{1, \ldots, 6\}$, pour produire les deux produits. Sur la base de cette composition, deux configurations peuvent être construites, comme le montre la Figure 2.3, où la configuration 1 (resp. 2) correspond à la disposition de ces modules dans les trois machines disponibles de manière à produire le produit 1 (resp. 2). Le passage d'une configuration à l'autre peut se faire en réarrangeant certains modules entre les machines.

Des études antérieures ont abordé des problèmes similaires dans le contexte de l'équilibrage de lignes d'assemblage/transfert, où un seul produit était pris en compte. Le présent travail constitue, à notre connaissance, la première tentative de traiter ce type de problème en tenant compte de plusieurs produits.

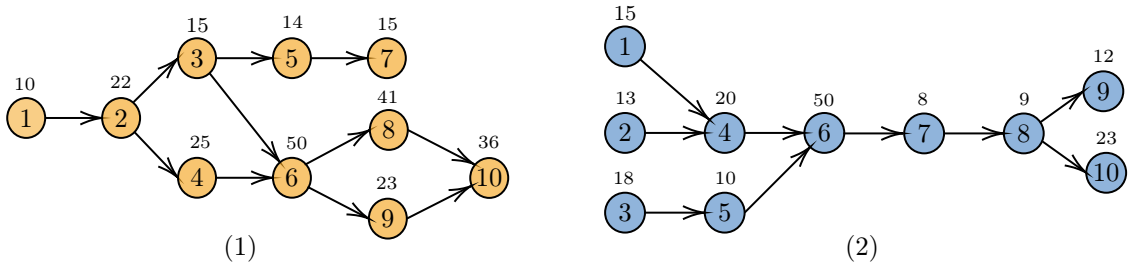

FIGURE 2.1 – (1) et (2) sont les graphes de précédence qui correspondent aux produits 1 et 2.

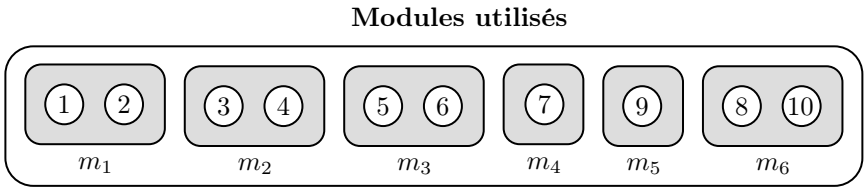

FIGURE 2.2 – Les modules optimaux générés

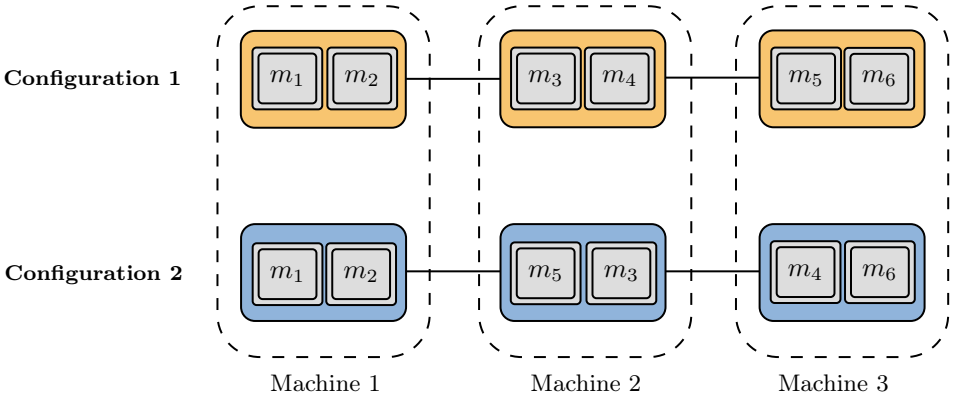

FIGURE 2.3 – Configurations optimales de produits qui minimisent le nombre total de modules.

Le reste du présent chapitre est organisé en cinq sections. La section 2.2 présente une analyse documentaire des problèmes connexes. Une formulation PLNB compacte pour le problème étudié est fournie dans la section 2.3. Sa reformulation plus efficace est présentée dans la section 2.4. Les résultats numériques obtenus pour ces deux formulations sont présentés et analysés dans la section 2.5. Enfin, dans la section 2.6, nous formulons quelques remarques finales et suggérons quelques orientations intéressantes pour la recherche future.

2.2 Revue de la littérature

Cette section présente une revue de la recherche liée au sujet de ce chapitre. Elle s'articule autour de deux axes principaux : la recherche sur les problèmes d'équilibrage de lignes d'assemblage à modèles multiples et, enfin, les publications sur la sélection de l'équipement et la modularité dans l'équilibrage de lignes d'assemblage.

Lorsqu'il s'agit de traiter plus d'un type de produit, deux variantes du SALBP [8] sont examinées dans la littérature (voir, *e.g.*, Scholl, 1999) : les problèmes d'équilibrage de lignes d'assemblage mixtes et à

8. SALBP – Simple Assembly Line Balancing Problem (en anglais).

modèles multiples. Le premier concerne une ligne unique où plusieurs produits de la même famille sont traités simultanément (voir, *e.g.*, Bukchin and Rabinowitch, 2006; Battini et al., 2006; Erel and Gokcen, 1999). Le second problème concerne une ligne unique où différents types de produits de la même famille sont fabriqués en lots séparés (voir, *e.g.*, van Zante-de Fokkert and de Kok, 1997). Ces dernières lignes doivent être reconfigurées lorsque l'on passe d'un type de produit à un autre. La reconfiguration consiste à rééquilibrer la ligne en réaffectant certaines tâches entre les postes de travail existants. Peu de recherches ont été menées sur le problème d'équilibrage de lignes d'assemblage à modèles multiples (MuMALBP [9]). Cependant, l'émergence de systèmes de production flexibles et reconfigurables implique que ce type de ligne devient plus avantageux, car il peut offrir différentes configurations (voir, *e.g.*, Battaïa and Dolgui, 2022).

Lapierre et al. (2000) et Tóth et al. (2018) considèrent une ligne d'assemblage de cartes de circuits imprimés (PCB [10]) à modèles multiples. Dans Lapierre et al. (2000), un modèle PLNM est d'abord formulé pour générer une configuration de ligne admissible pour chaque type de PCB. La fonction objectif cherche à minimiser le temps de cycle nécessaire à l'assemblage de tout type de PCB. Une relaxation lagrangienne est également proposée pour traiter les instances de grande taille du problème étudié. Plus tard, Tóth et al. (2018) ont abordé le même problème en considérant des lots de PCB de taille et de type différents en présence de machines modulaires. L'objectif est de fournir un équilibrage de la charge de travail et de sélectionner une configuration de machines modulaires appropriée pour chaque type de lot. La fonction objectif minimise le temps de production total, qui inclut le temps de préparation pour le changement de type de lot. Les auteurs ont proposé une approche itérative basée sur un algorithme génétique pour générer la configuration de ligne et une formulation PLNE pour l'équilibrage de ligne. Pereira (2018) s'attaque à un problème MuMALBP qui se pose dans l'industrie textile. Étant donné plusieurs types de marchandises, l'objectif est de trouver un équilibrage de ligne approprié pour chacun d'entre eux. Les auteurs ont proposé une approche hybride, qui combine l'heuristique d'Hoffmann avec un algorithme d'estimation de distribution. L'objectif est de minimiser les coûts liés aux ressources et aux équipements. Berger et al. (1992) considèrent un problème MuMALBP dans un environnement de production flexible. Les auteurs proposent un algorithme de type «branch-and-bound» pour minimiser le nombre de postes de travail. Pour ce faire, ils ont combiné les graphes de précédence de chaque produit en un seul, transformant le problème en un SALBP-1. De la même manière, Nazarian et al. (2010) traitent une ligne hybride multi-produit composée de machines flexibles et non flexibles. Les machines flexibles peuvent être facilement reconfigurées, tandis que les autres sont des machines dédiées qui peuvent exécuter un nombre limité de tâches. Les auteurs proposent un outil d'aide à la décision pour déterminer le nombre de machines flexibles et non flexibles nécessaires, sur la base de la variation de la durée des tâches. Un modèle PLNE et une heuristique, basés sur un seul graphe de précédence combiné, ont été développés pour minimiser le coût d'investissement.

En ce qui concerne la sélection d'équipements et modularité, Graves and Redfield (1988) étudient un problème où une famille de produits similaires doit être assemblée sur une ligne. Les auteurs proposent une méthodologie pour trouver une configuration de ligne unique capable de produire tous les produits et visant à minimiser le coût total, qui comprend les coûts d'investissement liés à l'équipement et les coûts d'exploitation variables. Bukchin and Tzur (2000) développent une approche de type «branch-and-bound» pour traiter le problème de la sélection d'équipements et de l'affectation des tâches aux postes de travail. La fonction objectif minimise le coût total de l'équipement. Plus tard, dans Bukchin and Rubinovitz (2003), les auteurs ont abordé le même problème en considérant la minimisation du nombre de postes de travail avec les coûts d'équipement.

Enfin, le problème le plus proche de celui étudié dans ce chapitre est celui de Belmokhtar et al. (2006). Dans leur travail, les auteurs considèrent une ligne d'usinage modulaire comprenant des unités multi-broches. L'objectif est de concevoir et d'équilibrer une ligne de production unique en sélectionnant et en affectant des unités de broches aux machines en tenant compte de la précédence entre les tâches, du temps de cycle et d'autres contraintes technologiques. Une formulation de programmation linéaire en nombres entiers (PLNE) a été développée pour minimiser le coût d'investissement associé aux unités sélectionnées. Borisovsky et al. (2014) s'attaquent au problème de la conception d'une ligne d'usinage optimale, en tenant

9. MuMALBP – Multi-Model Assembly Line Balancing Problem (en anglais).
10. PCB – Printed Circuit Boards (en anglais).

compte du temps de préparation de la séquence de tâches et de plusieurs contraintes technologiques. Les auteurs proposent de réduire ce problème à un problème de partitionnement de l'ensemble. Cela implique de générer tous les postes de travail possibles composant les machines et de trouver pour chacun d'eux la séquence optimale de tâches qui minimise le temps de préparation total, en utilisant une approche de programmation dynamique. Par la suite, trois approches exactes ont été proposées pour résoudre le problème de partitionnement de l'ensemble : la génération de contraintes, l'algorithme «branch-and-cut» et l'algorithme «branch-and-cut» parallèle. Dans le même domaine, Battaïa et al. (2012) affrontent un problème de localisation d'équipement dans une ligne d'usinage. Les auteurs proposent plusieurs procédures de pré-traitement, basées sur l'analyse des contraintes, afin de réduire l'espace de recherche. En outre, ils ont développé un algorithme pour générer une borne inférieure sur le nombre d'équipements utilisés.

Plusieurs travaux dans la littérature RMS ont étudié le problème de conception de configuration de lignes multi-produits (voir, *e.g.*, Yelles-Chaouche et al., 2021). Dans ce type de problème, un ensemble de machines reconfigurables disponibles est considéré comme connu. De plus, chaque machine peut avoir différentes configurations, chacune lui permettant d'effectuer différentes tâches. Le problème consiste donc à déterminer le nombre de postes de travail et le nombre de machines par poste de travail afin de satisfaire la demande de produits. Chaque poste de travail peut avoir plusieurs machines parallèles avec des configurations différentes. Par exemple, Youssef and ElMaraghy (2006, 2007, 2008) considèrent le problème de conception de configuration de lignes multi-produits avec l'objectif principal de minimiser le coût du capital. Des objectifs similaires sont pris en compte par Dou et al. (2009) et Dou et al. (2016). Ashraf and Hasan (2018) abordent le même problème en tenant compte de la reconfigurabilité, de la capacité opérationnelle et de la fiabilité du système dans la fonction objectif, en plus de la minimisation du coût total. Dans Saxena and Jain (2012), des coûts supplémentaires sont introduits dans la fonction objectif, tels que les coûts de reconfiguration, d'exploitation et de maintenance. Bryan et al. (2013) proposent une approche pour concevoir une ligne d'assemblage reconfigurable tout en tenant compte de l'évolution du produit dans le temps. L'objectif principal consiste à trouver la planification de reconfiguration optimale qui minimise le coût du cycle de vie, calculé sur la base des périodes de production et de reconfiguration.

Dans le contexte du problème de la conception de configurations multi-produits, les approches existantes supposent généralement que les configurations des machines et leurs capacités opérationnelles sont connues et données. Ce n'est pas le cas dans notre travail, où les configurations de machines doivent être conçues pour chaque produit en assignant les modules appropriés. En outre, alors que les études antérieures portent souvent sur des lignes mixtes, où plusieurs types de produits sont fabriqués simultanément, le présent travail porte sur une ligne à modèles multiples où chaque type de produit nécessite une configuration de ligne spécifique. Ainsi, le passage d'un produit à l'autre nécessite une reconfiguration de la ligne. Enfin, contrairement à la présente étude, la littérature existante sur la conception de configurations dans le contexte des RMS considère principalement des lignes à flux non cadencé, où les machines ne sont pas limitées par un temps de cycle spécifique où les produits ne sont pas obligés de passer par toutes les machines. Ces lignes sont particulièrement efficaces pour les scénarios impliquant des produits très mélangés et de faibles volumes de production.

Au vu des différentes études citées ci-dessus, on constate que

- Les problèmes de conception et d'équilibrage de lignes à modèles multiples ne sont abordés que comme cas d'étude industriels, en raison de l'intérêt croissant que suscite aujourd'hui ce type de structure de ligne. Cependant, il y a un manque d'études généralisées de ce type de problème.

- Le problème de l'équilibrage de la ligne et de la sélection/génération d'équipements a été abordé en présence d'un seul type de produit.

Dans notre recherche, nous visons à combler cette lacune en considérant une ligne modulaire générique multi-produit. L'objectif est de trouver le nombre minimum de modules qui peuvent être utilisés pour fabriquer tous les produits issus d'une même famille.

2.3 Formulation PLNB compacte

Étant donné une famille de produits et un ensemble de machines disponibles, chacune disposant d'un nombre limité d'emplacements de modules, la formulation PLNB compacte suivante vise à générer le nombre minimum de modules requis pour fabriquer tous les produits, tout en tenant compte des contraintes de précédence des tâches et de temps de cycle. Le nombre maximal de tâches pouvant être exécutées par un module est également indiqué. Les contraintes d'inclusion et d'exclusion ne sont pas prises en compte dans le présent travail pour les tâches et les modules, mais elles peuvent facilement être incorporées si nécessaire. Par conséquent, toute tâche peut potentiellement être assignée à n'importe quel module et tout module peut être placé dans n'importe quel emplacement disponible au sein d'une machine. Les notations et variables utilisées sont décrites ci-dessous.

Notations :

- V est l'ensemble de toutes les tâches ;
- W est l'ensemble des machines disponibles ;
- P est l'ensemble des produits ;
- M est l'ensemble de tous les modules qui pourraient être crées ;
- $m_{\max}$ est le nombre maximum de modules par machine ;
- $r_{\max}$ est le nombre maximum de tâches par module ;
- $E = \{1, \ldots, |W| \cdot m_{\max}\}$ est l'ensemble des emplacements de module ;
- $E_k = \{(k-1) \cdot m_{\max} + 1, \ldots, k \cdot m_{\max}\}$ est l'ensemble des emplacements de module de la machine $k \in W$;
- $C^{(p)}$ est le temps de cycle correspondant au produit $p \in P$;
- $t_i^{(p)}$ est la durée de la tâche $i \in V$ correspondant au produit $p \in P$;
- $G^{(p)} = (V, A^{(p)})$ est un graphe acyclique orienté représentant les contraintes de précédence pour le produit $p \in P$. Ici, $A^{(p)}$ est l'ensemble des arcs de $G^{(p)}$, où un arc $(i, j) \in A^{(p)}$ signifie que la tâche j doit être assignée soit au même module que la tâche i, soit à un module suivant.

Variables de décision :

- x_{im} est égal à 1 si la tâche $i \in V$ est assignée au module $m \in M$, 0 sinon ;
- $y_{me}^{(p)}$ est égal à 1 si le module $m \in M$ est assigné à l'emplacement $e \in E$ dans la configuration correspondant au produit $p \in P$, 0 sinon ;
- $z_{ime}^{(p)}$ est égal à 1 si la tâche $i \in V$ est assignée au module $m \in M$, qui à son tour est alloué à l'emplacement $e \in E$ dans la configuration correspondant au produit $p \in P$, 0 sinon ;
- s_m est égal à 1 si le module $m \in M$ n'est pas vide, 0 dans le cas contraire.

$$\min \sum_{m \in M} s_m, \quad \text{s.c.} \tag{2.1}$$

$$1 \leq \sum_{m \in M} x_{im} \leq |P|, \quad \forall i \in V \tag{2.2}$$

$$\sum_{e \in E} y_{me}^{(p)} \leq 1, \quad \forall m \in M, \quad \forall p \in P \tag{2.3}$$

$$\sum_{m \in M} y_{me}^{(p)} \leq 1, \quad \forall e \in E, \quad \forall p \in P \tag{2.4}$$

$$\sum_{m \in M} \sum_{e \in E} z_{ime}^{(p)} = 1, \quad \forall i \in V, \quad \forall p \in P \tag{2.5}$$

$$x_{im} \leq s_m, \quad \forall i \in V, \quad \forall m \in M \tag{2.6}$$

$$x_{im} + y_{me}^{(p)} \leq z_{ime}^{(p)} + 1, \quad \forall i \in V, \quad \forall m \in M, \quad \forall e \in E, \quad \forall p \in P \tag{2.7}$$

$$z_{ime}^{(p)} \leq x_{im}, \quad \forall i \in V, \quad \forall m \in M, \quad \forall e \in E, \quad \forall p \in P \tag{2.8}$$

$$z_{ime}^{(p)} \leq y_{me}^{(p)}, \quad \forall i \in V, \quad \forall m \in M, \quad \forall e \in E, \quad \forall p \in P \tag{2.9}$$

$$\sum_{m \in M} \sum_{e \in E} e \cdot z_{ime}^{(p)} \leq \sum_{m \in M} \sum_{e \in E} e \cdot z_{jme}^{(p)}, \quad \forall (i,j) \in A^{(p)}, \quad \forall p \in P \tag{2.10}$$

$$\sum_{e \in E_k} \sum_{m \in M} \sum_{i \in V} t_i^{(p)} \cdot z_{ime}^{(p)} \leq C^{(p)}, \quad \forall k \in W, \quad \forall p \in P \tag{2.11}$$

$$\sum_{i \in V} t_i^{(p)} \cdot x_{im} \leq C^{(p)}, \quad \forall m \in M, \quad \forall p \in P \tag{2.12}$$

$$\sum_{i \in V} x_{im} \leq r_{\max}, \quad \forall m \in M \tag{2.13}$$

$$\left\lceil \frac{|V|}{r_{\max}} \right\rceil \leq \sum_{m \in M} s_m \leq |V| \tag{2.14}$$

$$s_{m+1} \leq s_m, \quad \forall m \in M \setminus \{|M|\} \tag{2.15}$$

$$\sum_{m \in M} y_{m,e+1}^{(p)} \leq \sum_{m \in M} y_{me}^{(p)}, \quad \forall e \in E_k \setminus \{k \cdot m_{\max}\}, \quad \forall k \in W, \quad \forall p \in P \tag{2.16}$$

$$z_{ime}^{(p)} = 0, \quad \forall i \in V, \quad \forall m \in M, \quad \forall e \notin \bigcup_{k \in Q_i^{(p)}} E_k, \quad \forall p \in P \tag{2.17}$$

$$x_{im} \in \{0,1\}, \quad \forall i \in V, \quad \forall m \in M$$

$$y_{me}^{(p)} \in \{0,1\}, \quad \forall m \in M, \quad \forall e \in E, \quad \forall p \in P$$

$$z_{ime}^{(p)} \in \{0,1\}, \quad \forall i \in V, \quad \forall m \in M, \quad \forall e \in E, \quad \forall p \in P$$

$$s_m \in \{0,1\}, \quad \forall m \in M$$

La fonction objectif (2.1) minimise le nombre total de modules non vides. Les expressions (2.2) garantissent que chaque tâche doit être affectée à au moins un et au plus $|P|$ modules. Les contraintes (2.3) et (2.4) stipulent qu'un module doit être attribué à un seul emplacement et qu'un emplacement ne peut pas accueillir plus d'un module, quelle que soit la configuration. Les contraintes (2.5) indiquent que toute tâche doit être attribuée exactement une fois pour chaque configuration de produit. Les inégalités (2.6) précisent qu'un module est considéré comme non vide si au moins une tâche lui est assignée. Les contraintes (2.7), (2.8) et (2.9) imposent qu'un module non vide généré soit affecté à un emplacement dans au moins une configuration. Les relations de précédence des tâches pour chaque configuration sont exprimées par des inégalités (2.10). Les contraintes (2.11) garantissent que la charge de travail effective d'une machine ne dépasse pas le temps de cycle, quelle que soit la configuration du produit. Les contraintes (2.12) évitent de générer des modules dont la charge de travail dépasse le temps de cycle de la configuration à laquelle ils peuvent être affectés. Les inégalités (2.13) imposent le nombre maximum de tâches par module. Les expressions (2.14) sont des contraintes basées sur l'optimalité qui limitent la valeur optimale de la fonction objectif entre son meilleur et son pire scénario. En effet, le meilleur scénario correspond au cas où tous les modules générés sont différents (sans tâches communes) et tous (sauf probablement un) ont le nombre maximum de tâches admissibles. Quant au pire scénario, il se produit lorsque chaque module est également différent, mais qu'une seule tâche lui est assignée. Dans les deux cas, les modules mentionnés sont évidemment communs à toutes les configurations. Les contraintes (2.15) sont des contraintes de rupture de symétrie permettant à un module d'être généré uniquement si le précédent n'est pas vide. Quant aux contraintes (2.16), elles sont similaires à celles de (2.15), mais concernent les emplacements à l'intérieur d'une machine. Enfin, les contraintes (2.17) exigent que chaque tâche soit affectée à un sous-ensemble restreint d'emplacements, défini par ce que l'on appelle l'intervalle d'affectation $Q_i^{(p)} = \left[l_i^{(p)}, u_i^{(p)}\right]$, où $l_i^{(p)}$ (resp. $u_i^{(p)}$) identifie la machine au plus tôt (resp. au plus tard)

à laquelle la tâche i peut être affectée en ce qui concerne le produit p. Grâce à Patterson and Albracht (1975), cet intervalle peut être calculé comme suit :

$$Q_i^{(p)} = \left[\left\lceil \frac{t_i^{(p)} + \sum\limits_{j \in \mathcal{P}_i^{(p)}} t_j^{(p)}}{C^{(p)}} \right\rceil, |W| + 1 - \left\lceil \frac{t_i^{(p)} + \sum\limits_{j \in \mathcal{S}_i^{(p)}} t_j^{(p)}}{C^{(p)}} \right\rceil \right],$$

où $\mathcal{P}_i^{(p)}$ (resp. $\mathcal{S}_i^{(p)}$) est l'ensemble de tous les prédécesseurs (resp. tous les successeurs) de la tâche i par rapport au graphe de précédence $G^{(p)}$ du produit p. Ici, l'idée de calculer l'indice de la machine au plus tôt (resp. au plus tard) pour la tâche i est basée sur la détermination d'une borne inférieure valide sur le nombre de machines nécessaires pour assigner l'ensemble $\mathcal{P}_i^{(p)} \bigcup \{i\}$ (resp. $\mathcal{S}_i^{(p)} \bigcup \{i\}$).

La Figure 2.4 illustre la logique qui sous-tend la formulation PLNB compacte (2.1)-(2.17). En bas de la figure l'ensemble des tâches qui doivent être assignées à un ensemble donné de modules. Ces modules sont ensuite affectés aux machines disponibles, en respectant le graphe de précédence et les contraintes de temps de cycle pour chaque configuration de produit.

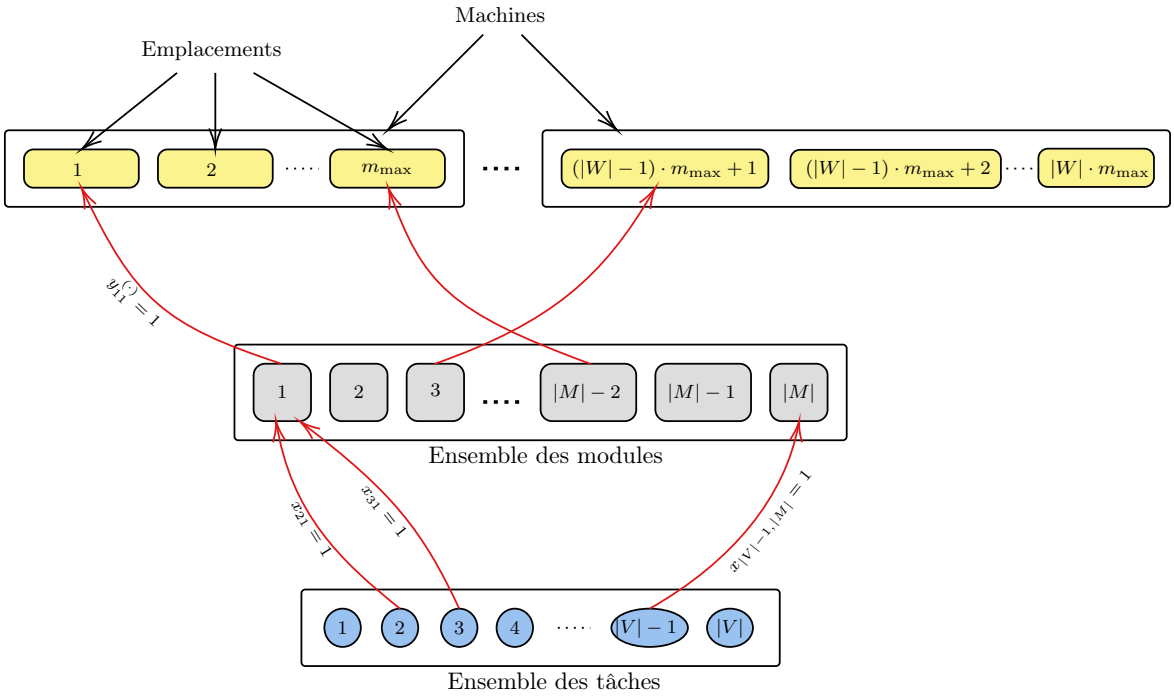

FIGURE 2.4 – Exemple illustrant le fonctionnement du modèle PLNB (2.1)-(2.17)

2.4 Reformulation du modèle PLNB compact

Cette section introduit une approche de décomposition du problème d'optimisation étudié. Pour ce faire, une reformulation du modèle PLNB compact est d'abord présentée. Ensuite, un nouvel algorithme qui génère tous les modules réalisables possibles à partir du graphe de précédence est décrit. Des notations supplémentaires et de nouvelles variables de décision sont expliquées ci-dessous.

Notations complémentaires :

- $M_k^{(p)}$ est l'ensemble des modules qui peuvent être assignés à la machine $k \in W$ dans la configuration correspondant au produit $p \in P$;

- $U_k^{(p)}$ est l'ensemble des paires de modules qui ne peuvent jamais être assignés ensemble à la même machine $k \in W$ dans la configuration correspondant au produit $p \in P$;
- α_{im} est égal à 1 si la tâche $i \in V$ est présente dans le module $m \in M$;
- $c_m^{(p)}$ est la charge correspondant au module $m \in M$ pour le produit $p \in P$.

Nouvelles variables de décision :

- y_m est égal à 1 si le module $m \in M$ est utilisé dans au moins une configuration, 0 sinon ;
- $\lambda_{mk}^{(p)}$ est égal à 1 si le module $m \in M$ est assigné à la machine $k \in W$ dans la configuration correspondant au produit $p \in P$, 0 sinon.

$$\min \sum_{m \in M} y_m, \quad \text{s.c.} \tag{2.18}$$

$$\sum_{k \in W} \sum_{m \in M} \alpha_{im} \cdot \lambda_{mk}^{(p)} = 1, \quad \forall i \in V, \quad \forall p \in P \tag{2.19}$$

$$\sum_{m \in M} c_m^{(p)} \cdot \lambda_{mk}^{(p)} \leq C^{(p)}, \quad \forall k \in W, \quad \forall p \in P \tag{2.20}$$

$$\lambda_{mk}^{(p)} \leq y_m, \quad \forall m \in M, \quad \forall k \in W, \quad \forall p \in P \tag{2.21}$$

$$\sum_{q=k}^{|W|} \sum_{m \in M} \alpha_{im} \cdot \lambda_{mq}^{(p)} \leq \sum_{q=k}^{|W|} \sum_{m \in M} \alpha_{jm} \cdot \lambda_{mq}^{(p)}, \quad \forall (i,j) \in A^{(p)}, \ \forall k \in W, \ \forall p \in P \tag{2.22}$$

$$\sum_{m \in M \setminus M_k^{(p)}} \lambda_{mk}^{(p)} = 0, \ \forall k \in W, \ \forall p \in P \tag{2.23}$$

$$\sum_{m \in M} \lambda_{mk}^{(p)} \leq m_{\max}, \quad \forall k \in W, \quad \forall p \in P \tag{2.24}$$

$$\lambda_{m_1,k}^{(p)} + \lambda_{m_2,k}^{(p)} \leq 1, \quad \forall (m_1, m_2) \in U_k^{(p)}, \quad \forall k \in W, \ \forall p \in P \tag{2.25}$$

$$y_m \in \{0,1\}, \quad \forall m \in M$$

$$\lambda_{mk}^{(p)} \in \{0,1\}, \quad \forall m \in M, \quad \forall k \in W, \quad \forall p \in P$$

La fonction objectif (2.18) minimise le nombre total de modules utilisés. Les égalités (2.19) garantissent que chaque tâche doit être assignée dans chaque configuration. Les contraintes (2.20) indiquent que la somme des durées des tâches effectuées par les modules affectés à une machine ne doit pas dépasser le temps de cycle de chaque produit correspondant. Les inégalités (2.21) expriment qu'un module est considéré comme «utilisé» s'il est affecté à au moins une configuration. Les relations de précédence entre les tâches pour toutes les configurations de produits sont modélisées par (2.22). Les expressions (2.23) sont basées sur $M_k^{(p)}$ qui détermine si un module peut être affecté à une machine ou non. Cela permet de mettre certaines variables de décision à zéro. L'ensemble $M_k^{(p)}$ est calculé sur la base de la technique de l'intervalle d'affectation des tâches, présentée dans la section 2.3. En d'autres termes, $M_k^{(p)} = \{m \in M \mid k \in \bigcap_{i \in m} Q_i^{(p)}\}$. Les contraintes (2.24) garantissent que chaque machine ne peut contenir plus de $m_{\max}$ modules. Enfin, les inégalités (2.25) évitent l'affectation de modules conflictuels à la même machine, en raison des contraintes de précédence des tâches.

Il convient de souligner que cette nouvelle reformulation se débarrasse de l'index des emplacements. Au lieu de cela, les contraintes (2.25) dictent indirectement la séquence d'exécution des modules au sein de chaque machine, car elles empêchent que des modules conflictuels lui soient alloués.

2.4.1 Algorithme de génération de modules

Le processus d'obtention de M consiste à identifier l'ensemble de tous les modules réalisables pour chaque produit p, noté $M^{(p)}$. L'ensemble M est alors formé par l'union de tous les $M^{(p)}$, où $p \in P$, *i.e.*, $M = \bigcup_{p \in P} M^{(p)}$. Ainsi, un module est considéré comme réalisable s'il contient un sous-ensemble de tâches qui n'est pas affecté par les contraintes de précédence et dont la charge n'excède pas le temps de cycle. La Figure 2.5 montre un exemple de modules réalisables et irréalisables par rapport au graphe de précédence présenté dans la partie gauche. Dans cet exemple, le temps de cycle est fixé à 100 et le nombre maximum de tâches par module est de 3. Il est donc facile de remarquer que les modules m_1, m_2, m_3 et m_4 sont des modules réalisables, puisqu'ils respectent les deux contraintes susmentionnées. Cependant, les modules m'_1, m'_2 et m'_3 ne sont pas réalisables en ce qui concerne les contraintes de précédence, le temps de cycle, ou les deux.

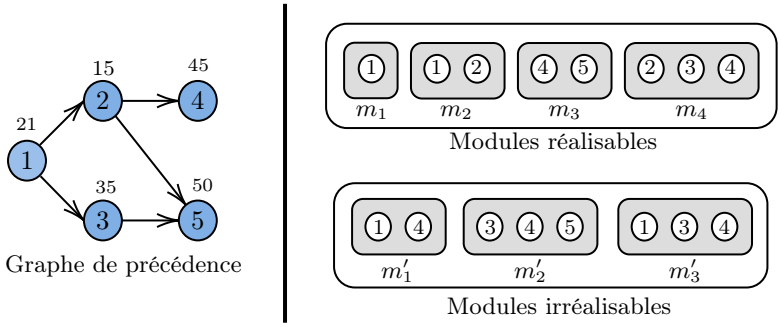

FIGURE 2.5 – Exemple de modules réalisables et irréalisables avec $C = 100$ et $r_{\max} = 3$.

La construction de $M^{(p)}$ consiste à identifier, à partir d'un graphe de précédence donné, tous les sous-graphes dont le nombre de nœuds ne dépasse pas $r_{\max}$ et qui satisfont certaines contraintes supplémentaires détaillées ci-dessous. Pour atteindre cet objectif, un nouvel algorithme, appelé `Gen-Mod`, est mis au point. L'algorithme commence par un ensemble de modules unitaires, composés d'une seule tâche. Ensuite, pour chacun d'entre eux, tous les nouveaux modules possibles sont générés en ajoutant de nouvelles tâches (une à la fois) de sorte que les contraintes de précédence, de capacité et de temps de cycle soient satisfaites. Pour s'assurer que tous les modules réalisables sont générés, la structure de données «pile» est utilisée pour stocker l'état actuel des modules en cours de construction. L'algorithme 2.1 fournit une description plus détaillée de ce processus.

Algorithme 2.1 `Gen-Mod`

Entrée : Un graphe acyclique orienté G, un temps de cycle C et $r_{\max}$.

Sortie : Un ensemble $\mathcal{L}$ de modules générés.

1. Considérer une pile vide $\mathcal{X}$ de modules et fixer $\mathcal{L} := \emptyset$ et $k := 1$.
2. Générer un nouveau module composé uniquement de la tâche k et l'empiler dans $\mathcal{X}$.
3. Si $\mathcal{X}$ n'est pas vide, dépiler un nouveau module m de $\mathcal{X}$. Sinon, passez à l'étape 6.
4. Ajouter m à $\mathcal{L}$. Si $|m| < r_{\max}$, passer à l'étape 5. Sinon, si $|m| = r_{\max}$, passer à l'étape 3.
5. Soit j la dernière tâche ajoutée à m. Pour chaque $i > j$, générer un nouveau module $m \cup \{i\}$ et l'empiler dans $\mathcal{X}$, si sa charge n'excède pas le temps de cycle, *i.e.*, $\sum_{q \in m} t_q \leq C - t_i$, et l'intersection entre tous les successeurs de m et tous les prédécesseurs de i sans tenir compte de $m \cup \{i\}$ est vide, *i.e.*, $\left(\left(\bigcup_{q \in m} \mathcal{S}_q\right) \bigcap \mathcal{P}_i\right) \setminus \left(m \cup \{i\}\right) = \emptyset$. Pour cette dernière condition, voir le Lemme 2.1. Passer à l'étape 3.
6. Si $k < |V|$, mettre $k := k + 1$ et passer à l'étape 2. Sinon, arrêter. L'ensemble $\mathcal{L}$ des modules est généré.

Lemme 2.1. *Soit m un module réalisable. Alors, tout module $m \cup \{i\}$ satisfaisant la contrainte de temps de cycle est considéré comme réalisable en termes de contraintes de précédence ssi la condition suivante est remplie :*

$$\left(\left(\bigcup_{q \in m} \mathcal{S}_q\right) \bigcap \mathcal{P}_i\right) \setminus \left(m \cup \{i\}\right) = \emptyset. \tag{2.26}$$

Démonstration. ($\Rightarrow$) Supposons le contraire. Soit $m \cup \{i\}$ un module réalisable qui ne satisfait pas (2.26). Cela signifie qu'il existe une tâche $j \notin m \cup \{i\}$ qui est à la fois l'un des prédécesseurs de la tâche i et l'un des successeurs des tâches composant m. Ce dernier point indique que j ne peut être exécuté ni avant ni après l'achèvement des tâches appartenant au module $m \cup \{i\}$. Par conséquent, j doit être dans $m \cup \{i\}$, ce qui est en contradiction avec la supposition.

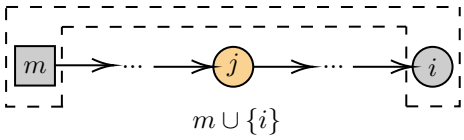

FIGURE 2.6 – Illustration simple d'un module irréalisable

($\Leftarrow$) Supposons le contraire. À savoir, qu'un module $m \cup \{i\}$ satisfasse l'équation (2.26), mais qu'il soit irréalisable. Cela nous conduira nécessairement à la situation décrite dans la Figure 2.6 et, par conséquent, à la non-satisfaction de (2.26). Cela contredit la supposition. □

2.4.2 Génération de contraintes

Le traitement de l'ensemble de la reformulation (2.18)-(2.25) peut s'avérer difficile en raison de la présence des contraintes (2.25). En effet, l'énumération et l'utilisation de toutes les inégalités (2.25) peuvent entraîner une augmentation significative du nombre total de contraintes et rendre la reformulation (2.18)-(2.25) inefficace, car toute solution réalisable de (2.18)-(2.24) ne viole généralement que quelques-unes des inégalités (2.25).

Ainsi, pour éviter l'inefficacité de l'énumération de toutes les contraintes (2.25), une approche algorithmique plus appropriée est proposée. Cette approche est basée sur le cadre de génération de contraintes, généralement fourni par tout solveur commercial moderne, et consiste à n'ajouter que les contraintes violées (2.25) au cours du processus de résolution, tout en ne résolvant que la reformulation (2.18)-(2.24). Les solveurs commerciaux peuvent gérer cela efficacement, ce qui en fait une solution appropriée pour ce type de problème.

Plus précisément, lorsqu'une nouvelle meilleure solution entière est trouvée par le solveur, l'analyse suivante est appliquée. Si, pour cette solution, il existe un produit p, une machine k et deux modules m_1 et m_2 tels que $(m_1, m_2) \in U_k^{(p)}$, $\lambda_{m_1,k}^{(p)} = 1$ et $\lambda_{m_2,k}^{(p)} = 1$, alors ajouter dans la reformulation (2.18)-(2.24) la contrainte $\lambda_{m_1,k}^{(p)} + \lambda_{m_2,k}^{(p)} \leq 1$ en utilisant les routines `LazyConstraints` et `UserCuts`, et continuer à résoudre la reformulation. Sinon, la solution entière trouvée est une solution optimale.

2.5 Résultats numériques

Cette section présente les résultats expérimentaux des deux approches développées. Il s'agit de la formulation compacte (2.1)-(2.17) et de sa reformulation (2.18)-(2.25). Les expériences ont été menées avec ILOG CPLEX 12.10.0 installé sur un ordinateur Intel(R) Core(TM) i7-8650U à 1.90 GHz avec 32 Go de RAM. Les instances de données de référence bien connues de Otto et al. (2013) ont été utilisées [11]. La formulation compacte a été testée uniquement sur un ensemble d'instances de petite taille avec $|V| = 20$, tandis que deux ensembles d'instances de petite et moyenne taille, avec $|V| = 20$ et $|V| = 50$, ont été respectivement utilisés pour la reformulation. Cette décision est basée sur l'observation que la formulation compacte présentait des performances médiocres pour les instances de petite taille et qu'il n'était donc

11. http://pagesperso.ls2n.fr/~gurevsky-e/data/RMS.zip

pas nécessaire de la tester sur des instances plus grandes. En outre, la célèbre heuristique COMSOAL, développée par Arcus (1965), a été employée pour fournir à CPLEX une solution réalisable à chaud pour chaque instance. Pour ce faire, la méthode COMSOAL classique a été appliquée à chaque produit $p \in P$ séparément afin de construire sa configuration réalisable, en considérant simplement toute tâche comme un module de taille un. Il en résulte une solution réalisable dont le nombre de modules est égal au nombre de tâches. La raison de l'ajout de cet algorithme est que CPLEX n'a pas été en mesure de trouver une solution réalisable pour certaines instances complexes dans le délai de résolution de 600 secondes.

Toutes les instances ont été regroupées en trois séries distinctes sur la base de la densité de leur graphe de précédence, qui est mesurée par sa force d'ordre (OS), définie comme $\frac{2 \cdot |E^{(p)}|}{|V| \cdot (|V|-1)}$, où $E^{(p)} = \left\{(i,j) \mid i \in V \text{ et } j \in \mathcal{S}_i^{(p)}\right\}$. Une valeur OS plus élevée indique un graphe de précédence plus dense. Les trois séries correspondent à des instances ayant un $OS \approx 0.2$, $OS \approx 0.6$, et $OS \approx 0.9$ (comme le montre la Table 2.1). Pour construire une instance avec trois produits, il suffit de sélectionner trois instances consécutives de la même série.

Table 2.1 – Enumération de trois séries

Série	1	2	3
OS	≈ 0.2	≈ 0.6	≈ 0.9

En outre, le temps de cycle $C^{(p)}$ correspondant au produit $p \in P$ est fixé à $\left\lceil 1.5 \cdot \max_{i \in V} t_i^{(p)} \right\rceil$, tandis que le nombre de machines et le nombre maximal de modules par machine sont respectivement fixés à

$$|W| = \max_{p \in P} \left\{ \left\lceil 1.2 \cdot \sum_{i \in V} t_i^{(p)} / C^{(p)} \right\rceil \right\} \quad \text{et} \quad m_{\max} = \max_{p \in P} \max \left\{ k \mid \sum_{i=1}^{k} t_{\pi_i}^{(p)} \leq C^{(p)} \right\},$$

où $(\pi_1, \pi_2, \ldots, \pi_{|V|})$ est une permutation de V selon l'ordre non décroissant de la durée des tâches $t_i^{(p)}$, où $i \in V$ et $p \in P$.

2.5.1 Comparaison des formulations

Les résultats expérimentaux pour des instances de petite taille avec $|V| = 20$, en utilisant la formulation PLNB compacte et sa reformulation, sont présentés dans les Tables 2.2 et 2.3, respectivement. Dans les deux tables, la première colonne montre le nombre de produits, tandis que les deuxième et troisième colonnes indiquent le nombre maximum de tâches par module et la série OS, respectivement. La quatrième colonne affiche le nombre total d'instances dans chaque série. Le nombre d'instances résolues à l'optimalité et leur temps moyen `CPU` sont donnés dans les cinquième et sixième colonnes, respectivement. Enfin, la dernière colonne montre la moyenne du `GAP` pour les instances restantes qui n'ont pas été résolues à l'optimalité dans la limite du temps de résolution fixé à 600 secondes. Le calcul de `GAP` consiste à comparer la valeur de la fonction objectif de la meilleure solution trouvée (`UB`) à la valeur de la borne inférieure (`LB`) déterminée par CPLEX. Elle s'exprime comme suit $\texttt{GAP} = 100\% \cdot (\texttt{UB} - \texttt{LB})/\texttt{UB}$.

En analysant les résultats présentés dans la Table 2.2, il est évident de voir que la formulation PLNB compacte est inefficace, avec seulement 11% des instances résolues à l'optimalité et un `GAP` relativement élevé pour les 89% d'entre elles restantes. Une autre observation remarquable est que lorsque le OS des instances augmente, il en va de même pour la moyenne du `GAP`. Ceci est principalement attribué au nombre croissant de contraintes de précédence, qui alourdit le modèle et rend difficile pour CPLEX d'améliorer les bornes supérieures et inférieures. Par conséquent, sur la base de ces résultats, il a été jugé inutile de tester la formulation PLNB compacte sur des instances de taille plus grande.

Contrairement à la formulation PLNB compacte, sa reformulation présente de meilleures performances sur les mêmes instances. Les résultats obtenus, présentés dans la Table 2.3, indiquent que 98.71% des instances correspondant à $|P| = 2$ ont été résolues à l'optimalité en moins d'une minute. Pour ces instances, le nombre de modules générés pour $r_{\max} = 2$ (resp. $r_{\max} = 3$) varie de 79 à 192 (resp. 81 à

Table 2.2 – Résultats computationnels pour la formulation PLNB compacte pour $|V| = 20$.

$\|P\|$	$r_{\max}$	OS	#Instances	#OPT	Moy. CPU, (s.)	Moy. GAP, (%)
2	2	1	3	3	177.77	⊕
		2	199	74	293.05	20.86
		3	68	10	331.32	21.34
	3	1	3	2	11.21	10.00
		2	199	29	222.50	23.97
		3	68	4	251.27	25.57
3	2	1	2	0	⊖	21.20
		2	193	1	568.00	41.60
		3	65	0	⊖	41.90
	3	1	2	0	⊖	43.70
		2	193	0	⊖	52.04
		3	65	0	⊖	51.24

(⊕) Toutes les solutions optimales ont été trouvées dans le délai de 600 secondes.
(⊖) Aucune solution optimale n'a été trouvée dans le délai de 600 secondes.

722), avec une moyenne de 138.43 (resp. 365.95). Comme prévu, le nombre de modules générés augmente avec $r_{\max}$ et le nombre de produits, ce qui rend le problème plus difficile. Cela peut être observé dans les résultats, où moins d'instances avec $|P| = 3$ ont été résolues à l'optimalité. Cependant, la reformulation a pu atteindre la solution optimale pour 92.12% des instances. Pour les presque 8% restants, la moyenne du GAP obtenue est relativement faible. Des observations similaires peuvent être faites pour les instances avec $|P| = 4$, où la reformulation a trouvé des solutions optimales pour 79.08% des instances dans un temps CPU raisonnable.

Table 2.3 – Résultats computationnels de la reformulation pour $|V| = 20$.

$\|P\|$	$r_{\max}$	OS	#Instances	#OPT	Moy. CPU, (s.)	Moy. GAP, (%)
2	2	1	3	3	1.72	⊕
		2	199	199	8.36	⊕
		3	68	68	2.39	⊕
	3	1	3	3	2.93	⊕
		2	199	193	51.66	16.28
		3	68	68	11.03	⊕
3	2	1	2	2	1.25	⊕
		2	193	190	48.47	9.10
		3	65	65	11.57	⊕
	3	1	2	2	5.32	⊕
		2	193	158	106.08	28.15
		3	65	64	20.32	11.10
4	2	1	1	1	3.95	⊕
		2	188	174	64.45	14.98
		3	62	62	52.36	⊕
	3	1	1	1	28.23	⊕
		2	188	101	128.51	34.64
		3	62	58	94.46	22.78

(⊕) Toutes les solutions optimales ont été trouvées dans le délai de 600 secondes.

Lorsqu'il s'agit d'instances de taille moyenne avec $|V| = 50$, la reformulation atteint ses limites, comme le montre la Table 2.4. Sur les 614 instances avec $|P| = 2$ considérées dans les expériences, seules 33 ont été résolues de manière optimale par le solveur, tandis que les autres sont restées non résolues dans le temps imparti. De plus, la moyenne relativement élevée du `GAP` pour ces instances non résolues suggère qu'elles étaient difficiles à traiter pour le solveur. Cette difficulté peut être attribuée au plus grand nombre de contraintes et de variables de décision résultant d'un plus grand ensemble de modules générés réalisables, où le nombre moyen de modules générés pour les instances avec $|P| = 2$ (resp. $|P| = 3$) est de 2039.25 (resp. 2512.76) allant de 266 à 15854 (resp. 311 à 17565) modules.

TABLE 2.4 – Résultats computationnels de la reformulation pour $|V| = 50$.

$\|P\|$	$r_{\max}$	OS	#`Instances`	#`OPT`	Moy. `CPU`, (s.)	Moy. `GAP`, (%)
2	2	1	15	0	⊖	50.00
		2	218	8	241.86	37.23
		3	74	10	348.74	15.61
	3	1	15	0	⊖	59.70
		2	218	5	388.86	57.86
		3	74	10	270.91	36.89
3	2	1	14	0	⊖	50
		2	218	0	⊖	48.68
		3	73	1	35.76	31.41
	3	1	14	0	⊖	74.49
		2	218	0	⊖	59.68
		3	73	0	⊖	45.84

(⊖) Aucune solution optimale n'a été trouvée dans le délai de 600 secondes.

2.6 Conclusion et perspectives

Alors que la recherche existante sur les lignes modulaires s'est principalement concentrée sur la conception de lignes de produits uniques, ce travail propose une solution pour la conception de lignes multi-produits composées de machines modulaires. Chaque machine dispose d'un nombre limité d'emplacements pour l'installation de modules, et les modules spécifiques requis pour chaque produit ne sont pas connus à l'avance. L'objectif est de concevoir une configuration de ligne pour chaque produit qui respecte les contraintes de temps de cycle et de précédence des tâches tout en minimisant le nombre total de modules utilisés. Ce chapitre présente une formulation PLNB compacte qui assigne les tâches aux modules et les alloue ensuite aux machines disponibles, ainsi qu'une reformulation basée sur une approche de décomposition. Cette dernière consiste à générer l'ensemble de tous les modules réalisables de taille limitée à partir d'un graphe de précédence, puis à utiliser la reformulation pour sélectionner le nombre optimal de modules à partir de cet ensemble. Les résultats des calculs montrent que la formulation PLNB compacte n'est pas efficace pour résoudre même les instances de petite taille, alors que sa reformulation est capable de résoudre les mêmes instances jusqu'à l'optimalité en quelques secondes. Une des pistes de recherche futures pourrait être d'étudier la relaxation continue de la reformulation (2.18)-(2.25), traitée par génération de colonnes, afin d'analyser la qualité de la borne supérieure ainsi obtenue.

Les résultats de ce chapitre sont issus de la thèse de doctorat d'Abdelkrim R. Yelles-Chaouche. Ils sont également publiés dans le journal CIE [12] (voir Yelles-Chaouche et al., 2024). Dans ce même article, une méthode heuristique basée sur la reformulation (2.18)-(2.25) ainsi que sur une technique de filtrage des modules a aussi été proposée afin de traiter plus rapidement et plus efficacement les instances de taille moyenne/grande du problème étudié. De plus, dans le cadre de cette même thèse, d'autres résultats ont

12. CIE – Computers & Industrial Engineering, édité par le groupe Elsevier B.V.

également été obtenus et publiés dans les journaux CIE (voir Yelles-Chaouche et al., 2022) et JMS [13] (voir Tremblet et al., 2023), ce dernier en collaboration avec David Tremblet (IMT Atlantique, Nantes). Ainsi, Yelles-Chaouche et al. (2022) et Tremblet et al. (2023) ont abordé le problème d'équilibrage d'une ligne reconfigurable multi-produit, où l'objectif était de minimiser le nombre total de tâches réaffectées lors du passage d'une configuration de produit à une autre. Plus précisément, Yelles-Chaouche et al. (2022) ont étudié le cas où la séquence des lots de produits est donnée, tandis que Tremblet et al. (2023) ont étudié le scénario où aucune information sur cette séquence n'est fournie. Dans les deux contributions, une formulation PLNB ainsi qu'une approche heuristique basée sur cette formulation ont été développées pour traiter le problème correspondant. Il est également important à noter que les résultats de cette thèse ont été présentés lors de 8 conférences internationales/nationales (voir l'annexe A pour plus de détails), sans oublier un article d'état de l'art sur la revue des problèmes d'optimisation dans le domaine des RMS, publié dans le journal IJPR [14] (voir Yelles-Chaouche et al., 2021).

Après sa thèse et un an de post-doctorat à l'École des Mines de Saint-Étienne à Gardanne, Abdelkrim R. Yelles-Chaouche, quant à lui, est actuellement Ingénieur des Achats chez Festo à Marseille.

13. JMS – Journal of Manufacturing Systems, édité par le groupe Elsevier B.V.
14. IJPR – International Journal of Production Research, édité par le groupe Taylor & Francis.

Chapitre 3

Problèmes de gestion du risque avec données contrôlables

Sommaire

3.1 Introduction **39**
3.2 Réduction à une série de problèmes plus simples **41**
3.3 Quelques cas polynomiaux **42**
3.3.1 Problème Min-Risk-Span 42
3.3.2 Problème Min-Risk-Path 43
3.4 Problèmes reverses **43**
3.5 Conclusion et perspectives **45**

3.1 Introduction

La minimisation du risque est l'un des principaux objectifs de la prise de décision. Dans ce chapitre, nous étudions ce type de problèmes dans des applications telles que le routage et la conception de réseaux fiables où la décision comprend un chemin dans un graphe acyclique orienté ou bien un arbre dans un graphe non orienté, et les quantités de ressources allouées aux arcs du chemin ou aux arêtes de l'arbre. Plus la quantité de ressources est importante, plus le risque de défaillance de l'élément correspondant est faible. L'objectif est donc de minimiser le risque total.

Soit $E = \{1, \ldots, m\}$ un ensemble d'éléments arbitraires, et soit $\mathcal{S}(E)$ une famille spécifique de sous-ensembles non vides de E, que nous appelons *structures*. Par exemple, E peut être l'ensemble des *arcs* d'un *graphe acyclique orienté* $G = (V, E)$, $E \subseteq \{(i, j) \mid i \in V, j \in V\}$, où V est l'ensemble des *nœuds*, et $\mathcal{S}(E)$ peut être l'ensemble de tous les *chemins* entre deux nœuds spécifiés de ce graphe. Chaque élément $e \in E$ est associé à une variable de ressource à valeur entière x_e, à des bornes inférieure et supérieure entières non négatives c_e et d_e sur cette variable, $c_e < d_e$, et à une *fonction de risque* $f_e(x_e)$.

Nous appelons les paires (S, x) *solutions*, où $S \in \mathcal{S}(E)$ est une structure et x est une collection de variables x_e, $e \in S$. L'hypothèse suivante est utilisée ici.

Hypothèse 3.1. *Toutes les fonctions de risque f_e, $e \in E$, sont non croissantes, concaves et satisfont aux égalités $f_e(c_e) = k_1$ et $f_e(d_e) = k_2$, où k_1 et k_2 sont des nombres donnés tels que $k_1 > k_2$. Les valeurs k_1 et k_2 sont les mêmes pour tout $e \in E$.*

Ainsi, une valeur de risque maximale k_1 est atteinte pour l'élément e, lorsque la quantité de ressource x_e est à sa borne inférieure c_e, et une valeur de risque minimale k_2 est atteinte lorsque x_e est à sa borne supérieure d_e. Soit B une borne supérieure entière sur la quantité totale disponible de la ressource. Le problème étudié ici peut être donc formulé comme suit.

Problème Min-Risk :

$$\min_{(S,x)} \sum_{e\in S} f_e(x_e), \quad \text{s.c.}$$
$$\sum_{e\in S} x_e \le B,$$
$$S \in \mathcal{S}(E),$$
$$x_e \in \{c_e, c_e+1, \dots, d_e\},\ e \in S.$$

Notre analyse plus approfondie montrera la validité du commentaire suivant.

Commentaire 3.1. *Tous les résultats pour le problème* Min-Risk*, obtenus ici, s'appliquent pour des variables réelles x_e, $e \in E$, et des contraintes $c_e \le x_e \le d_e$, $e \in E$.*

La motivation initiale de nos études est le problème introduit par Chen et al. (2009a,b). Nous désignons ce dernier problème par Min-Linear-Risk.

Problème Min-Linear-Risk :

$$\min_{(S,x)} \sum_{e\in S} \frac{d_e - x_e}{d_e - c_e}, \quad \text{s.c.}$$
$$\sum_{e\in S} x_e \le B,$$
$$S \in \mathcal{S}(E),$$
$$x_e \in \{c_e, c_e+1, \dots, d_e\},\ e \in S.$$

Dans Chen et al. (2009a,b), E est l'ensemble des arêtes (ou arcs) d'un graphe non orienté (resp. acyclique orienté), et $\mathcal{S}(E)$ est l'ensemble des arbres couvrants (resp. chemins entre deux nœuds spécifiés) du graphe correspondant. Les auteurs considèrent la fonction $f_e(x_e) = \frac{d_e - x_e}{d_e - c_e}$ comme la valeur du risque d'attribuer le poids x_e à l'arête (resp. l'arc) e dans le cadre de la conception de réseaux (resp. du routage). Dans le cadre du routage, x_e est considéré comme le temps de trajet via le lien e d'un réseau, et $f_e(x_e)$ comme le risque d'échec (accident) du déplacement via ce lien. L'objectif est de trouver un chemin entre deux nœuds spécifiés et de déterminer le temps de trajet pour chaque lien de ce chemin, qui minimise le risque total, à condition que le temps de trajet total ne dépasse pas une limite supérieure donnée B. Dans le cadre de la conception d'un réseau, x_e est considéré comme le coût d'établissement d'un lien entre les deux nœuds de l'arête e, et $f_e(x_e)$ comme le risque de défaillance de ce lien. L'objectif est de trouver un réseau de communication, c'est-à-dire un arbre couvrant, et de déterminer le coût d'établissement d'un lien pour chaque arête de cet arbre, qui minimise le risque total, à condition que le coût total ne dépasse pas une borne supérieure donnée B. Le problème Min-Linear-Risk est un cas particulier du problème Min-Risk. Les fonctions $f_e(x_e) = \frac{d_e - x_e}{d_e - c_e}$ satisfont l'hypothèse 3.1 avec $k_1 = 1$ et $k_2 = 0$.

Les problèmes de minimisation du risque dans les applications de réseau ont été récemment étudiés par Bayrak et al. (2019), Bozkaya et al. (2017), Richards et al. (2018) et Le Carrer et al. (2020). Les problèmes avec des paramètres d'entrée contrôlables (dépendant des ressources) ont été étudiés pour les problèmes d'ordonnancement par Cheng et al. (1998, 2006), Grigoriev et al. (2007), Janiak and Kovalyov (1996), Janiak et al. (2005), Ng et al. (2003), Shabtay and Steiner (2007) et Shioura et al. (2015).

Le présent chapitre est organisé comme suit. Dans la section suivante, nous présentons une réduction du problème Min-Risk à une série de problèmes plus faciles à résoudre. La section 3.3 présente des algorithmes en temps polynomial pour plusieurs cas particuliers du problème Min-Risk basés sur notre réduction. Ils sont asymptotiquement meilleurs pour tous les cas de Chen et al. (2009a,b). Dans la section 3.4, nous étudions une version *reverse* du problème Min-Risk, dénommée Rev-Min-Risk. Le problème reverse diffère du problème original en ce que les rôles de la fonction objectif et de la contrainte de type de ressource sont échangés, c'est-à-dire que l'opérateur original «min» est remplacé par l'opérateur «$\le C$», et l'opérateur original «$\le B$» est remplacé par l'opérateur «min». Des cas spéciaux linéaires de ce problème

ont été étudiés par Álvarez-Miranda et al. (2011). Comme pour le problème original, nous proposons une réduction du problème reverse en une série de problèmes plus faciles et, sur cette base, nous développons des algorithmes en temps polynomial pour les cas spéciaux. Ces algorithmes sont asymptotiquement plus rapides que ceux de Álvarez-Miranda et al. (2011).

3.2 Réduction à une série de problèmes plus simples

Dans cette section, nous exposons de manière synthétique quelques propriétés des solutions optimales du problème Min-Risk, puis nous utilisons ces propriétés dans la réduction de ce problème à une série de problèmes plus faciles à résoudre. Il est important de noter pour la suite que les détails complets de toutes les preuves peuvent être trouvés dans Gurevsky et al. (2022a).

Le théorème suivant est une généralisation du Lemme 1 de Chen et al. (2009a,b), qui, à son tour, est une généralisation du résultat de Dantzig (1957), au cas des fonctions concaves non croissantes f_e.

Théorème 3.1 (Gurevsky et al. (2022a)). *Si le problème* Min-Risk *a une solution, alors il existe une solution optimale* (S^*, x^*) *de ce problème, dans lequel il y a au plus un élément* $e^* \in S^*$ *tel que toutes les variables sauf la variable* $x^*_{e^*}$ *sont égales à leurs bornes inférieures ou supérieures :* $x^*_e \in \{c_e, d_e\}$, $e \in S^* \backslash \{e^*\}$, *et* $x^*_{e^*} = \min\left\{d_{e^*}, B - \sum_{e \in S^* \backslash \{e^*\}} x^*_e\right\}$.

Il découle du Théorème 3.1 que, si le problème Min-Risk a une solution, alors il existe une solution optimale (S^*, x^*) telle que $\sum_{e \in S^* \backslash \{e^*\}} f_e(x^*_e) = i^* \cdot k_1 + j^* \cdot k_2$, où i^* est le nombre de variables $x^*_e = c_e$ et j^* est le nombre de variables $x^*_e = d_e$, $e \in S^* \setminus \{e^*\}$. Soit K l'ensemble qui contient toutes ces paires (i^*, j^*). Il est facile de voir que l'on peut prendre

$$K = \Big\{(i,j) \mid i + j \leq s_{\max} - 1,\ i = 0, 1, \ldots, s_{\max} - 1,\ j = 0, 1, \ldots, s_{\max} - 1\Big\},$$

où $s_{\max}$ est la cardinalité de la plus grande structure dans $\mathcal{S}(E)$. La cardinalité de l'ensemble K ci-dessus est $|K| = s_{\max} \cdot (s_{\max} + 1)/2$.

Considérons un élément donné $e_0 \in E$ et une paire donnée $(i, j) \in K$, qui peuvent être considérés comme des «suppositions» telles qu'il existe une solution optimale (S, x) du problème Min-Risk, dans lequel $x_e \in \{c_e, d_e\}$ pour $e \in S \setminus \{e_0\}$, i est le nombre de variables $x_e = c_e$ et j est le nombre de variables $x_e = d_e$, $e \in S \setminus \{e_0\}$. Introduisons le problème auxiliaire suivant.

Problème $P(e_0, i, j)$:

$$\min_{(S,x)} \sum_{e \in S \backslash \{e_0\}} x_e, \quad \text{s.c.}$$

$$e_0 \in S,\ S \in \mathcal{S}(E),$$

$$\Big|\{e \in S \setminus \{e_0\} \mid x_e = c_e\}\Big| = i,\ \Big|\{e \in S \setminus \{e_0\} \mid x_e = d_e\}\Big| = j,$$

$$x_e \in \{c_e, d_e\},\ e \in S \setminus \{e_0\}.$$

Nous désignons une solution optimale du problème $P(e_0, i, j)$ par $(S^{(e_0,i,j)}, x^{(e_0,i,j)})$, si elle existe. Le théorème suivant montre une relation entre une solution optimale de Min-Risk et celle du problème auxiliaire.

Théorème 3.2 (Gurevsky et al. (2022a)). *Si le problème* Min-Risk *a une solution, alors il existe une solution optimale* $(\hat{S}^*, \hat{x}^*)$ *de ce problème telle que* $\hat{S}^* = S^{(e^*,i^*,j^*)}$, $\hat{x}^*_e = x^{(e^*,i^*,j^*)}_e$ *pour* $e \in \hat{S}^* \setminus \{e^*\}$ *et* $\hat{x}^*_{e^*} = \min\left\{d_{e^*}, B - \sum_{e \in S^{(e^*,i^*,j^*)} \backslash \{e^*\}} x^{(e^*,i^*,j^*)}_e\right\}$, *pour certain* $e^* \in E$ *et certain* $(i^*, j^*) \in K$.

Sur la base du Théorème 3.2, la procédure d'énumération suivante peut être employée pour résoudre le problème Min-Risk.

Procédure `Enum`.

1. Résoudre les $m \cdot |K|$ problèmes $P(e_0, i, j)$, $e_0 \in E$, $(i, j) \in K$. Soit Q l'ensemble de solutions optimales de ces problèmes. Notons qu'il est possible que certains problèmes $P(e_0, i, j)$ n'aient pas de solution.
2. Pour chaque solution $(S^{(e_0,i,j)}, x^{(e_0,i,j)}) \in Q$ satisfaisant $\sum_{e \in S^{(e_0,i,j)} \setminus \{e_0\}} x_e^{(e_0,i,j)} \leq B - c_{e_0}$, construire une *solution étendue* $(S^{(e_0,i,j)}, \hat{x}^{(e_0,i,j)})$ telle que $\hat{x}_e^{(e_0,i,j)} = x_e^{(e_0,i,j)}$ pour $e \in S^{(e_0,i,j)} \setminus \{e_0\}$ et $\hat{x}_{e_0}^{(e_0,i,j)} = \min \left\{ d_{e_0}, B - \sum_{e \in S^{(e_0,i,j)} \setminus \{e_0\}} x_e^{(e_0,i,j)} \right\}$. Soit Q_E l'ensemble des solutions étendues.
3. Choisir une solution étendue $(S^{(e_0,i,j)}, \hat{x}^{(e_0,i,j)}) \in Q_E$ avec la valeur minimale de $i \cdot k_1 + j \cdot k_2 + f_{e_0}(\hat{x}_{e_0}^{(e_0,i,j)})$. Elle est optimale pour le problème Min-Risk.

Si tous les problèmes $P(e_0, i, j)$, $e_0 \in E$, $(i, j) \in K$, peuvent être résolus en temps $O(T)$, alors la procédure `Enum` peut être implémentée pour s'exécuter en temps $O(T)$ également. Si tous les problèmes $P(e_0, i, j)$, $(i, j) \in K$, peuvent être résolus en temps $O(D)$ pour tout $e_0 \in E$, alors la procédure `Enum` peut être implémentée pour s'exécuter en temps $O(mD)$. Dans la section suivante, nous considérons des cas particuliers du problème Min-Risk pour lequel nous présentons des algorithmes en temps polynomial pour résoudre tous les problèmes $P(e_0, i, j)$ pour tous les $(i, j) \in K$ et un $e_0 \in E$ donné, ou pour tous les $(i, j) \in K$ et tous les $e_0 \in E$.

3.3 Quelques cas polynomiaux

Dans cette section, nous présentons des algorithmes polynomiaux pour plusieurs cas particuliers du problème Min-Risk.

3.3.1 Problème Min-Risk-Span

Dans le problème Min-Risk-Span, les éléments sont les arêtes et les structures sont les *arbres couvrants* d'un graphe non orienté $G = (V, E)$, $|V| = n$, $|E| = m$. Rappelons qu'un arbre couvrant $T = (V, E_T)$, $E_T \subseteq E$, est un sous-graphe de G qui inclut tous les nœuds de G, et c'est un arbre. Par la définition de l'arbre couvrant, $|E_T| = n - 1$. Par conséquent, pour le problème Min-Risk-Span, $K = \{(i, n-2-i) \mid i = 0, 1, \ldots, n-2\}$ et $|K| = n - 1$.

Pour $e_0 \in E$ et $(i, n-2-i) \in K$ donnés, chaque problème $P(e_0, i, n-2-i)$ peut être résolu par l'approche suivante. Tout d'abord, construire un *graphe* $G^* = (V^*, E^*)$, qui diffère du graphe original $G = (V, E)$ en ce sens qu'il y a une copie supplémentaire de chaque arête de E mais que l'arête e_0 et les arêtes parallèles à e_0 sont *contractées* en un seul nœud v_0 de sorte que les autres arêtes incidentes aux extrémités de e_0 dans G deviennent incidentes à v_0 dans G^*, voir les Figures 3.1 et 3.2 pour une illustration.

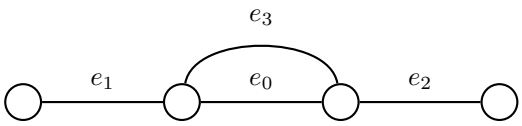

Figure 3.1 – Un sous-graphe de G contenant e_0

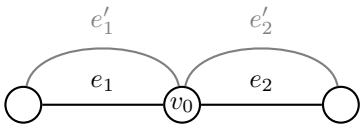

Figure 3.2 – Un sous-graphe de G^* contenant v_0

Considérons une paire d'arêtes e et e', où e est une arête originale de $E \setminus \{e_0\}$ et e' est sa copie dans E^*. Nous appelons l'arête e *rouge* et lui associons le poids $w_e = c_e$. On appelle l'arête e' *blanc* et on lui associe le poids $w_{e'} = d_e$.

Théorème 3.3 (Gurevsky et al. (2022a)). *Le problème $P(e_0, i, n-2-i)$ sur le graphe G se réduit au problème de trouver un arbre couvrant de poids minimum avec exactement i arêtes rouges sur le graphe G^*, que nous dénotons par* Span-i-Red.

Gabow and Tarjan (1984) (Théorème 4.1, p. 97) ont développé un algorithme en $O(tree(m,n)+n\log n)$ temps pour résoudre le problème SPAN-i-RED, où $tree(m,n)$ est le temps de résolution du problème classique de l'arbre couvrant de poids minimum dans un graphe avec m arêtes et n nœuds. Si l'algorithme de Chazelle (2000) est utilisé, alors $tree(m,n) = O(m\alpha(m,n))$, où $\alpha(\cdot,\cdot)$ est la fonction d'Ackermann inverse. Par le Théorème 3.3, l'approche de Gabow and Tarjan (1984) peut être utilisée pour résoudre tous les problèmes $P(e_0, i, n-2-i)$, $(i, n-2-i) \in K$, $e_0 \in E$, en temps $O(mn(m\alpha(m,n)+n\log n))$. Par conséquent, le même temps est nécessaire pour résoudre le problème MIN-RISK-SPAN par la procédure `Enum`. Chen et al. (2009b) a proposé un algorithme en temps $O(m^2 \log m \log n(m + n\log n))$ pour le cas spécial du problème MIN-RISK-SPAN dans lequel les fonctions de risque sont $f_e(x_e) = \frac{d_e - x_e}{d_e - c_e}$.

3.3.2 Problème MIN-RISK-PATH

Dans le problème MIN-RISK-PATH, les éléments sont les arcs et les structures sont les chemins entre deux nœuds spécifiés a et b d'un graphe acyclique orienté $G = (V, E)$, $|V| = n$, $|E| = m$. Comme chaque chemin d'un graphe acyclique comprend au plus n noeuds et, par conséquent, au plus $n-1$ arcs, nous avons $s_{\max} \leq n-1$ pour le problème MIN-RISK-PATH, et nous pouvons fixer $K = \{(i,j) \mid i = 0, 1, \ldots, n-2,\ j = 0, 1, \ldots, n-2,\ i+j \leq n-2\}$. Par conséquent, $|K| = (n-1)\cdot n/2$.

Commentaire 3.2. *Si G contient des cycles, alors le problème* MIN-RISK-PATH *est fortement NP-difficile, car le problème fortement NP-complet* HAMILTONIAN PATH *s'y réduit en fixant $c_e = 0$, $d_e = 1$, $f_e(0) = 0$, $f_e(1) = -1$, $e \in E$, et $B = n-1$. Si la valeur optimale de ce problème est égale à $1-n$, alors le graphe contient un chemin hamiltonien, sinon, il ne contient pas de chemin hamiltonien.*

Pour $e_0 \in E$ et $(i,j) \in K$, chaque problème $P(e_0, i, j)$ peut être résolu par l'approche suivante. Construire un graphe acyclique $G^* = (V, E^*)$, qui diffère du graphe original $G = (V, E)$ en ce qu'il y a une copie supplémentaire, e', de chaque arc $e \in E \setminus \{e_0\}$. Chaque arc $u \in E^*$ est associé à un poids w_u et aux valeurs de trois ressources $r_u^{(1)}$, $r_u^{(2)}$ et $r_u^{(3)}$. Pour les arcs couplés e et e', nous définissons $w_e = c_e$, $r_e^{(1)} = 1$, $r_e^{(2)} = 0$, $r_e^{(3)} = 0$, et $w_{e'} = d_e$, $r_{e'}^{(1)} = 0$, $r_{e'}^{(2)} = 1$, $r_{e'}^{(3)} = 0$. Pour l'arc e_0, on définit $w_{e_0} = 0$, $r_{e_0}^{(1)} = 0$, $r_{e_0}^{(2)} = 0$, $r_{e_0}^{(3)} = 1$.

Considérons le problème de recherche d'un chemin de poids minimal S dans le graphe G^* entre les nœuds a et b de sorte que les valeurs totales des ressources 1, 2 et 3 le long de ce chemin soient égales à i, j et 1, respectivement : $\sum_{e\in S} r_e^{(1)} = i$, $\sum_{e\in S} r_e^{(2)} = j$ et $\sum_{e\in S} r_e^{(3)} = 1$. Nous désignons ce problème par RESOURCE-PATH(i, j, e_0).

Théorème 3.4 (Gurevsky et al. (2022a))**.** *Le problème $P(e_0, i, j)$ se réduit au problème* RESOURCE-PATH(i, j, e_0).

Tous les problèmes RESOURCE-PATH(i, j, e_0), $e_0 \in E$, $(i,j) \in K$, $i = 0, 1, \ldots, n-2$, $j = 0, 1, \ldots, n-2$, peuvent être résolus en temps $O(mn^2)$ par l'algorithme de programmation dynamique de Desrochers (1986), qui est une extension de l'algorithme de Joksch (1966) pour le problème avec une seule ressource. Le temps d'exécution de l'algorithme de Desrochers (1986) est de $O(mR_1R_2R_3)$, où R_h est une borne supérieure sur la valeur totale de la ressource h, $h = 1, 2, 3$. Par conséquent, le problème MIN-RISK-PATH peut être résolu en temps $O(mn^2)$ en employant la procédure `Enum`.

Commentaire 3.3. *Si $k_1 = 0$ (ou $k_2 = 0$) dans l'hypothèse 3.1, il n'est pas nécessaire d'énumérer les valeurs i (resp. valeurs j) dans les paires $(i,j) \in K$, car elles n'ont pas d'effet sur la valeur de la fonction objective. Par conséquent, si $k_1 = 0$ ou $k_2 = 0$, le problème* MIN-RISK-PATH *peut être résolu en temps $O(mn)$.*

Chen et al. (2009a) a présenté un algorithme en temps $O(mn^3)$ pour résoudre le cas du problème MIN-RISK-PATH, dans lequel les fonctions de risque sont $f_e(x_e) = \frac{d_e - x_e}{d_e - c_e}$, et donc, $k_2 = 0$.

3.4 Problèmes reverses

Dans cette section, on étudie une version reverse du problème MIN-RISK. Soit C un nombre réel donné. Le problème reverse peut être formulé comme suit.

Problème Rev-Min-Risk :

$$\min_{(S,x)} \sum_{e\in S} x_e, \quad \text{s.c.}$$
$$\sum_{e\in S} f_e(x_e) \le C,$$
$$S \in \mathcal{S}(E),$$
$$x_e \in \{c_e, c_e+1, \dots, d_e\},\ e \in S.$$

Le problème Rev-Min-Risk est une généralisation du problème sous contrainte de risque introduit par Álvarez-Miranda et al. (2011), que nous désignons par Rev-Min-Linear-Risk.

Problème Rev-Min-Linear-Risk :

$$\min_{(S,x)} \sum_{e\in S} x_e, \quad \text{s.c.}$$
$$\sum_{e\in S} \frac{d_e - x_e}{d_e - c_e} \le C,$$
$$S \in \mathcal{S}(E),$$
$$x_e \in \{c_e, c_e+1, \dots, d_e\},\ e \in S.$$

Exposons quelques propriétés d'une solution optimale du problème Rev-Min-Risk similaires à celles du problème Min-Risk. Le théorème suivant est une généralisation du Lemme 2.2 de Álvarez-Miranda et al. (2011) au cas des fonctions concaves non croissantes f_e.

Théorème 3.5 (Gurevsky et al. (2022a)). *Si le problème* Rev-Min-Risk *a une solution, alors il existe une solution optimale (S^*, x^*) de ce problème, dans lequel il y a au plus un élément $e^* \in S^*$ tel que toutes les variables sauf la variable $x^*_{e^*}$ sont égales à leurs bornes inférieures ou supérieures : $x^*_e \in \{c_e, d_e\}$, $e \in S^*\backslash\{e^*\}$, et $x^*_{e^*} = \min\left\{x \mid x \in \{c_{e^*}, c_{e^*}+1, \dots, d_{e^*}\},\ f_{e^*}(x) \le C - \sum_{e\in S^*\backslash\{e^*\}} f_e(x^*_e)\right\}$.*

Sur la base du Théorème 3.5, une procédure d'énumération similaire à `Enum` peut être employée pour résoudre le problème Rev-Min-Risk.

Procédure `Rev-Enum`.

1. Résoudre les $m \cdot |K|$ problèmes $P(e_0, i, j)$, $e_0 \in E$, $(i,j) \in K$. Soit Q l'ensemble des solutions optimales de ces problèmes.
2. Pour chaque solution $(S^{(e_0,i,j)}, x^{(e_0,i,j)}) \in Q$ telle que $\sum_{e\in S^{(e_0,i,j)}\backslash\{e_0\}} f_e(x_e^{(e_0,i,j)}) \le C - f_{e_0}(d_{e_0})$, construire une *solution étendue* $(S^{(e_0,i,j)}, \hat{x}^{(e_0,i,j)})$ telle que $\hat{x}_e^{(e_0,i,j)} = x_e^{(e_0,i,j)}$ pour $e \in S^{(e_0,i,j)}\backslash\{e_0\}$ et $\hat{x}_{e_0}^{(e_0,i,j)} = \min\left\{x \mid x \in \{c_{e_0}, c_{e_0}+1, \dots, d_{e_0}\},\ f_{e_0}(x) \le C - \sum_{e\in S^{(e_0,i,j)}\backslash\{e_0\}} f_e(x_e^{(e_0,i,j)})\right\}$. Soit Q_E l'ensemble des solutions étendues.
3. Choisir une solution étendue $(S^{(e_0,i,j)}, \hat{x}^{(e_0,i,j)}) \in Q_E$ avec la valeur minimale de $\sum_{e\in S^{(e_0,i,j)}} \hat{x}_e^{(e_0,i,j)}$. Elle est optimale pour le problème Rev-Min-Risk.

La procédure `Rev-Enum` peut être mise en œuvre pour s'exécuter en temps $O(R_1 + m|K|R_2)$, où R_1 est la complexité temporelle de son étape 1 et R_2 est la complexité temporelle du calcul de $\min\left\{x \mid x \in \{c_{e_0}, c_{e_0}+1, \dots, d_{e_0}\},\ f_{e_0}(x) \le C - \sum_{e\in S^{(e_0,i,j)}\backslash\{e_0\}} f_e(x_e^{(e_0,i,j)})\right\}$ pour tout $e_0 \in E$ à l'étape 2. Comme toutes les fonctions f_e sont non croissantes, le dernier minimum peut être calculé par une recherche dichotomique en temps $O(\log W)$, où $W = \max_{e\in E}\{d_e - c_e + 1\}$. Ainsi, la procédure `Rev-Enum` peut être implémentée pour s'exécuter en temps $O(R_1 + m|K|\log W)$.

Pour les cas de l'arbre couvrant minimal et du chemin le plus court du problème Rev-Min-Risk, les algorithmes des sections 3.3.1 et 3.3.2, respectivement, peuvent être utilisés à l'étape 1 de la procédure `Rev-Enum`.

3.5 Conclusion et perspectives

Dans ce chapitre, nous avons proposé quelques approches de résolution pour les problèmes Min-Risk, Rev-Min-Risk et leurs cas particuliers. Ces problèmes sont des généralisations des problèmes de minimisation du risque et de contrainte de risque introduits dans Chen et al. (2009a,b) et Álvarez-Miranda et al. (2011). Nos résultats s'inspirent des études précédentes, les généralisent et étendent les approches de résolution suggérées précédemment en introduisant des séries de problèmes $P(e_0, i, j)$ et en les résolvant à l'aide des algorithmes existants les plus efficaces.

La Table 3.1 contient les temps d'exécution asymptotiques des algorithmes proposés pour les problèmes avec la fonction de risque définie dans Chen et al. (2009a,b) et Álvarez-Miranda et al. (2011). Notons que les temps d'exécution asymptotiques de tous nos algorithmes sont inférieurs à ceux des méthodes précédentes dans Chen et al. (2009a,b) et Álvarez-Miranda et al. (2011).

Table 3.1 – Temps d'exécution d'algorithmes pour les problèmes avec fonctions de risque $f_e(x_e) = \frac{d_e - x_e}{d_e - c_e}$.

Problème	Gurevsky et al. (2022a)	Résultats précédents
Min-Risk-Span	$O(mn(m\alpha(m,n) + n\log n))$	Chen et al. (2009b) $O(m^2 \log m \log n(m + n\log n))$
Rev-Min-Risk-Span	$O(mn(m\alpha(m,n) + n\log n + \log W))$	Álvarez-Miranda et al. (2011) $O(m^3 \log m \log(md_{\max}) \log n(m + n\log n))$
Min-Risk-Path	$O(mn)$	Chen et al. (2009a) $O(mn^3)$
Rev-Min-Risk-Path	$O(mn \log W)$	Álvarez-Miranda et al. (2011) $O(m^2 n^3 \log(md_{\max}))$

Les résultats de ce chapitre sont issus d'une collaboration, débutée encore en 2015, avec Dmitry Kopelevich (UIIP, NASB, Minsk, Biélorussie), Sergey Kovalev (INSEEC Business School, Lyon) et Mikhail Y. Kovalyov (UIIP, NASB, Minsk, Biélorussie). Ils sont également publiés dans le journal Networks [15] (voir Gurevsky et al., 2022a). Dans le cadre de cette même collaboration, nous avons également apporté des améliorations pour le problème, introduit dans Hu (2010), qui vise à minimiser le risque maximum. Nous notons ce problème comme Min-Max-Risk. Ainsi, en se basant sur la compréhension plus avancée du problème traité, nous avons démontré dans Gurevsky et al. (2021), publié dans le journal 4OR [16], que la variante Min-Max-Risk avec fonctions de risque linéaires peut être réduite à un problème équivalent de programmation fractionnaire pour lequel la recherche paramétrique de Megiddo conduit à des algorithmes polynomiaux plus performants que ceux proposés par Hu (2010). Une autre approche, basée sur la recherche dichotomique, a été également proposée. La Table 3.2 présente la comparaison des algorithmes mentionnés.

Table 3.2 – Temps d'exécution d'algorithmes pour les problèmes avec fonctions de risque $f_e(x_e) = \frac{d_e - x_e}{d_e - c_e}$.

Problème	Hu (2010)	Gurevsky et al. (2021)	
		Recherche paramétrique	Recherche dichotomique
Min-Max-Risk-Path	$O(n^5)$	$O(m^2 + n^2(\log n)^2)$	$O((m + n\log n)\log W)$
Min-Max-Risk-Span	$O(nm^3 \log m)$	$O(m\alpha(m,n)\log n)$	$O(m\alpha(m,n)\log W)$

Enfin, nous avons également démontré dans Gurevsky et al. (2023), publié dans le journal 4OR, que le problème de type «sac-à-dos» avec données contrôlables peut être résolu en temps linéaire avec fonctions de profit convexes non-décroissantes, mais qui reste cependant NP-difficile avec celles non-décroissantes quelconques. Pour cette dernière variante, nous avons toutefois développé un schéma d'approximation polynomial FPTAS en temps $O(n^3 \log n + \frac{n^3}{\varepsilon^2})$ où n est le nombre d'objets. Il est important de noter que en plus des 3 contributions dans des journaux, les résultats de cette troisième direction de recherche ont été également présentés lors de 4 conférences internationales/nationales et 2 séminaires invités (voir l'annexe A pour plus de détails).

15. Networks est un journal international dédié à la résolution des problèmes de conception et d'optimisation des réseaux, édité par le groupe Wiley.
16. 4OR – A Quarterly Journal of Operations Research, édité par le groupe Springer.

En ce qui concerne les perspectives de recherche, avec Nadia Brauner (Université Grenoble Alpes), Mikhail Y. Kovalyov et Erwin Pesch (Université de Siegen, Allemagne), nous portons actuellement notre attention sur une direction scientifique qui traite des données sous forme d'intervalles, mais qui se situe davantage dans le domaine de l'optimisation robuste, à savoir la robustesse de type «min-max-min» ou la robustesse à deux étapes. Ainsi, une première contribution a été récemment publiée dans la revue DAM (voir Brauner et al., 2024) où, pour une variante du problème de sélection, nous avons pu proposer un algorithme polynomial plus efficace que celui de Chassein and Goerigk (2021).

Conclusion générale et perspectives

Ce mémoire d'habilitation à diriger des recherches peut être vu comme une sorte de résumé de mes activités principales scientifiques et d'encadrement depuis mon recrutement à l'Université de Nantes en 2012. Il est structuré autour de 3 chapitres, dont chacun présente une orientation scientifique à part entière.

Ainsi, le chapitre 1 concerne l'une des mes plus anciennes activités de recherche et porte sur l'équilibrage robuste de lignes de production avec tâches de durée incertaine. Il est dédié à une contribution réalisée dans le cadre de la thèse d'Aleksandr Pirogov que j'ai pu co-encadrer. De nombreuses autres contributions et collaborations relevant de cette première orientation scientifique sont également évoquées. Le chapitre 2, quant à lui, aborde l'une de mes orientations académiques les plus récentes, à savoir l'équilibrage de lignes de production reconfigurables avec équipement modulaire. Il s'agit de l'un des résultats obtenus dans le cadre de la thèse d'Abdelkrim R. Yelles-Chaouche, financée par le programme IRT PERFORM 2018. Comme dans le chapitre 1, le chapitre 2 se termine également par la présentation d'autres résultats de cette thèse. Enfin, le chapitre 3 est une orientation scientifique, initiée en 2015, que j'ai pu poursuivre en parallèle et en dehors de mes encadrements doctoraux et post-doctoraux. Il s'agit ici d'une contribution issue de l'étude d'une nouvelle classe de problèmes de gestion du risque avec données contrôlables. À la fin de ce dernier chapitre, je mentionne tous les résultats obtenus dans cette direction, et je présente également une nouvelle orientation qui a émergé.

Depuis mon recrutement à l'Université de Nantes en 2012, le bilan de la production scientifique [17] issue de ces trois orientations de recherche comprend 13 articles dans des journaux internationaux, 12 communications dans des conférences internationales, 9 communications dans des conférences nationales et 3 communications dans des séminaires invités. Jusqu'à présent, mes activités en matière de recherche ont été principalement articulées autour de deux grands domaines tels que «systèmes de production» (chapitres 1 et 2) et «incertitude et risque» (chapitres 1 et 3). En termes d'approches de résolution, je me suis principalement appuyé sur des techniques de pré-traitement permettant la réduction de l'espace de recherche et la génération de contraintes spécifiques (chapitres 1 et 2) ainsi que sur l'étude de la complexité algorithmique (chapitre 3).

Cette conclusion générale ne serait pas complète sans mentionner les trois co-encadrements de thèse que j'effectue actuellement et qui représentent mes nouvelles perspectives scientifiques, du moins à court terme. Il s'agit tout d'abord de Siwar Arbi qui a commencé sa thèse de doctorat en novembre 2022 dans le cadre du projet ANR «ReconfiDurable». Cette thèse porte sur la conception évolutive de lignes de production reconfigurables sous contraintes de ressources et d'incertitude. L'objectif de cette thèse est d'utiliser des ressources supplémentaires (mais limitées) pour reconfigurer de manière optimale la ligne de production en cas d'incertitude concernant la durée des tâches ou la demande. Siwar Arbi est également co-encadrée par Audrey Cerqueus (IMT Atlantique, Nantes), Alexandre Dolgui et Ali Siadat (ENSAM, Metz), et se concentre sur les méthodes méta-heuristiques, l'optimisation multi-objectif et la programmation stochastique.

La deuxième thèse est celle d'Emmanuelle Amann, débutée en février 2023 dans le cadre du projet CIFRE avec la société «La Poste». L'un des objectifs de cette thèse est de revoir l'approche actuelle de la planification du tri du courrier à La Poste, afin de l'améliorer et de la rendre plus dynamique. Les problèmes étudiés sont essentiellement des variantes du problème de «bin-packing» avec soit des

17. Pour plus de détails, j'invite à consulter mon CV, présenté dans l'annexe A où, entre autres, mes responsabilités collectives et d'intérêt général sont également indiquées.

contraintes de précédence particulières sous la forme de plusieurs chaînes indépendantes soit des objets de type vectoriel. Emmanuelle Amann est également co-encadrée par Arnaud Laurent (Université de Nantes) et Nasser Mebarki (Université de Nantes).

Enfin, Victor Spitzer, qui a entamé sa thèse de doctorat en mars 2023 dans le cadre d'un autre projet CIFRE avec l'entreprise «Lhyfe». Le sujet principal de sa thèse est de minimiser le coût de production de l'hydrogène obtenu par le procédé d'électrolyse de l'eau, en utilisant deux sources d'énergie intermittentes : l'électricité verte générée par le parc éolien en mer, dont le prix est négligeable, et le réseau électrique national, dont le prix est plus élevé. Trois types d'incertitude sont étudiés : l'incertitude liée aux erreurs de prévision de la puissance éolienne, l'incertitude sur le prix de l'électricité un jour à l'avance du réseau national, et l'incertitude sur la demande des clients. Les problèmes étudiés sont de type «unit commitment» et «lot-sizing». Les méthodes utilisées pour aborder ces problèmes sous incertitude sont la programmation stochastique et l'optimisation robuste couplées à des techniques d'apprentissage et d'analyse de données pour la génération plus fidèle de scénarios de comportement des paramètres incertains. Victor Spitzer est co-encadré également par Céline Gicquel (Université Paris-Saclay, Gif-sur-Yvette).

Dans les années à venir, j'ai l'intention de me recentrer sur les trois sujets suivants : me tourner davantage vers les applications liées à l'utilisation rationnelle des ressources énergétiques, me pencher plus spécifiquement sur l'étude de la pertinence de la méthode de décomposition de Benders pour les problèmes d'optimisation sous incertitude, et poursuivre l'analyse de la complexité algorithmique dans le cadre de l'optimisation robuste à deux étapes :

- Aujourd'hui, les questions de transition, de transformation et d'efficacité énergétiques prennent une importance croissante et concernent pratiquement tous les États et toutes les industries du monde, afin d'offrir aux consommateurs, entre autres, un approvisionnement énergétique durable et respectueux de l'environnement à un coût raisonnable. De nombreuses solutions et approches sont proposées pour répondre à ces questions. Cependant, leur mise en place réussie nécessitent souvent l'utilisation d'outils d'optimisation et de recherche opérationnelle très élaborés, impliquant des techniques telles que la programmation stochastique à plusieurs étapes, l'optimisation à deux niveaux, l'analyse de données multi-dimensionnelles et l'apprentissage automatique. C'est dans ce contexte que j'ai commencé à co-encadrer la thèse CIFRE de Victor Spitzer sur le sujet très motivant et passionnant de l'optimisation de la production d'hydrogène vert, en coopération avec l'entreprise «Lhyfe», avec laquelle j'ai l'intention de poursuivre la collaboration au-delà de la thèse de Victor. Une autre application sur laquelle je compte également m'investir concerne les systèmes de production, où la prise en compte opérationnelle et efficace des prix de l'électricité proposés aux industriels, qui peuvent varier d'un jour à l'autre (voire d'une heure à l'autre) est une tâche cruciale dans la planification des reconfigurations du système, et a un impact direct sur le prix du produit final. Le travail sur ce dernier sujet n'a pas encore démarré, mais j'ai déjà pris contact avec des collègues de l'École des Mines de Saint-Étienne, qui travaillent dans ce domaine depuis un certain nombre d'années.
- Malgré son ancienneté, la décomposition de Benders reste souvent une technique de résolution indispensable pour de nombreuses applications en recherche opérationnelle. Toutefois, son efficacité dépend fortement de la manière dont le problème d'optimisation en question est décomposé, ainsi que de son degré de difficulté. Son utilisation est assez naturelle au sein de la méthode, dite $\mathcal{L}$-shaped, employée pour la résolution de problèmes en programmation stochastique, où la décomposition est effectuée pour chaque scénario modélisant le comportement des paramètres incertains. Nous nous sommes déjà rendus compte de l'efficacité de cette technique pour certains problèmes stochastiques de dimensionnement de lots dans le cadre de la thèse de Victor Spitzer. Dans un avenir proche, j'aimerais poursuivre cette ligne de recherche et, en collaboration avec André Rossi et Rui S. Shibasaki, aborder également les problèmes d'équilibrage robuste de lignes de production avec des tâches de durée incertaine, où la décomposition de Benders aboutit au problème maître cherchant à déterminer une configuration de ligne réalisable sous certaines contraintes relaxées, tandis que le sous-problème vise à calculer la mesure de robustesse à maximiser pour la configuration fournie par le problème maître. En outre, il serait également intéressant de voir le comportement de la décomposition de Benders appliquée à la résolution de problèmes de planification de reconfiguration de lignes de production soumis à l'incertitude de la demande dans le cadre de la thèse de

Siwar Arbi. L'une des particularités de ces problèmes réside dans le fait que le problème maître n'a pas de contraintes de capacité à vérifier et dans la génération de coupes de Benders adaptatives qui sont ajoutées au problème maître dans un format modifié par rapport à leur version originale à la suite de la résolution des sous-problèmes.

- En ce qui concerne la robustesse de type «min-max-min», il s'agit d'une orientation scientifique très récente en recherche opérationnelle, avec de nombreuses applications intéressantes. Elle consiste à construire une solution (ou bien déterminer une politique) pour le problème étudié qui se comporterait de la manière la plus résiliente possible face à la réaction adversariale de l'état de la nature qui viserait à mettre notre solution dans son pire cas, mais avec l'idée de pouvoir la corriger légèrement durant la seconde étape une fois que l'état de la nature s'est réalisé. La plupart de ces problèmes d'optimisation sont, bien entendu, NP-difficiles, et les seuls qui présentent parfois des cas polynomiaux non évidents sont les variantes du problèmes de sélection (ou «items selection problems» en anglais). C'est en collaboration avec Nadia Brauner, Mikhail Y. Kovalyov et Erwin Pesch que je compte approfondir l'étude de cas polynomiaux élégants pour ces problèmes. Une autre piste de travail future consiste à étudier des problèmes robustes à deux étapes avec le comportement modifié des deux agents (décideur et adversaire), où leurs politiques comportementales sont, par exemple, annoncées dès le départ pour les deux étapes en même temps pour les deux agents afin d'approcher des situations déjà décrites dans le domaine de la théorie des jeux.

Pour terminer, je tiens à remercier tous mes co-auteurs et tous mes collègues de l'équipe MODELIS ainsi que du laboratoire LS2N, dont j'apprécie grandement l'environnement scientifique motivant, inspirant et dynamique, sans lequel ce mémoire n'aurait probablement jamais vu le jour.

Annexe A

Curriculum vitæ

A.1 Éducation

École des Mines de Saint-Étienne, France
Centre de Génie Industriel et Informatique — *2008 – 2011*
Doctorat en Génie Industriel

Université d'État de Biélorussie, Minsk, Biélorussie
UFR Mécanique et Mathématiques — *2003 – 2008*
Master en Mathématiques

Lycée UEB, Minsk, Biélorussie
Specialité «Mathématiques» — *2003*
Baccalauréat Scientifique

A.2 Parcours académique

Université de Nantes, France — **Maître de Conférences**
Équipe MODELIS[18]*, LS2N*[19] — *Septembre 2012 – à présent*
Département d'Informatique, UFR Sciences et Techniques

Université d'Avignon, France — **ATER (100%)**
Équipe «Recherche Opérationnelle et Optimisation», LIA[20] — *Septembre 2011 – Août 2012*
Centre d'Enseignement et de Recherche en Informatique
`Tuteur : Boris Detienne`

École des Mines de Saint-Étienne, France — **Doctorant - Moniteur**
Équipe «Recherche Opérationnelle pour le Génie Industriel», LSTI[21] — *Octobre 2008 – Août 2011*
Centre de Génie Industriel et Informatique
`Encadrants : Olga Battaïa et Alexandre Dolgui`

Université d'État de Biélorussie, Minsk, Biélorussie — **Chercheur Junior**
Équipe «Recherche Opérationnelle et Algorithmes» — *Septembre 2005 – Août 2008*
`Encadrant : Vladimir A. Emelichev`

18. MODELIS – Modélisation, Optimisation et DEcision pour la Logistique, l'Industrie et les Services.
19. LS2N (UMR CNRS 6004) – Laboratoire des Sciences du Numérique de Nantes.
20. LIA (EA 4128) – Laboratoire Informatique d'Avignon.
21. LSTI – Laboratoire en Sciences et Technologies de l'Information.

A.3 Thèmes de recherche et production scientifique

Mon principal domaine d'activité scientifique est la recherche opérationnelle et ses applications. Je m'intéresse actuellement aux problèmes d'optimisation liés à la production/fabrication, à la planification et aux systèmes de tri ainsi qu'à l'optimisation robuste et stochastique.

La démarche de mes études est souvent composée en trois étapes :

1. Analyse de la complexité algorithmique du problème traité.
2. Étude de la structure du problème afin d'identifier des propriétés intéressantes et des bornes supérieures/inférieures de bonne qualité sur l'optimum.
3. Développement de méthodes de résolution exactes ou approchées en se basant sur des règles de dominance efficaces et des techniques de pré-traitement appropriées.

À présent, mes activités de recherche se concentrent autour des cinq thèmes suivants :

1. Équilibrage robuste de lignes de production avec tâches de durée incertaine.
2. Modularité et évolutivité dans le cadre de lignes de production reconfigurables.
3. Robustesse et contrôlabilité en optimisation combinatoire.
4. Planification de la production d'hydrogène vert en présence d'erreurs dans les prévisions de la puissance éolienne.
5. Lissage de la charge et optimisation des plans de tri du courrier dans la distribution postale.

Ci-dessous, je présente sous forme de tableau tous les collaborateurs avec lesquels j'ai pu travailler sur au moins un de ces cinq thèmes depuis mon arrivée à l'Université de Nantes en 2012.

Collaborateur	**Affiliation actuelle**	**Thèmes**				
		1	**2**	**3**	**4**	**5**
Emmanuelle Amann	La Poste, Nantes					✓
Siwar Arbi	IMT Atlantique, Nantes		✓			
Olga Battaïa	Kedge Business School, Talence	✓				
Nadjib Brahimi	Rennes School of Business		✓			
Nadia Brauner	Univeristé Grenoble Alpes			✓		
Audrey Cerqueus	IMT Atlantique, Nantes		✓			
Alexandre Dolgui	IMT Atlantique, Nantes	✓	✓			
Paul Feliot	Lhyfe, Nantes				✓	
Céline Gicquel	Université Paris-Saclay, Gif-sur-Yvette				✓	
Dmitry Kopelevich	UIIP, NASB, Minsk, Biélorussie			✓		
Sergey Kovalev	INSEEC Business School, Lyon			✓		
Mikhail Y. Kovalyov	UIIP, NASB, Minsk, Biélorussie			✓		
Arnaud Laurent	Université de Nantes					✓
Nasser Mebarki	Université de Nantes					✓
Aleksandr Pirogov	Tesco, Londres, Angleterre	✓				
Andry Rasamimanana	Université de Nantes	✓				
André Rossi	Université Paris-Dauphine	✓			✓	
François Sanson	Lhyfe, Nantes				✓	
Xavier Schepler	Generix, Paris	✓				
Rui S. Shibasaki	Université d'Amiens	✓				
Ali Siadat	ENSAM, Metz		✓			
Victor Spitzer	Lhyfe, Nantes				✓	
David Tremblet	HEC Montréal, Canada		✓			
Abdelkrim R. Yelles-Chaouche	Festo, Marseille		✓			

Le tableau suivant donne une vue synthétique de toutes mes productions scientifiques.

Type	**Nombre**
Articles dans des journaux internationaux	22
Articles dans des journaux nationaux	5
Publications dans des conférences internationales	31
Publications dans des conférences nationales	24
Communications sans publication	7
Mémoires (Master + Doctorat)	2

Articles dans des journaux internationaux (22) .....

1. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** Optimizing modular equipment in the design of multi-product reconfigurable production lines. *Computers & Industrial Engineering*, 192 : 110226, 2024. (Elsevier)

2. **Brauner N., Gurevsky E., Kovalyov M.Y.** A single representative min-max-min robust selection problem with alternatives and budgeted uncertainty. *Discrete Applied Mathematics*, 349 : 106–112, 2024. (Elsevier)

3. **Shibasaki R.S., Rossi A., Gurevsky E.** A new upper bound based on Dantzig-Wolfe decomposition to maximize the stability radius of a simple assembly line under uncertainty. *European Journal of Operational Research*, 313(3) : 1015–1030, 2024. (Elsevier)

4. **Gurevsky E., Kopelevich D., Kovalev S., Kovalyov M.Y.** Integer knapsack problems with profit functions of the same value range. *4OR : A Quarterly Journal of Operations Research*, 21(3) : 405–419, 2023. (Springer)

5. **Tremblet D., Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** Optimizing task reassignments for reconfigurable multi-model assembly lines with unknown order of product arrival. *Journal of Manufacturing Systems*, 67 : 190–200, 2023. (Elsevier)

6. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** Minimizing task reassignments under balancing multi-product reconfigurable manufacturing lines. *Computers & Industrial Engineering*, 173 : 108660, 2022. (Elsevier)

7. **Gurevsky E., Rasamimanana A., Pirogov A., Dolgui A., Rossi A.** Stability factor for robust balancing of simple assembly lines under uncertainty. *Discrete Applied Mathematics*, 318 : 113–132, 2022. (Elsevier)

8. **Gurevsky E., Kopelevich D., Kovalev S., Kovalyov M.Y.** Min-sum controllable risk problems with concave risk functions of the same value range. *Networks*, 79(1) : 105–116, 2022. (Wiley)

9. **Schepler X., Rossi A., Gurevsky E., Dolgui A.** Solving robust bin-packing problems with a branch-and-price approach. *European Journal of Operational Research*, 297(3) : 831–843, 2022. (Elsevier)

10. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** Reconfigurable manufacturing systems from an optimization perspective : a focused review of literature. *International Journal of Production Research*, 59(21) : 6400–6418, 2021. (Taylor & Francis)

11. **Gurevsky E., Kovalev S., Kovalyov M.Y.** Min-max controllable risk problems. *4OR : A Quarterly Journal of Operations Research*, 19(1) : 93–101, 2021. (Springer)

12. **Pirogov A., Gurevsky E., Rossi A., Dolgui A.** Robust balancing of transfer lines with blocks of uncertain parallel tasks under fixed cycle time and space restrictions. *European Journal of Operational Research*, 290(3) : 946–955, 2021. (Elsevier)

13. **Rossi A., Gurevsky E., Battaïa O., Dolgui A.** Maximizing the robustness for simple assembly lines with fixed cycle time and limited number of workstations. *Discrete Applied Mathematics*, 208 : 123–126, 2016. (Elsevier)

14. **Gurevsky E., Hazır Ö., Battaïa O., Dolgui A.** Robust balancing of simple assembly lines with interval task processing times. *Journal of the Operational Research Society*, 64(11) : 1607–1613, 2013. (Palgrave Macmillan)

15. **Battaïa O., Gurevsky E., Makssoud F., Dolgui A.** Equipment location in machining transfer lines with multi-spindle heads. *Journal of Mathematical Modeling and Algorithms in Operations Research*, 12(2) : 117–133, 2013. (Springer)

16. **Gurevsky E., Battaïa O., Dolgui A.** Stability measure for a generalized assembly line balancing problem. *Discrete Applied Mathematics*, 161(3) : 377–394, 2013. (Elsevier)

17. **Gurevsky E., Battaïa O., Dolgui A.** Balancing of simple assembly lines under variations of task processing times. *Annals of Operations Research*, 201(1) : 265–286, 2012. (Springer)

18. **Guschinskaya O., Gurevsky E., Dolgui A., Eremeev A.** Metaheuristic approaches for the design of machining lines. *The International Journal of Advanced Manufacturing Technology*, 55(1-4) : 11–22, 2011. (Springer)

19. **Emelichev V.A., Gurevsky E.E., Kuzmin K.G.** On stability of some lexicographic integer optimization problem. *Control and Cybernetics*, 39(3) : 811–826, 2010.

20. **Gurevskii E.E., Emelichev V.A.** On five types of stability of the lexicographic variant of the combinatorial bottleneck problem. *Discrete Mathematics and Applications*, 19(4) : 337–348, 2009. (Walter de Gruyter)

21. **Emelichev V.A., Gurevskii E.E.** On the regularization of vector integer quadratic programming problems. *Cybernetics and Systems Analysis*, 45(2) : 274–280, 2009. (Springer)

22. **Emelichev V.A., Gurevskii E.E.** Stability of a combinatorial vector partition problem. *Cybernetics and Systems Analysis*, 43(3) : 462–465, 2007. (Springer)

Articles dans des journaux nationaux (5) .....

1. **Emelichev V.A., Gurevsky E.E., Platonov A.A.** On stability and quasi-stability radii for a vector combinatorial problem with a parametric optimality principle. *Buletinul Academiei de Ştiinţe a Republicii Moldova. Matematica*, (2) : 55–61, 2009.

2. **Gurevsky E.E., Emelichev V.A.** Postoptimal analysis of a vector partition problem with ordered criteria. *Proceedings of the Institute of Mathematics of the NAS of Belarus*, 16(2) : 14–22, 2008. (in Russian)

3. **Emelichev V.A., Gurevsky E.E.** On stability kernel of a multicriteria combinatorial minimax problem. *Discrete Analysis and Operations Research*, 15(5) : 6–19, 2008. (in Russian)

4. **Emelichev V.A., Gurevsky E.E.** Linear convolution of criteria in the vector p-center problem. *Buletinul Academiei de Ştiinţe a Republicii Moldova. Matematica*, (1) : 73–82, 2007.

5. **Emelichev V.A., Gurevsky E.E., Platonov A.A.** Measure of stability for a finite cooperative game with a generalized concept of equilibrium. *Buletinul Academiei de Ştiinţe a Republicii Moldova. Matematica*, (3) : 17–26, 2006.

Publications dans des conférences internationales (31) .....

1. **Amann E., Gurevsky E., Laurent A., Mebarki N.** Minimizing the number of mail sorting sessions as a variant of vector bin-packing : A case study at La Poste. *Proceedings of the 18th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2024). Vienna (Austria), August 28–30, 2024.* 4 pages.

2. **Spitzer V., Gicquel C., Gurevsky E., Sanson F.** Day-ahead lot-sizing under uncertainty : An application to green hydrogen production. *33rd European Conference on Operational Research (EURO 2024). Copenhagen (Denmark), June 30 – July 03, 2024.* Conference Program, p. 367.

3. **Arbi S., Cerqueus A., Gurevsky E., Dolgui A.** Bi-objective based scalable balancing of a reconfigurable assembly line. *33rd European Conference on Operational Research (EURO 2024). Copenhagen (Denmark), June 30 – July 03, 2024.* Conference Program, p. 318.

4. **Arbi S., Cerqueus A., Gurevsky E., Dolgui A., Siadat A.** A multi-objective approach to design a scalable reconfigurable assembly line in an uncertain environment. *27th International Conference on Multiple Criteria Decision Making (MCDM2024). Hammamet (Tunisia), June 2–7, 2024.* Abstracts, 1 page.

5. **Spitzer V., Gicquel C., Gurevsky E., Sanson F.** Day-ahead lot-sizing under uncertainty : An application to green hydrogen production. *Proceedings of the 8th International Symposium on Combinatorial Optimization (ISCO 2024). La Laguna (Tenerife, Canary Islands, Spain), May 22–24, 2024.* In : Basu A., Mahjoub A.R. and Salazar González J.J. (Eds.) Combinatorial Optimization. *Lecture Notes in Computer Science*, 14594 : 402–416, 2024. (Springer)

6. **Arbi S., Cerqueus A., Gurevsky E., Dolgui A., Siadat A.** Simulated annealing for a bi-objective-based scalable balancing of a reconfigurable assembly line under uncertainty and limited resources. *Proceedings of the 50th International Conference on Computers & Industrial Engineering (CIE50). Sharjah (UAE), October 30 – November 02, 2023.* Volume 2, p. 1074–1083.

7. **Amann E., Gurevsky E., Laurent A., Mebarki N.** Mail sorting optimization : a case study of the French postal company La Poste. *Proceedings of the 22nd IFAC World Congress. Yokohama (Japan), July 09–14, 2023.* 6 pages.

8. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** A MILP-based heuristic approach for the design of multi-product modular reconfigurable lines. *Proceedings of the 10th IFAC Conference on Manufacturing Modeling, Management and Control (MIM 2022). Nantes (France), June 22–24, 2022.* 5 pages.

9. **Tremblet D., Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** Balancing multi-model assembly lines in a reconfigurable environment. *31st European Conference on Operational Research (EURO 2021). Athens (Greece), July 11–14, 2021.* Conference Handbook, p. 244.

10. **Pirogov A., Rossi A., Gurevsky E., Dolgui A.** Search space reduction in MILP approaches for the robust balancing of transfer lines. *17th International Workshop on Project Management and Scheduling (PMS 2020/2021). Toulouse (France), April 21–23, 2021.* Book of Extended Abstracts, p. 257–260.

11. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** Modular equipment optimization in the design of multi-product reconfigurable manufacturing systems. *17th International Workshop on Project Management and Scheduling (PMS 2020/2021). Toulouse (France), April 21–23, 2021.* Book of Extended Abstracts, p. 362–365.

12. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** Minimizing task reassignments in the design of reconfigurable manufacturing lines with space restrictions. *Proceedings of the 21st IFAC World Congress. Berlin (Germany), July 12–17, 2020.* 6 pages.

13. **Brahimi N., Dolgui A., Gurevsky E., Yelles-Chaouche A.R.** A literature review of optimization problems for reconfigurable manufacturing systems. *Proceedings of the 9th IFAC Conference on Manufacturing Modeling, Management and Control (MIM 2019). Berlin (Germany), August 28–30, 2019.* 6 pages.

14. **Pirogov A., Gurevsky E., Rossi A., Dolgui A.** Optimality-based cuts generation for robust balancing of paced production lines with blocks of uncertain parallel tasks and space restrictions. *29th European Conference on Operational Research (EURO 2018). Valencia (Spain), July 08–11, 2018.* Conference Handbook, p. 136.

15. **Pirogov A., Gurevsky E., Rossi A., Dolgui A.** A MILP approach for robust transfer line design. *7th International Conference on Industrial Engineering and Systems Management (IESM 2017). Saarbrücken (Germany), October 11–13, 2017.* Conference Proceedings, p. 20–24.

16. **Schepler X., Rossi A., Gurevsky E., Dolgui A.** A branch-and-price algorithm for robust bin packing with variable size items. *Joint EURO/ORSC/ECCO Conference 2017 on Combinatorial Optimization (ECCO XXX – 2017). Koper (Slovenia), May 3–6, 2017.* Book of Abstracts, p. 19–20.

17. **Gurevsky E., Kovalev S., Kovalyov M.Y.** A min-max risk version of some combinatorial optimization problems with controllable data. *Joint EURO/ORSC/ECCO Conference 2017 on Combinatorial Optimization (ECCO XXX – 2017). Koper (Slovenia), May 3–6, 2017.* Book of Abstracts, p. 22.

18. **Gurevsky E., Kovalev S., Kovalyov M.Y.** Combinatorial problems of risk management for routing and network design. *28th European Conference on Operational Research (EURO 2016). Poznań (Poland), July 3–6, 2016.* Conference Handbook, p. 70.

19. **Gurevsky E., Kovalev S., Kovalyov M.Y.** Polynomially solvable knapsack type problems with box constraints on combinatorial structures. *28th Conference of the European Chapter on Combinatorial Optimization (ECCO XXVIII – 2015). Catania (Italy), May 28–30, 2015.* Abstracts, p. 59.

20. **Gurevsky E., Merel A., Gandibleux X.** A column generation based algorithm for a bi-objective capacity evaluation of a railway infrastructure. *22nd International Conference on Multiple Criteria Decision Making (MCDM2013). Málaga (Spain), June 17–21, 2013.* Book of Abstracts, p. 78.

21. **Gurevsky E., Guschinskaya O., Dolgui A.** Post-optimal analysis for a design problem of machining lines. *Proceedings of the 5th IFAC International Conference on Management and Control of Production and Logistics (MCPL 2010). Coimbra (Portugal), September 08–10, 2010.* 5 pages.

22. **Gurevsky E., Guschinskaya O., Dolgui A.** Analyse de sensibilité du problème de l'équilibrage des lignes d'usinage : perturbation des temps opératoires. *Actes de la 8ème ENIM IFAC Conférence Internationale de Modélisation et Simulation (MOSIM'10). Hammamet (Tunisie), 10–12 Mai, 2010.* 8 pages.

23. **Gurevsky E., Guschinskaya O., Dolgui A.** Assembly line balancing problems under uncertainties. *Proceedings of the 25th Mini-EURO Conference on Uncertainty and Robustness in Planning and Decision Making (URPDM 2010). Coimbra (Portugal), April 15–17, 2010.* 8 pages.

24. **Gurevsky E., Guschinskaya O., Eremeev A., Dolgui A.** Balancing machining transfer lines using genetic algorithms. *Proceedings of the 39th International Conference on Computers & Industrial Engineering (CIE39). Troyes (France), July 6–8, 2009.* P. 1850–1855.

25. **Guschinskaya O., Gurevsky E., Dolgui A., Eremeev A.** Métaheuristiques pour la conception en avant-projet des lignes d'usinage. *Actes du 8ème Congrès International de Génie Industriel (CIGI'09). Bagnères-de-Bigorre (France), 10–12 Juin, 2009.* 8 pages.

26. **Gurevsky E., Guschinskaya O., Eremeev A., Dolgui A.** Metaheuristic methods for balancing machining transfer lines. *Proceedings of the 42nd CIRP Conference on Manufacturing Systems. Grenoble (France), June 3–5, 2009.* 7 pages.

27. **Emelichev V., Gurevsky E.** Postoptimal analysis of lexicographic combinatorial minimax problems. *The Second International Conference «Problems of Cybernetics and Informatics» (PCI'2008). Baku (Azerbaijan), September 10–12, 2008.* Volume 3, p. 73–75.

28. **Gurevsky E.E., Emelichev V.A.** Regularizing and ε-regularizing of vector integer quadratic programming problems. *Proceedings of the XIV Baikal International School-Seminar «Optimization Methods and Their Applications». Irkutsk, Baikal (Russia), July 2–8, 2008.* Volume 1. Mathematical Programming, p. 288–295. (in Russian)

29. **Gurevsky E.E., Emelichev V.A., Platonov A.A.** Finite cooperative games with a parametric concept of equilibrium under uncertainties. *The Third International Scientific Conference «Scheduling Theory and Decomposition Methods. Tanaev's Readings». Minsk (Belarus), March 28, 2007.* Transactions, p. 40–45. (in Russian)

30. **Gurevsky E.E., Emelichev V.A.** A multicriteria combinatorial partition problem under uncertainties. *Materials of the IX International Conference «Intelligent Systems and Computer Science». Moskow (Russia), October 23–27, 2006.* Volume 1. Part 1, p. 117–123. (in Russian)

31. **Emelichev V.A., Gurevsky E.E.** A vector partition problem with uncertain numeric parameters. *The XIV Conference on Applied and Industrial Mathematics, dedicated to the 60th anniversary of the foundation of the Faculty of Mathematics and Computer Science of Moldova State University. Kishinev (Moldova), August 17–19, 2006.* Communications, p. 163–165.

Publications dans des conférences nationales (24) .....

1. **Spitzer V., Gicquel C., Gurevsky E., Sanson F.** Structural learning of electricity prices for production. *26th Congress of the French Society of Operations Research and Decision Aid (ROADEF 2025). Champs-sur-Marne (France), February 26–28, 2025.* Abstracts, 6 pages.
   `Finaliste du Prix ROADEF 2025 pour le Meilleur Article Étudiant.`

2. **Spitzer V., Gicquel C., Gurevsky E., Sanson F.** Dynamic two-stage programming for the stochastic lot-sizing problem with inventory bounds and lost sales. *Conference «PGMO Days 2024». Palaiseau (France), November 19–20, 2024.* Book of Abstracts, p. 119.

3. **Arbi S., Cerqueus A., Gurevsky E., Dolgui A., Siadat A.** A hyper-volume based metric for scalable design of reconfigurable manufacturing systems with multiple objectives. *2nd Congress of the Society of Automatics, Industrial Engineering and Production (SAGIP 2024). Villeurbanne (France), May 29–31, 2024.* Abstracts, 2 pages.

4. **Amann E., Laurent A., Gurevsky E., Mebarki N.** Constitution optimale des sessions de tri du courrier à La Poste. *Résumés du 2ème Congrès Annuel de la Société d'Automatique, de Génie Industriel et de Productique (SAGIP 2024). Villeurbanne (France), 29–31 Mai, 2024.* 2 pages.

5. **Arbi S., Cerqueus A., Gurevsky E., Dolgui A., Siadat A.** A literature review of optimization problems focused on scalability aspects for manufacturing systems. *25th Congress of the French Society of Operations Research and Decision Aid (ROADEF 2024). Amiens (France), March 4–7, 2024.* Abstracts, 2 pages.

6. **Shibasaki R.S. , Rossi A., Gurevsky E.** Dantzig-Wolfe decomposition for robust balancing of a simple assembly line under uncertainty. *25th Congress of the French Society of Operations Research and Decision Aid (ROADEF 2024). Amiens (France), March 4–7, 2024.* Abstracts, 2 pages.

7. **Spitzer V., Gicquel C., Gurevsky E., Sanson F.** Dimensionnement de lots de production un jour à l'avance sous incertitudes éoliennes : application à la production d'hydrogène vert. *Résumés du 25ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2024). Amiens (France), 4–7 Mars, 2024.* 6 pages.
   `Lauréat du Prix ROADEF 2024 pour le Meilleur Article Étudiant.`

8. **Amann E., Gurevsky E., Mebarki N.** Optimisation du tri du courrier : un cas d'étude à La Poste. *Résumés du 24ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2023). Rennes (France), 20–23 Février, 2023.* 2 pages.

9. **Spitzer V., Sanson F., Feliot P., Gurevsky E., Rossi A.** Optimisation de la production d'hydrogène vert renouvelable sous incertitudes. *Résumés du 24ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2023). Rennes (France), 20–23 Février, 2023.* 2 pages.

10. **Arbi S., Derrien A., Gurevsky E., Dolgui A., Siadat A.** Resource-constrained balancing of assembly lines to optimize their productivity and robustness in a reconfigurable environment. *24th Congress of the French Society of Operations Research and Decision Aid (ROADEF 2023). Rennes (France), February 20–23, 2023.* Abstracts, 2 pages.

11. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** An extended MILP formulation for the design of modular multi-model reconfigurable manufacturing lines. *23rd Congress of the French Society of Operations Research and Decision Aid (ROADEF 2022). Villeurbanne (France), February 23–25, 2022.* Abstracts, 2 pages.

12. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** A MILP-based heuristic for minimizing the number of reassignments under balancing multi-model reconfigurable manufacturing lines. *22nd Congress of the French Society of Operations Research and Decision Aid (ROADEF 2021). Mulhouse (France), April 26–30, 2021.* Abstracts, 2 pages.

13. **Yelles-Chaouche A.R., Gurevsky E., Brahimi N., Dolgui A.** Optimizing task reassignments in the design of reconfigurable manufacturing lines. *21st Congress of the French Society of Operations Research and Decision Aid (ROADEF 2020). Montpellier (France), February 19–21, 2020.* Abstracts, 2 pages.

14. **Schepler X., Dolgui A., Gurevsky E., Rossi A.** Résolution du problème de bin-packing robuste par un algorithme de branch-and-price. *Résumés du 21ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2020). Montpellier (France), 19–21 Février, 2020.* 2 pages.

15. **Pirogov A., Gurevsky E., Rossi A., Dolgui A.** Enhanced MILP approach for maximizing the robustness of paced transfer lines with uncertain parallel tasks and fixed number of machines. *19th Congress of the French Society of Operations Research and Decision Aid (ROADEF 2018). Lorient (France), February 21–23, 2018.* Abstracts, 2 pages.

16. **Schepler X., Rossi A., Gurevsky E., Dolgui A.** Robust scheduling of assembly lines with and without parallel workstations. *18th Congress of the French Society of Operations Research and Decision Aid (ROADEF 2017). Metz (France), February 22–24, 2017.* Abstracts, 2 pages.

17. **Gurevsky E., Kovalev S., Kovalyov M.Y.** Combinatorial optimization problems with risk functions, budget type constraints and controllable data. *17th Congress of the French Society of Operations Research and Decision Aid (ROADEF 2016). Compiègne (France), February 10–12, 2016.* Abstracts, p. 411–412.

18. **Rossi A., Gurevsky E., Battaïa O., Dolgui A.** Maximisation du rayon de stabilité pour l'affectation d'opérations de durée incertaine sur une ligne d'assemblage. *Résumés du 14ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2013). Troyes (France), 13–15 Février, 2013.* 2 pages.

19. **Gurevsky E., Battaïa O., Detienne B., Dolgui A.** Étude expérimentale de la sensibilité de configurations d'un système d'assemblage en présence de données incertaines. *Résumés du 13ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2012). Angers (France), 11–13 Avril, 2012.* P. 253–254.

20. **Gurevsky E., Hazır Ö, Dolgui A.** Équilibrage robuste flexible de SALBP-1 avec les données incertaines. *Résumés du 12ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2011). Saint-Étienne (France), 2–4 Mars, 2011.* P. 659–660.

21. **Gurevsky E., Guschinskaya O., Dolgui A.** Analyse de sensibilité pour le problème d'équilibrage des lignes d'assemblage de type SALBP-2. *Résumés du 11ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2010). Toulouse (France), 24–26 Février, 2010.* 2 pages.

22. **Gurevsky E.E., Emelichev V.A.** Regularizing of vector integer quadratic programming problems. *The International Scientific Conference «X Belarusian Mathematical Conference». Minsk (Belarus), November 3–7, 2008.* Abstracts. Part 5, p. 75–77. (in Russian)

23. **Gurevsky E.E., Emelichev V.A.** On stability conditions of some vector lexicographic Boolean problem. *The Russian Conference «Discrete Optimization and Operations Research» (DOOR-2007). Vladivostok (Russia), September 07–14, 2007.* Materials of the Conference, p. 102. (in Russian)

24. **Gurevsky E.E., Emelichev V.A.** Stability measure of an efficient solution of a multicriteria combinatorial partition problem. *The III All-Russian Conference «Optimization Problems and Economic Applications», dedicated to the 50th anniversary of the foundation of the Siberian Branch of RAS. Omsk (Russia), July 11–15, 2006.* Materials of the Conference, p. 85. (in Russian)

Communications sans publication (7)

1. **Gurevsky E.** A bi-objective optimization approach for balancing reconfigurable assembly lines subject to uncertainty and limited resources. *Seminar «Decision Support under Risk», Systems Research Institute, Polish Academy of Sciences. Warsaw (Poland), September 13, 2024.*

2. **Gurevsky E.** Optimality-based cut generation in MILP approaches for robust transfer line balancing. *Seminar «Decision Support under Risk», Systems Research Institute, Polish Academy of Sciences. Warsaw (Poland), May 11, 2023.*

3. **Gurevsky E.** Some new results for combinatorial problems with risk functions and controllable data. *Seminar «Aide à la Décision», LAMSADE, Université Paris-Dauphine. Paris (France), February 25, 2020.*

4. **Gurevsky E.** Some contributions to combinatorial problems of risk handling with budget constraint and controllable data. *Seminar «Applied Mathematics and Statistics», Department of Mathematics and Statistics, University of Turku. Turku (Finland), October 31, 2016.*

5. **Gurevsky E.** Production systems optimization. *Seminar «Logistic and Production Systems», IRCCyN, École des Mines de Nantes. Nantes (France), June 26, 2015.*

6. **Gurevsky E., Goisque G., Gandibleux X.** Inverse multiobjective 0-1 linear programming : evaluation of perturbations of the objective functions coefficients in the l_1-norm. *Workshop «GUEPARD». Paris (France), September 5–6, 2013.*

7. **Gurevsky E.** Régularisation d'un problème multicritère de programmation quadratique en nombres entiers en présence de données incertaines. *Séminaire «Analyse Non Linéaire et Optimisation», Laboratoire de Mathématiques d'Avignon, Université d'Avignon. Avignon (France), 21 Juin, 2012.*

Mémoires (2)

1. **Evgeny Gurevsky (2011).**
   Conception de lignes de fabrication sous incertitudes : analyse de sensibilité et approche robuste.
   Ph.D. in Industrial Engineering, École des Mines de Saint-Étienne, France.

Members of jury

President :	Nathalie Sauer	Professor, Université Paul Verlaine, Metz
Reviewers :	Jean-Charles Billaut	Professor, Université François Rabelais, Tours
	Mireille Jacomino	Professor, Institut Polytechnique de Grenoble
Examiners :	Mhand Hifi	Professor, Université de Picardie Jules Verne, Amiens
	Sylvie Norre	Professor, Université Blaise Pascal, Clermont-Ferrand
	André Rossi	Associate Professor, Université de Bretagne-Sud, Lorient
Advisor :	Alexandre Dolgui	Professor, École des Mines de Saint-Étienne
Co-advisor :	Olga Battaïa	Researcher, École des Mines de Saint-Étienne

2. **Evgeny Gurevsky (2008).**
   Multiobjective discrete optimization : sensitivity analysis and scalaring.
   M.S. in Mathematics, Belarusian State University, Minsk, Belarus. (in Russian)

Advisor

Vladimir A. Emelichev Professor, Belarusian State University, Minsk, Belarus

A.4 Encadrement doctoral et scientifique

Depuis mon arrivée à l'Université de Nantes, j'ai eu l'occasion de co-encadrer 1 post-doc, 5 thèses de doctorat (2 achevées avec succès et 3 en cours), 7 stages de fin d'études, 19 étudiants pour leur TER [22] en M1 et 6 étudiants pour leur stage en L3.

Ci-dessous, pour chaque thèse de doctorat, les co-encadrants sont indiqués avec leur taux de participation officiel et le nom de leur structure de recherche, s'ils ne font pas partie du LS2N. Le directeur de thèse est signalé par le symbole *.

Post-doc (1)

1. **Xavier Schepler** (novembre 2016 – août 2017)
   Problèmes de bin-packing avec objets de taille incertaine
   Co-encadr. avec : Alexandre Dolgui et André Rossi
   Financeur : Projet RFI AtlanSTIC 2020 «ADLACI»

Doctorants (5)

1. **Victor Spitzer** (mars 2023 – à présent)
   Planification de la production et gestion de l'approvisionnement en hydrogène vert en présence d'erreurs dans les prévisions de l'énergie éolienne et de prix incertains de l'électricité
   Co-encadr. avec : Céline Gicquel* (50%, LISN [23])
   Financeur : Projet CIFRE «Lhyfe»

2. **Emmanuelle Amann** (février 2023 – à présent)
   Lissage de la charge et optimisation des plans de tri du courrier dans la distribution postale
   Co-encadr. avec : Arnaud Laurent (30%) et Nasser Mebarki* (40%)
   Financeur : Projet CIFRE «La Poste»

3. **Siwar Arbi** (novembre 2022 – à présent)
   Conception évolutive de lignes de production reconfigurables
   Co-encadr. avec : Audrey Cerqueus (20%), Alexandre Dolgui* (40%) et Ali Siadat (20%, LCFC [24])
   Financeur : Projet ANR 2021 «ReconfiDurable»

4. **Abdelkrim R. Yelles-Chaouche** (novembre 2018 – décembre 2021)
   Outils d'aide à la décision pour la mise en place et la configuration dynamique des RMS
   Co-encadr. avec : Nadjib Brahimi (30%, RSB [25]) et Alexandre Dolgui* (40%)
   Financeur : Projet IRT PERFORM 2018

5. **Aleksandr Pirogov** (novembre 2016 – novembre 2019)
   Robust balancing of production lines : MILP models and pre-processing rules
   Co-encadr. avec : Alexandre Dolgui* (40%) et André Rossi (30%, LAMSADE [26])
   Financeur : Bourse Ministérielle «IMT Atlantique»

22. TER – Travail d'Étude et de Recherche.
23. LISN (UMR CNRS 9015) – Laboratoire Interdisciplinaire des Sciences du Numérique.
24. LCFC (EA 4495) – Laboratoire de Conception, Fabrication et Commande.
25. RSB – Rennes School of Business.
26. LAMSADE (UMR CNRS 7243) – Laboratoire d'Analyse et de Modélisation de Systèmes d'Aide à la DEcision.

Stages de fin d'études (7)

1. **Victor Spitzer** (6 mois, 2022)
   Optimisation de la production d'hydrogène vert sous incertitudes
   Master en Informatique, École Polytechnique, Palaiseau, France
   Co-encadr. avec : Paul Feliot, André Rossi et François Sanson
   Financeur : Entreprise «Lhyfe» à Nantes
   `Lauréat du Prix ROADEF pour le Meilleur Mémoire de Master en RO/AD en 2022`

2. **Wassim Garred** (6 mois, 2022)
   Balancing of resilient production lines
   Master en Génie Industriel, ENIT, Tunis, Tunisie
   Co-encadr. avec : Nadjib Brahimi, Alexandre Dolgui et Abdelkrim R. Yelles-Chaouche
   Financeur : Bourse «Rennes School of Business»

3. **Siwar Arbi** (6 mois, 2022)
   Design of RMS to optimize their productivity and robustness
   Master en Génie Industriel, ENIT, Tunis, Tunisie
   Co-encadr. avec : Audrey Cerqueus, Alexandre Dolgui et Ali Siadat
   Financeur : Projet ANR 2021 «ReconfiDurable»

4. **Emmanuelle Amann** (6 mois, 2022)
   Optimisation des plans de tri du courrier à La Poste
   Master en Informatique, Sorbonne Université, Paris, France
   Co-encadr. avec : Nasser Mebarki et Didier Thieriot
   Financeur : Société «La Poste» à Nantes

5. **Guillaume Balloud** (5 mois, 2021)
   Techniques de pré-traitement pour l'équilibrage robuste de lignes de transfert
   Master en Informatique, Université de Nantes, France
   Co-encadr. avec : Alexandre Dolgui et André Rossi
   Financeur : Bourse LS2N «Stage M2»

6. **David Tremblet** (5 mois, 2021)
   Équilibrage de lignes d'assemblage multi-modèles dans l'environnement reconfigurable
   Master en Informatique, Université de Nantes, France
   Co-encadr. avec : Nadjib Brahimi, Alexandre Dolgui et Abdelkrim R. Yelles-Chaouche
   Financeur : Projet IRT PERFORM 2018

7. **Ramzi Ben Mhenni** (5 mois, 2016)
   Approches de type «branch-and-bound» pour l'optimisation parcimonieuse en traitement du signal
   Master en Informatique, Université de Monastir, Tunisie
   Co-encadr. avec : Sébastien Bourguignon
   Financeur : Bourse LS2N «Stage M2»

A.5 Participation à des projets de recherche

Depuis mon arrivée à l'Université de Nantes, j'ai eu l'occasion de porter, co-porter ou bien participer à six projets scientifiques : 2 CIFRE, 1 ANR PRC, 1 IRT, 1 RFI et 1 CNRS PEPS.

1. **Mars 2023 – Février 2026**
   Optimisation sous incertitudes de la production et de la distribution d'hydrogène vert
   Financeur : CIFRE avec l'entreprise «Lhyfe» à Nantes
   Porteurs : Céline Gicquel (Université Paris-Saclay, Gif-sur-Yvette)
   Evgeny Gurevsky (Université de Nantes)
   Participants : Paul Feliot (Lhyfe, Nantes)
   François Sanson (Lhyfe, Nantes)
   Étudiant : Victor Spitzer
   Montant : Thèse + 30k€

   Le projet CIFRE avec l'entreprise «Lhyfe» est la suite logique du stage de fin d'études de Victor Spitzer, encadré initialement par Paul Feliot, moi-même, André Rossi et François Sanson. Le sujet principal de ce projet est de minimiser le coût de production de l'hydrogène obtenu par le procédé d'électrolyse de l'eau, en utilisant deux sources d'énergie intermittentes : l'électricité verte générée par le parc éolien en mer, dont le prix est négligeable, et le réseau électrique national, dont le prix est plus élevé. Trois types d'incertitude sont étudiés : l'incertitude liée aux erreurs de prévision de la puissance éolienne, l'incertitude sur le prix de l'électricité un jour à l'avance du réseau national, et l'incertitude sur la demande des clients. Les problèmes étudiés sont de type «unit commitment» et «lot-sizing». Les méthodes utilisées pour aborder ces problèmes sous incertitude sont la programmation stochastique et l'optimisation robuste couplées à des techniques d'apprentissage et d'analyse de données pour la génération plus fidèle de scénarios de comportement des paramètres incertains.

2. **Février 2023 – Janvier 2026**
   Conception d'un système dynamique de tri et de distribution du courrier : Application aux centres de tri postaux de nouvelle génération de La Poste
   Financeur : CIFRE avec la société «La Poste» à Nantes
   Porteurs : Evgeny Gurevsky (Université de Nantes)
   Nasser Mebarki (Université de Nantes)
   Participants : Thierry Auger (La Poste, Nantes)
   Arnaud Laurent (Université de Nantes)
   Didier Thieriot (La Poste, Nantes)
   Étudiante : Emmanuelle Amann
   Montant : Thèse + 60k€

   Le projet CIFRE avec la société «La Poste» est né suite au stage de fin d'études d'Emmanuelle Amann, encadrée par moi-même, Nasser Mebarki et Didier Thieriot. L'un des objectifs de ce projet CIFRE est de revoir l'approche actuelle de la planification du tri du courrier à La Poste, afin de l'améliorer et de la rendre plus dynamique. Les problèmes étudiés sont essentiellement des variantes du problème de «bin-packing» avec soit des contraintes de précédence particulières sous la forme de plusieurs chaînes indépendantes soit des objets de type vectoriel.

3. **Février 2022 – Janvier 2026**
   Conception et pilotage de systèmes de production reconfigurables et durables
   Financeur : ANR PRC 2021 (ANR-21-CE10-0019)
   Acronyme : «ReconfiDurable»
   Coordinateur : Alexandre Dolgui (IMT Atlantique, Nantes)
   Participants : Olga Battaïa (Kedge Business School, Talence)
   Lyès Benyoucef (Université d'Aix-Marseille)
   Audrey Cerqueus (IMT Atlantique, Nantes)
   Guillaume Delcroix (AI France, Grenoble)
   Xavier Delorme (Mines Saint-Étienne)
   Paolo Gianessi (Mines Saint-Étienne)
   Evgeny Gurevsky (Université de Nantes)
   Hichem Haddou-Benderbal (Université d'Aix-Marseille)
   Damien Lamy (Mines Saint-Étienne)
   Ali Siadat (ENSAM, Metz)
   Simon Thevenin (IMT Atlantique, Nantes)
   Montant : 579k€

   En 2021, nous avons eu l'opportunité de décrocher un projet ANR PRC 2021, appelé «ReconfiDurable», pour une durée de 48 mois à partir de février 2022. Le but de ce projet est d'intégrer les deux niveaux de décision, à savoir la phase de conception et le niveau opérationnel, afin de concevoir des systèmes de production reconfigurables (RMS) et de fournir un ensemble de reconfigurations du système initial pour anticiper les perturbations de production (pannes de machines, changements de la demande, problèmes de qualité, etc). L'efficacité de ce système repose sur sa productivité et ses coûts de reconfiguration.

   Le budget du projet est généralement dédié au payement des salaires de 3 post-docs, 2 doctorants et 5 stages de fin d'études. J'ai déjà participé à l'encadrement du stage de fin d'études de Siwar Arbi, et je continue à co-encadrer sa thèse de doctorat. Dans le cadre du même projet, le co-encadrement d'un post-doc est également prévu en 2025.

4. **Novembre 2018 – Octobre 2021**
   Outils d'aide à la décision pour la mise en place et la configuration dynamique des RMS
   Financeur : IRT PERFORM 2018
   Porteur : Alexandre Dolgui (IMT Atlantique, Nantes)
   Participants : Nadjib Brahimi (Rennes School of Business)
   Evgeny Gurevsky (Université de Nantes)
   Montant : Thèse + 30k€

   En 2018, nous avons réussi à obtenir un nouveau projet dans le cadre du programme «PERFORM», financé par l'Institut de Recherche Technologique Jules Verne (IRT Jules Verne), basé à Bouguenais près de Nantes. Ce projet de type académique aborde des problèmes fortement combinatoires de gestion optimale d'équipements modulaires lors de la conception et la mise en œuvre de lignes de production reconfigurables (RMS).

   La majeure partie du financement du projet a été dédiée à payer le salaire de notre doctorant Abdelkrim R. Yelles-Chaouche. Les 30k€ restants ont servi à financer nos déplacements pour assister à des conférences scientifiques.

5. **Juillet 2016 – Décembre 2017**
   Aide à la décision pour la conception de lignes d'assemblage en contexte incertain
   Financeur : RFI AtlanSTIC 2020
   Acronyme : «ADLACI»
   Porteur : André Rossi (Université d'Angers)
   Co-porteurs : Alexandre Dolgui (IMT Atlantique, Nantes)
   Evgeny Gurevsky (Université de Nantes)
   Montant : 42,9k€

   Ce projet s'est inscrit dans la continuité du travail collaboratif avec Alexandre Dolgui et André Rossi que nous menons ensemble depuis 2012 sur la thématique «Équilibrage robuste de lignes de production en contexte incertain». Les fonds du projet ont été principalement utilisés pour payer le salaire de 10 mois de notre post-doc Xavier Schepler.

6. **Mars 2015 – Décembre 2015**
   Approche par optimisation pour l'extraction de mots-clés
   Financeur : CNRS PEPS INS2I/INSMI «Fondements et Applications de la SD [27]»
   Porteur : Florian Boudin (Université de Nantes)
   Participant : Evgeny Gurevsky (Université de Nantes)
   Montant : 6k€

   Travailler avec Florian Boudin sur un appel à projets CNRS PEPS a été ma toute première participation dans ce type de processus académique (*i.e.* la recherche de financement par projet), mais aussi ma toute première collaboration avec une autre équipe de laboratoire, à savoir l'équipe TALN (Traitement Automatique du Langage Naturel).

   L'objectif de ce projet était de considérer le problème de l'extraction de mots-clés dans le domaine du traitement du langage comme un problème d'optimisation, et de le résoudre en utilisant des outils de recherche opérationnelle comme la recherche locale.

A.6 Animation scientifique

Depuis ma thèse de doctorat, j'ai pris une part très active à l'animation et à la diffusion d'événements scientifiques, que ce soit en participant à des comités de programme ou à des comités d'organisation de conférences, ou en organisant des sessions invitées. J'ai également pu effectuer quatre visites scientifiques de courte durée (une semaine chacune) dans des universités étrangères, donner six séminaires invités et être sollicité pour participer en tant qu'examinateur à deux jurys de thèse. À noter aussi l'attribution récente d'une prime RIPEC C3 pour une période de 3 ans à compter du 1er octobre 2022.

Membre du comité d'organisation (3)

1. 10th IFAC Conference on Manufacturing Modeling, Management and Control (MIM 2022). Nantes (France), June 22–24, 2022.

2. IFIP International Conference on Advances in Production Management Systems (APMS 2021). Nantes (France), September 5–9, 2021.

3. 12ème Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2011). Saint-Étienne (France), 2–4 Mars, 2011.

27. SD – Science des Données.

Membre du comité de programme (4)

1. 23rd International Conference «Mathematical Optimization Theory and Operations Research» (MOTOR 2024). Omsk (Russia), June 30 – July 06, 2024.

2. 21st International Conference «Mathematical Optimization Theory and Operations Research» (MOTOR 2022). Petrozavodsk (Russia), July 2–6, 2022.

3. 19th International Conference «Mathematical Optimization Theory and Operations Research» (MOTOR 2020). Novosibirsk (Russia), July 06–10, 2020.

4. 7th International Conference «Optimization Problems and Their Applications» (OPTA-2018). Omsk (Russia), July 08–14, 2018.

Chair/Co-chair de session (8)

1. Invited session «Production Optimization and Supply Chain Management of Green Hydrogen under Uncertainties» for the Stream «Energy Markets» at the 33rd European Conference on Operational Research (EURO 2024). Copenhagen (Denmark), June 30 – July 03, 2024.

2. Two invited sessions «Assembly Line Balancing and Scheduling Problems» at the 10th IFAC Conference on Manufacturing Modeling, Management and Control (MIM 2022). Nantes (France), June 22–24, 2022.

3. Invited session «Workforce Scheduling and Line Balancing» for the Stream «Scheduling and Project Management» at the 31th European Conference on Operational Research (EURO 2021). Athens (Greece), July 11–14, 2021.

4. Two invited sessions «Workforce Scheduling and Line Balancing» for the Stream «Project Management and Scheduling» at the 29th European Conference on Operational Research (EURO 2018). Valencia (Spain), July 08–11, 2018.

5. Invited session for the Stream «Combinatorial Optimization» at the 28th European Conference on Operational Research (EURO 2016). Poznań (Poland), July 3–6, 2016.

6. Session at the Workshop «Optimization in Operations Research» (ORO'2013). Nantes (France), December 13, 2013.

7. Invited session «Robust Combinatorial Optimization : Challenges, Solution Techniques, and Applications» at the 12th Congress of the French Society of Operations Research and Decision Aid (ROADEF 2011). Saint-Étienne (France), March 2–4, 2011.

8. Session «Production Control» at the 5th IFAC International Conference on Management and Control of Production and Logistics (MCPL 2010). Coimbra (Portugal), September 08–10, 2010.

Visites scientifiques (4)

Polish Academy of Sciences, Warsaw, Poland
Decision Support under Risk Group, Systems Research Institute *09.2024 (1 week) + 05.2023 (1 week)*
Host : Dr. Dmitry Podkopaev

University of Turku, Finland
Turku Optimization Group, Turku Centre for Computer Science *10.2016 (1 week)*
Host : Dr. Yury Nikulin

University of Leeds, United Kingdom
Algorithms and Complexity Group, School of Computing *05.2016 (1 week)*
Host : Dr. Natalia V. Shakhlevich

Séminaires invités (6)

1. Seminar «Decision Support under Risk», Systems Research Institute, PAS [28].
   Warsaw (Poland), September 13, 2024.
   Warsaw (Poland), May 11, 2023.

2. Seminar «Aide à la Décision», LAMSADE, Université Paris-Dauphine.
   Paris (France), February 25, 2020.

3. Seminar «Applied Mathematics and Statistics», TUCS, University of Turku.
   Turku (Finland), October 31, 2016.

4. Seminar «Logistic and Production Systems», IRCCyN, École des Mines de Nantes.
   Nantes (France), June 26, 2015.

5. Seminar «Analyse Non Linéaire et Optimisation», LMA, Université d'Avignon.
   Avignon (France), June 21, 2012.

Participations à des jurys de thèse (Examinateur) (2)

1. **Damien Lovato** (21 novembre 2022, IRIT, Université de Toulouse)
   Développement de critères d'optimisation pour la planification de tâches dans une chaîne d'assemblage aéronautique
   Spécialité : Mathématiques Appliquées

2. **Alena Melnikava** (28 novembre 2016, LIA, Université d'Avignon)
   Optimization methods for density of human activity in the perspective of a sustainable urban form
   Spécialité : Informatique

A.7 Activités d'enseignement

Depuis le début de ma carrière à l'Université de Nantes, j'enseigne à tous les niveaux de formation au sein du Département d'Informatique (ci-après DPT-INFO) de l'UFR Sciences et Techniques (ci-après UFR ST), allant du L1 au M2. Il ne s'agit pas seulement d'un engagement de ma part, mais d'un choix volontaire qui me semble important, car l'accompagnement des étudiants tout au long de leur cursus me permet à la fois de bénéficier d'une vision globale de l'offre de formation et surtout d'analyser sa cohérence.

28. PAS – Polish Academy of Sciences.

L'ensemble des unités d'enseignement (ci-après UE) que je dispense régulièrement au sein du DPT-INFO est présenté ci-dessous sous le format de tableau avec leur niveau et leur volume horaire. Je suis responsable des trois premières UE. À noter également la responsabilité des stages de fin d'études depuis 2013 (et sans interruption) pour le parcours «Optimisation en Recherche Opérationnelle» (ci-après ORO) du Master Informatique.

Unité d'enseignement	**Niveau**	**CM**	**TD**	**TP**
Optimisation sous Incertitude	M1/M2 ORO	12h	8h	4h
Ingénierie de la Décision	M1 Info	12h	12h	-
Recherche Opérationnelle	L3 Maths-Éco	12h	12h	12h
Architecture des Ordinateurs	L3 Info	-	16h	12h
Algorithmique et Structures de Données 3	L3 Info	-	16h	12h
Algorithmique et Structures de Données 2	L2 Info	-	24h	8h
Algorithmique et Structures de Données 1	L2 Info	-	-	8h
Programmation Orientée Objet	L2 Info	-	-	12h
Algorithmique et Programmation	L1 Info	-	16h	12h
Bases de Données	L1 Info	-	-	12h

A.8 Responsabilités collectives et d'intérêt général

Je dresse ici la liste de mes responsabilités administratives les plus importantes depuis mon arrivée à l'Université de Nantes.

— **Janvier 2019 – Décembre 2024** : *3 mandats de 2 ans chacun*
Membre Invité au Conseil du DPT-INFO de l'UFR ST de l'Université de Nantes

— **Janvier 2019 – Décembre 2024** : *3 mandats de 2 ans chacun*
Directeur Adjoint (Chargé RH) du DPT-INFO de l'UFR ST de l'Université de Nantes

— **Janvier 2017 – à présent** : *2ème mandat en cours de 5 ans chacun*
Membre de l'ARE [29] du laboratoire LS2N (UMR CNRS 6004)

— **Janvier 2017 – à présent** : *2ème mandat en cours de 5 ans chacun*
Responsable Adjoint de l'équipe MODELIS du laboratoire LS2N (UMR CNRS 6004)

— **Mars 2014 – Février 2018** : *2 mandats de 2 ans chacun*
Membre Élu au Conseil du DPT-INFO de l'UFR ST de l'Université de Nantes

29. ARE – Assemblée des Responsables d'Équipe.

Bibliographie

Álvarez-Miranda, E., Chen, X., Hu, J., Hu, X., and Candia-Véjar, A. (2011). Deterministic risk control for cost-effective network connections. *Theoretical Computer Science*, 412(3) :257–264.

Arcus, A. L. (1965). A computer method of sequencing operations for assembly lines. *International Journal of Production Research*, 4(4) :259–277.

Ashraf, M. and Hasan, F. (2018). Configuration selection for a reconfigurable manufacturing flow line involving part production with operation constraints. *The International Journal of Advanced Manufacturing Technology*, 98(5-8) :2137–2156.

Ağpak, K. and Gökçen, H. (2007). A chance-constrained approach to stochastic line balancing problem. *European Journal of Operational Research*, 180(3) :1098–1115.

Baldwin, C. Y. and Clark, K. B. (2006). Modularity in the design of complex engineering systems. In *Understanding Complex Systems*, pages 175–205. Springer Berlin Heidelberg.

Battaïa, O. and Dolgui, A. (2013). A taxonomy of line balancing problems and their solution approaches. *International Journal of Production Economics*, 142(2) :259–277.

Battaïa, O., Gurevsky, E., Makssoud, F., and Dolgui, A. (2012). Equipment location in machining transfer lines with multi-spindle heads. *Journal of Mathematical Modelling and Algorithms in Operations Research*, 12(2) :117–133.

Battaïa, O. and Dolgui, A. (2022). Hybridizations in line balancing problems : A comprehensive review on new trends and formulations. *International Journal of Production Economics*, 250 :108673.

Battini, D., Faccio, M., Ferrari, E., Persona, A., and Sgarbossa, F. (2006). Design configuration for a mixed-model assembly system in case of low product demand. *The International Journal of Advanced Manufacturing Technology*, 34(1-2) :188–200.

Bayrak, H. I., Kargar, K., and Pınar, M. Ç. (2019). Bilateral trade with risk-averse intermediary using linear network optimization. *Networks*, 74(4) :325–332.

Belmokhtar, S., Dolgui, A., Guschinsky, N., and Levin, G. (2006). Integer programming models for logical layout design of modular machining lines. *Computers & Industrial Engineering*, 51(3) :502–518.

Berger, I., Bourjolly, J.-M., and Laporte, G. (1992). Branch-and-bound algorithms for the multi-product assembly line balancing problem. *European Journal of Operational Research*, 58(2) :215–222.

Borisovsky, P., Delorme, X., and Dolgui, A. (2014). Balancing reconfigurable machining lines via a set partitioning model. *International Journal of Production Research*, 52(13) :4026–4036.

Bozkaya, B., Salman, F. S., and Telciler, K. (2017). An adaptive and diversified vehicle routing approach to reducing the security risk of cash-in-transit operations. *Networks*, 69(3) :256–269.

Brauner, N., Gurevsky, E., and Kovalyov, M. Y. (2024). A single representative min-max-min robust selection problem with alternatives and budgeted uncertainty. *Discrete Applied Mathematics*, 349 :106–112.

Bryan, A., Jack Hu, S., and Koren, Y. (2013). Assembly system reconfiguration planning. *Journal of Manufacturing Science and Engineering*, 135(4) :13 pages.

Bukchin, J. and Rubinovitz, J. (2003). A weighted approach for assembly line design with station paralleling and equipment selection. *IIE Transactions*, 35(1) :73–85.

Bukchin, J. and Tzur, M. (2000). Design of flexible assembly line to minimize equipment cost. *IIE Transactions*, 32(7) :585–598.

Bukchin, Y. and Rabinowitch, I. (2006). A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task duplication costs. *European Journal of Operational Research*, 174(1) :492–508.

Chassein, A. and Goerigk, M. (2021). On the complexity of min-max-min robustness with two alternatives and budgeted uncertainty. *Discrete Applied Mathematics*, 296 :141–163.

Chazelle, B. (2000). A minimum spanning tree algorithm with inverse-ackermann type complexity. *Journal of the ACM*, 47(6) :1028–1047.

Chen, X., Hu, J., and Hu, X. (2009a). A new model for path planning with interval data. *Computers & Operations Research*, 36(6) :1893–1899.

Chen, X., Hu, J., and Hu, X. (2009b). A polynomial solvable minimum risk spanning tree problem with interval data. *European Journal of Operational Research*, 198(1) :43–46.

Cheng, T. C. E., Janiak, A., and Kovalyov, M. Y. (1998). Bicriterion single machine scheduling with resource dependent processing times. *SIAM Journal on Optimization*, 8(2) :617–630.

Cheng, T. C. E., Kovalyov, M. Y., and Shakhlevich, N. V. (2006). Scheduling with controllable release dates and processing times : Makespan minimization. *European Journal of Operational Research*, 175(2) :751–768.

Dantzig, G. B. (1957). Discrete-variable extremum problems. *Operations Research*, 5(2) :266–277.

Desrochers, M. (1986). *La fabrication d'horaires de travail pour les conducteurs d'autobus par une méthode de génération de colonnes*. PhD thesis, Centre de Recherche sur les Transports, Université de Montréal, Canada.

Dolgui, A. and Kovalev, S. (2012). Scenario based robust line balancing : Computational complexity. *Discrete Applied Mathematics*, 160(13-14) :1955–1963.

Dou, J., Dai, X., and Meng, Z. (2009). Optimisation for multi-part flow-line configuration of reconfigurable manufacturing system using GA. *International Journal of Production Research*, 48(14) :4071–4100.

Dou, J., Li, J., and Su, C. (2016). Bi-objective optimization of integrating configuration generation and scheduling for reconfigurable flow lines using NSGA-II. *The International Journal of Advanced Manufacturing Technology*, 86(5-8) :1945–1962.

Erel, E. and Gokcen, H. (1999). Shortest-route formulation of mixed-model assembly line balancing problem. *European Journal of Operational Research*, 116(1) :194–204.

Gabow, H. N. and Tarjan, R. E. (1984). Efficient algorithms for a family of matroid intersection problems. *Journal of Algorithms*, 5(1) :80–131.

Gen, M., Tsujimura, Y., and Li, Y. (1996). Fuzzy assembly line balancing using genetic algorithms. *Computers & Industrial Engineering*, 31(3-4) :631–634.

Graves, S. C. and Redfield, C. H. (1988). Equipment selection and task assignment for multiproduct assembly system design. *International Journal of Flexible Manufacturing Systems*, 1(1) :31–50.

Grigoriev, A., Sviridenko, M., and Uetz, M. (2007). Machine scheduling with resource dependent processing times. *Mathematical Programming, Series B*, 110(1) :209–228.

Gurevsky, E., Battaïa, O., and Dolgui, A. (2012). Balancing of simple assembly lines under variations of task processing times. *Annals of Operations Research*, 201(1) :265–286.

Gurevsky, E., Battaïa, O., and Dolgui, A. (2013a). Stability measure for a generalized assembly line balancing problem. *Discrete Applied Mathematics*, 161(3) :377–394.

Gurevsky, E., Hazır, O., Battaïa, O., and Dolgui, A. (2013b). Robust balancing of straight assembly lines with interval task times. *Journal of the Operational Research Society*, 64(1) :1607–1613.

Gurevsky, E., Kopelevich, D., Kovalev, S., and Kovalyov, M. Y. (2022a). Min-sum controllable risk problems with concave risk functions of the same value range. *Networks*, 79(1) :105–116.

Gurevsky, E., Kopelevich, D., Kovalev, S., and Kovalyov, M. Y. (2023). Integer knapsack problems with profit functions of the same value range. *4OR : A Quarterly Journal of Operations Research*, 21(3) :405–419.

Gurevsky, E., Kovalev, S., and Kovalyov, M. Y. (2021). Min-max controllable risk problems. *4OR : A Quarterly Journal of Operations Research*, 19(1) :93–101.

Gurevsky, E., Rasamimanana, A., Pirogov, A., Dolgui, A., and Rossi, A. (2022b). Stability factor for robust balancing of simple assembly lines under uncertainty. *Discrete Applied Mathematics*, 318 :113–132.

Hop, N. V. (2006). A heuristic solution for fuzzy mixed-model line balancing problem. *European Journal of Operational Research*, 168(3) :798–810.

Hu, J. (2010). Minimizing maximum risk for fair network connection with interval data. *Acta Mathematicae Applicatae Sinica, English Series*, 26(1) :33–40.

Huang, C.-C. and Kusiak, A. (1998). Modularity in design of products and systems. *IEEE Transactions on Systems, Man, and Cybernetics - Part A : Systems and Humans*, 28(1) :66–77.

Janiak, A. and Kovalyov, M. Y. (1996). Single machine scheduling subject to deadlines and resource dependent processing times. *European Journal of Operational Research*, 94(2) :284–291.

Janiak, A., Kovalyov, M. Y., Kubiak, W., and Werner, F. (2005). Positive half-products and scheduling with controllable processing times. *European Journal of Operational Research*, 165(2) :416–422.

Joksch, H. C. (1966). The shortest route problem with constraints. *Journal of Mathematical Analysis and Application*, 14(2) :191–197.

Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Ulsoy, G., and Van Brussel, H. (1999). Reconfigurable manufacturing systems. In *Manufacturing Technologies for Machines of the Future*, pages 627–665. Springer Berlin Heidelberg.

Kouvelis, P. and Yu, G. (1997). *Robust Discrete Optimization and Its Applications*, volume 14 of *Nonconvex Optimization and Its Applications*. Springer US.

Lapierre, S. D., Debargis, L., and Soumis, F. (2000). Balancing printed circuit board assembly line systems. *International Journal of Production Research*, 38(16) :3899–3911.

Le Carrer, N., Ferson, S., and Green, P. L. (2020). Optimising cargo loading and ship scheduling in tidal areas. *European Journal of Operational Research*, 280(3) :1082–1094.

Nazarian, E., Ko, J., and Wang, H. (2010). Design of multi-product manufacturing lines with the consideration of product change dependent inter-task times, reduced changeover and machine flexibility. *Journal of Manufacturing Systems*, 29(1) :35–46.

Ng, C. T., Cheng, T. C. E., and Kovalyov, M. Y. (2003). Batch scheduling with controllable setup and processing times to minimize total completion time. *Journal of the Operational Research Society*, 54(5) :499–506.

Otto, A., Otto, C., and Scholl, A. (2013). Systematic data generation and test design for solution algorithms on the example of SALBPGen for assembly line balancing. *European Journal of Operational Research*, 228(1) :33–45.

Özcan, U. (2010). Balancing stochastic two-sided assembly lines : A chance-constrained, piecewise-linear, mixed integer program and a simulated annealing algorithm. *European Journal of Operational Research*, 205(1) :81–97.

Patterson, J. H. and Albracht, J. J. (1975). Assembly-line balancing : Zero-one programming with Fibonacci search. *Operations Research*, 23(1) :166–172.

Pereira, J. (2018). Modelling and solving a cost-oriented resource-constrained multi-model assembly line balancing problem. *International Journal of Production Research*, 56(11) :3994–4016.

Pirogov, A., Gurevsky, E., Rossi, A., and Dolgui, A. (2021). Robust balancing of transfer lines with blocks of uncertain parallel tasks under fixed cycle time and space restrictions. *European Journal of Operational Research*, 290(3) :946–955.

Rekiek, B., Dolgui, A., Delchambre, A., and Bratcu, A. (2002). State of art of optimization methods for assembly line design. *Annual Reviews in Control*, 26(2) :163–174.

Richards, C., Odom, C., Morton, D., and Newman, A. (2018). Minimum-risk routing through a mapped minefield. *Networks*, 73(3) :358–376.

Rossi, A., Gurevsky, E., Battaïa, O., and Dolgui, A. (2016). Maximizing the robustness for simple assembly lines with fixed cycle time and limited number of workstations. *Discrete Applied Mathematics*, 208 :123–136.

Saxena, L. K. and Jain, P. K. (2012). A model and optimisation approach for reconfigurable manufacturing system configuration design. *International Journal of Production Research*, 50(12) :3359–3381.

Schepler, X., Rossi, A., Gurevsky, E., and Dolgui, A. (2022). Solving robust bin-packing problems with a branch-and-price approach. *European Journal of Operational Research*, 297(3) :831–843.

Scholl, A. (1999). *Balancing and Sequencing of Assembly Lines*. Contributions to Management Science. Springer-Verlag Berlin Heidelberg, 2 edition.

Scholl, A. and Becker, C. (2006). State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research*, 168(3) :666–693.

Shabtay, D. and Steiner, G. (2007). A survey of scheduling with controllable processing times. *Discrete Applied Mathematics*, 155 :1643–1666.

Shaik, A. M., Kesava Rao, V. V. S., and Srinivasa Rao, C. (2015). Development of modular manufacturing systems – a review. *The International Journal of Advanced Manufacturing Technology*, 76(5-8) :789–802.

Shibasaki, R. S., Rossi, A., and Gurevsky, E. (2024). A new upper bound based on dantzig-wolfe decomposition to maximize the stability radius of a simple assembly line under uncertainty. *European Journal of Operational Research*, 313(3) :1015–1030.

Shioura, A., Shakhlevich, N. V., and Strusevich, V. A. (2015). Decomposition algorithms for submodular optimization with applications to parallel machine scheduling with controllable processing times. *Mathematical Programming, Series A*, 153(2) :495–534.

Sotskov, Y., Dolgui, A., and Portmann, M.-C. (2006). Stability analysis of an optimal balance for an assembly line with fixed cycle time. *European Journal of Operational Research*, 168(3) :783–797.

Sotskov, Y., Dolgui, A., Sotskova, N., and Werner, F. (2005). Stability of optimal line balance with given station set. In Dolgui, A., Soldek, J., and Zaikin, O., editors, *Supply Chain Optimisation : Product/Process Design, Facility Location and Flow Control*, volume 94 of *Applied Optimization*, chapter 10, pages 135–149. Springer US.

Tóth, A., Knuutila, T., and Nevalainen, O. S. (2018). Machine configuration and workload balancing of modular placement machines in multi-product PCB assembly. *International Journal of Computer Integrated Manufacturing*, 31(9) :815–830.

Tremblet, D., Yelles-Chaouche, A. R., Gurevsky, E., Brahimi, N., and Dolgui, A. (2023). Optimizing task reassignments for reconfigurable multi-model assembly lines with unknown order of product arrival. *Journal of Manufacturing Systems*, 67 :190–200.

Tsujimura, Y., Gen, M., and Kubota, E. (1995). Solving fuzzy assembly-line balancing problem with genetic algorithms. *Computers & Industrial Engineering*, 29(1-4) :543–547.

Urban, T. and Chiang, W.-C. (2006). An optimal piecewise-linear program for the U-line balancing problem with stochastic task times. *European Journal of Operational Research*, 168(3) :771–782.

van Zante-de Fokkert, J. I. and de Kok, T. G. (1997). The mixed and multi model line balancing problem : a comparison. *European Journal of Operational Research*, 100(3) :399–412.

Yelles-Chaouche, A. R., Gurevsky, E., Brahimi, N., and Dolgui, A. (2021). Reconfigurable manufacturing systems from an optimisation perspective : a focused review of literature. *International Journal of Production Research*, 59(21) :6400–6418.

Yelles-Chaouche, A. R., Gurevsky, E., Brahimi, N., and Dolgui, A. (2022). Minimizing task reassignments under balancing multi-product reconfigurable manufacturing lines. *Computers & Industrial Engineering*, 173 :108660.

Yelles-Chaouche, A. R., Gurevsky, E., Brahimi, N., and Dolgui, A. (2024). Optimizing modular equipment in the design of multi-product reconfigurable production lines. *Computers & Industrial Engineering*, 192 :110226.

Youssef, A. M. A. and ElMaraghy, H. A. (2006). Modelling and optimization of multiple-aspect RMS configurations. *International Journal of Production Research*, 44(22) :4929–4958.

Youssef, A. M. A. and ElMaraghy, H. A. (2007). Optimal configuration selection for reconfigurable manufacturing systems. *International Journal of Flexible Manufacturing Systems*, 19(2) :67–106.

Youssef, A. M. A. and ElMaraghy, H. A. (2008). Availability consideration in the optimal selection of multiple-aspect RMS configurations. *International Journal of Production Research*, 46(21) :5849–5882.

Zacharia, P. T. and Nearchou, A. C. (2012). Multi-objective fuzzy assembly line balancing using genetic algorithms. *Journal of Intelligent Manufacturing*, 23(3) :615–627.

Résumé

Ce manuscrit retrace mes activités de recherche ainsi que l'encadrement scientifique que j'ai pu réaliser depuis mon recrutement à l'Université de Nantes en 2012. Il est principalement structuré autour de trois contributions scientifiques qui se trouvent elles-mêmes à l'intersection de thématiques telles que la recherche opérationnelle, les systèmes de production, l'incertitude et le risque. Ainsi, le chapitre 1 est dédié à l'équilibrage robuste de lignes de transfert avec des tâches dont la durée est incertaine. Pour pouvoir aborder ce problème, nous nous sommes appuyés sur un certain nombre de techniques de pré-traitement, ainsi que sur la génération de coupes fondées sur le principe d'optimalité pour le programme linéaire en nombres mixtes correspondant. Le chapitre 2, quant à lui, se focalise sur le problème de l'équilibrage de lignes de production reconfigurables avec des équipements modulaires. Pour affronter efficacement ce problème, nous l'avons décomposé en deux sous-problèmes. Le premier consiste à générer intelligemment l'ensemble des modules réalisables à partir desquels la ligne peut être construite. Le second est un programme linéaire en nombres binaires (PLNB) basé sur cet ensemble de modules pré-générés. Afin de résoudre exactement ce PLNB, une technique de génération de contraintes dites «lentes» a été mise en œuvre. Finalement, le chapitre 3 se concentre sur l'étude de la complexité algorithmique d'une classe de problèmes d'optimisation combinatoire avec des données contrôlables et des fonctions de risque, qui trouvent des applications dans des domaines tels que le routage ou la conception de réseaux fiables. Pour certaines variantes de ces problèmes, une approche de réduction originale a été développée, nous permettant de proposer des algorithmes polynomiaux significativement plus rapides et plus efficaces (en termes d'ordres de grandeur) que dans la littérature.

Enfin, ce manuscrit se termine par quelques perspectives de recherche pour les années à venir et une annexe présentant mon CV actuel.

Mots-clés: recherche opérationnelle, systèmes de production, modularité, reconfigurabilité, équilibrage, incertitude, robustesse, rayon de stabilité, risque, contrôlabilité, PLNM, pré-traitement, génération de contraintes, complexité.

Abstract

This manuscript describes my research activities and the scientific supervision that I have been able to carry out since my joining the University of Nantes in 2012. It is mainly structured around three scientific contributions, which are themselves at the intersection of topics such as operations research, production systems, uncertainty and risk. Thus, Chapter 1 is dedicated to the robust balancing of transfer lines with tasks of uncertain processing time. To address this problem, we rely on a certain number of pre-processing techniques, as well as on the generation of optimality-based cuts for its corresponding mixed-integer linear program. Chapter 2, in its turn, focuses on the problem of balancing reconfigurable production lines with modular equipment. To tackle this problem efficiently, we have decomposed it into two sub-problems. The first consists in intelligently generating the set of feasible modules from which the line can be designed. The second is a 0-1 linear program based on this set of pre-generated modules. In order to solve it exactly, the so-called lazy constraint generation technique was implemented. Finally, Chapter 3 deals with the study of the algorithmic complexity of a class of combinatorial optimization problems with controllable data and risk functions, which find applications in fields such as routing or the design of reliable networks. For certain variants of these problems, an original reduction approach has been developed, enabling us to propose polynomial time algorithms that are significantly faster and more efficient than those found in the literature.

To conclude, this manuscript provides some research perspectives for the coming years and an appendix presenting my current CV.

Keywords: operations research, production systems, modularity, reconfigurability, balancing, uncertainty, robustness, stability radius, risk, controllability, MILP, pre-processing, constraint generation, complexity.