



NNT : 2011 EMSE 0634

THÈSE

présentée par

Evgeny GUREVSKY

pour obtenir le grade de
Docteur de l'École Nationale Supérieure des Mines de Saint-Étienne

Spécialité : Génie Industriel

Conception de lignes de fabrication sous incertitudes : analyse de
sensibilité et approche robuste

soutenue à Saint-Étienne, le 13 décembre 2011

Membres du jury

Présidente :	Nathalie Sauer	Professeur, Université Paul-Verlaine, Metz
Rapporteurs :	Jean-Charles Billaut	Professeur, Université François-Rabelais, Tours
	Mireille Jacomino	Professeur, Institut Polytechnique de Grenoble
Examineurs :	Mhand Hifi	Professeur, Université de Picardie Jules-Verne, Amiens
	Sylvie Norre	Professeur, Université Blaise Pascal, Clermont-Ferrand
	André Rossi	Maître de Conférences, Université de Bretagne-Sud, Lorient
Directeur de thèse :	Alexandre Dolgui	Professeur, École des Mines de Saint-Étienne
Co-encadrante :	Olga Battaïa	Chargée de Recherche, École des Mines de Saint-Étienne

Spécialités doctorales :

SCIENCES ET GENIE DES MATERIAUX
 MECANIQUE ET INGENIERIE
 GENIE DES PROCEDES
 SCIENCES DE LA TERRE
 SCIENCES ET GENIE DE L'ENVIRONNEMENT
 MATHEMATIQUES APPLIQUEES
 INFORMATIQUE
 IMAGE, VISION, SIGNAL
 GENIE INDUSTRIEL
 MICROELECTRONIQUE

Responsables :

J. DRIVER Directeur de recherche – Centre SMS
 A. VAUTRIN Professeur – Centre SMS
 F. GRUY Professeur – Centre SPIN
 B. GUY Maître de recherche – Centre SPIN
 J. BOURGOIS Professeur – Fayol
 E. TOUBOUL Ingénieur – Fayol
 O. BOISSIER Professeur – Fayol
 JC. PINOLI Professeur – Centre CIS
 P. BURLAT Professeur – Fayol
 Ph. COLLOT Professeur – Centre CMP

Enseignants-chercheurs et chercheurs autorisés à diriger des thèses de doctorat (titulaires d'un doctorat d'État ou d'une HDR)

AVRIL	Stéphane	MA	Mécanique & Ingénierie	CIS
BATTON-HUBERT	Mireille	MA	Sciences & Génie de l'Environnement	Fayol
BENABEN	Patrick	PR 1	Sciences & Génie des Matériaux	CMP
BERNACHE-ASSOLLANT	Didier	PR 0	Génie des Procédés	CIS
BIGOT	Jean-Pierre	MR	Génie des Procédés	SPIN
BILAL	Essaïd	DR	Sciences de la Terre	SPIN
BOISSIER	Olivier	PR 1	Informatique	Fayol
BORBELY	Andras	MR	Sciences et Génie des Matériaux	SMS
BOUCHER	Xavier	MA	Génie Industriel	Fayol
BOUDAREL	Marie-Reine	PR 2	Génie Industriel	DF
BOURGOIS	Jacques	PR 0	Sciences & Génie de l'Environnement	Fayol
BRODHAG	Christian	DR	Sciences & Génie de l'Environnement	Fayol
BURLAT	Patrick	PR 2	Génie industriel	Fayol
COLLOT	Philippe	PR 1	Microélectronique	CMP
COURNIL	Michel	PR 0	Génie des Procédés	SPIN
DAUZERE-PERES	Stéphane	PR 1	Génie industriel	CMP
DARRIEULAT	Michel	IGM	Sciences & Génie des Matériaux	SMS
DECHOMETS	Roland	PR 1	Sciences & Génie de l'Environnement	Fayol
DESRAYAUD	Christophe	MA	Mécanique & Ingénierie	SMS
DELAFOSSÉ	David	PR 1	Sciences & Génie des Matériaux	SMS
DOLGUI	Alexandre	PR 0	Génie Industriel	Fayol
DRAPIER	Sylvain	PR 2	Mécanique & Ingénierie	SMS
DRIVER	Julian	DR 0	Sciences & Génie des Matériaux	SMS
FEILLET	Dominique	PR 2	Génie Industriel	CMP
FOREST	Bernard	PR 1	Sciences & Génie des Matériaux	CIS
FORMISYN	Pascal	PR 1	Sciences & Génie de l'Environnement	Fayol
FRACZKIEWICZ	Anna	DR	Sciences & Génie des Matériaux	SMS
GARCIA	Daniel	MR	Sciences de la terre	SPIN
GIRARDOT	Jean-Jacques	MR	Informatique	Fayol
GOEURIOT	Dominique	MR	Sciences & Génie des Matériaux	SMS
GRAILLOT	Didier	DR	Sciences & Génie de l'Environnement	Fayol
GROSSEAU	Philippe	MR	Génie des Procédés	SPIN
GRUY	Frédéric	MR	Génie des Procédés	SPIN
GUY	Bernard	MR	Sciences de la Terre	SPIN
GUYONNET	René	DR	Génie des Procédés	SPIN
HERRI	Jean-Michel	PR 2	Génie des Procédés	SPIN
INAL	Karim	PR 2	Microélectronique	CMP
KLÖCKER	Helmut	DR	Sciences & Génie des Matériaux	SMS
LAFOREST	Valérie	CR	Sciences & Génie de l'Environnement	Fayol
LERICHE	Rodolphe	CR CNRS	Mécanique et Ingénierie	SMS
LI	Jean-Michel	EC (CCI MP)	Microélectronique	CMP
MALLIARAS	George Grégory	PR 1	Microélectronique	CMP
MOLIMARD	Jérôme	MA	Mécanique et Ingénierie	SMS
MONTHEILLET	Frank	DR 1 CNRS	Sciences & Génie des Matériaux	SMS
PERIER-CAMBY	Laurent	PR 2	Génie des Procédés	SPIN
PIJOLAT	Christophe	PR 1	Génie des Procédés	SPIN
PIJOLAT	Michèle	PR 1	Génie des Procédés	SPIN
PINOLI	Jean-Charles	PR 0	Image, Vision, Signal	CIS
ROUSTANT	Olivier	MA	Mathématiques Appliquées	Fayol
STOLARZ	Jacques	CR	Sciences & Génie des Matériaux	SMS
SZAFNICKI	Konrad	MR	Sciences & Génie de l'Environnement	Fayol
THOMAS	Gérard	PR 0	Génie des Procédés	SPIN
TRIA	Assia		Microélectronique	CMP
VALDIVIESO	François	MA	Sciences & Génie des Matériaux	SMS
VIRICELLE	Jean-Paul	MR	Génie des procédés	SPIN
WOLSKI	Krzysztof	DR	Sciences & Génie des Matériaux	SMS
XIE	Xiaolan	PR 1	Génie industriel	CIS

Glossaire :

PR 0	Professeur classe exceptionnelle
PR 1	Professeur 1 ^{ère} classe
PR 2	Professeur 2 ^{ème} classe
MA(MDC)	Maître assistant
DR	Directeur de recherche
Ing.	Ingénieur
MR(DR2)	Maître de recherche
CR	Chargé de recherche
EC	Enseignant-chercheur
IGM	Ingénieur général des mines

Centres :

SMS	Sciences des Matériaux et des Structures
SPIN	Sciences des Processus Industriels et Naturels
Fayol	Institut Henri Fayol
CMP	Centre de Microélectronique de Provence
CIS	Centre Ingénierie et Santé

Remerciements

Tout d'abord, je tiens à remercier Monsieur Alexandre Dolgui, mon directeur de thèse et Professeur à l'École des Mines de Saint-Étienne, pour m'avoir fait confiance et m'avoir permis de réaliser cette thèse dans d'excellentes conditions. J'ai apprécié son encadrement, ses conseils et son soutien à tout moment.

Je voudrais aussi exprimer toute ma gratitude à Madame Olga Battaïa, ma co-encadrante et Chargée de Recherche à l'École des Mines de Saint-Étienne. Je lui suis particulièrement reconnaissant pour sa disponibilité, pour toutes les discussions que nous avons pu avoir tout au long de ces trois années, et pour toutes ses remarques qui ont permis d'améliorer la qualité de mes travaux.

Je remercie également Monsieur Xavier Delorme, Maître-Assistant à l'École des Mines de Saint-Étienne, qui a contribué à l'aboutissement de ce travail et pour les nombreuses discussions scientifiques que nous avons pu avoir.

J'exprime ma profonde gratitude aux membres du jury qui m'ont fait l'honneur d'accepter d'évaluer mon travail. Je remercie donc tout d'abord Madame Nathalie Sauer, Professeur à l'Université Paul-Verlaine, d'avoir acceptée de présider le jury de cette thèse. Ensuite, j'exprime toute ma reconnaissance à Madame Mireille Jacomino, Professeur à l'Institut Polytechnique de Grenoble, et à Monsieur Jean-Charles Billaut, Professeur à l'Université François-Rabelais, pour leur critique constructive, pour leur lecture attentive, et pour d'avoir accepté de rapporter ma thèse. Je remercie également Madame Sylvie Norre, Professeur à l'Université Blaise Pascal, Monsieur Mhand Hifi, Professeur à l'Université de Picardie Jules-Verne, et Monsieur André Rossi, Maître de Conférences à l'Université de Bretagne-Sud, de m'avoir fait l'honneur d'examiner mon travail.

Enfin, un grand merci à tous mes collègues et thésards du Centre G2I pour leur amitié, leur soutien et leur aide pendant ces années de thèse.

À mes parents

Table des matières

Introduction générale	11
1 Prise en compte de l'incertitude en recherche opérationnelle	15
1.1 Modélisation de l'incertitude	15
1.2 Approches robustes	17
1.3 Analyse de sensibilité	21
1.4 Conclusion	26
2 Conception de lignes de fabrication	27
2.1 Lignes de fabrication	27
2.2 Problématique de l'équilibrage de lignes	29
2.3 Équilibrage sous incertitudes	33
2.4 Conclusion	36
3 Équilibrage de lignes d'assemblage de type SALBP-E : analyse de sensibilité	37
3.1 Introduction	37
3.2 Définitions et propriétés de base	38
3.3 Exemple	43
3.4 Solutions faisables	44
3.5 Solutions quasi-faisables	46
3.6 Solutions optimales	47
3.7 Recherche du compromis	54
3.8 Résultats numériques	57
3.9 Conclusion	60

4	Équilibrage de lignes d'usinage à boîtiers multibroches : analyse de sensibilité	63
4.1	Introduction	63
4.2	Définitions et propriétés de base	65
4.3	Solutions faisables	68
4.4	Solutions quasi-faisables	72
4.5	Solutions optimales	78
4.6	Recherche du compromis	80
4.7	Résultats numériques	81
4.8	Conclusion	83
5	Équilibrage de lignes d'assemblage de type SALBP-1 : approche robuste	93
5.1	Introduction	93
5.2	Définition du problème	94
5.2.1	Modèle déterministe	94
5.2.2	Modèle robuste	95
5.3	Algorithme par séparation et évaluation	96
5.4	Résultats numériques	98
5.5	Conclusion	101
	Conclusion	105
	Bibliographie	109

Table des figures

2.1	Architecture linéaire homogène.	28
2.2	Ligne de fabrication bilatérale.	28
2.3	Ligne de fabrication à stations parallèles.	29
3.1	Modélisation des perturbations.	41
3.2	Contraintes de précédence.	43
3.3	Solution s^1	43
3.4	Solution s^2	43
3.5	Comparaison de deux approches pour l'instance de Gunther.	62
4.1	Configuration d'une ligne d'usinage à boîtiers multibroches.	65
4.2	Lignes d'usinage à boîtiers multibroches.	65
4.3	Le bloc l de la station k de la solution s : Δ_{kl}	68
4.4	Le bloc l de la station k de la solution s : $\hat{\Delta}_{kl}$	74
4.5	Valeur moyenne de \mathcal{N} en fonction du pourcentage d'opérations incertaines.	84
4.6	Valeur moyenne ρ_{moy} en fonction du pourcentage d'opérations incertaines.	84
4.7	$\mathcal{ASN}\mathcal{D}$ pour le Test 6 de la Série 1 et pour le Test 5 de la Série 2 avec 90% d'opérations incertaines.	85
5.1	Les contraintes de précédence pour l'instance de Jackson.	100
5.2	Arbre énumératif de RLA pour l'instance de Jackson.	101
5.3	Le temps de résolution pour l'instance de Heskia.	104
5.4	Le temps de résolution moyen pour les instances de grande taille.	104
5.5	Évolution du nombre moyen de stations pour les trois séries.	104

Liste des tableaux

3.1	Instances	59
3.2	Première approche	60
3.3	Deuxième approche	61
3.4	Comparaison des approches	62
4.1	Instances	82
4.2	Résultats pour la Série 1 (10%-30%)	86
4.3	Résultats pour la Série 1 (40%-60%)	87
4.4	Résultats pour la Série 1 (70%-90%)	88
4.5	Résultats pour la Série 2 (10%-30%)	89
4.6	Résultats pour la Série 2 (40%-60%)	90
4.7	Résultats pour la Série 2 (70%-90%)	91
5.1	Instances	99
5.2	Le temps de résolution (sec) pour les instances de taille moyenne	102
5.3	Le temps de résolution (sec) pour les instances de grande taille	103
5.4	Évolution du nombre de stations pour les instances de petite taille	103
5.5	Évolution du nombre de stations pour les instances de taille moyenne	103
5.6	Évolution du nombre de stations pour les instances de grande taille	103

Introduction générale

La conception des lignes de fabrication est un problème industriel important dont les enjeux sont la réussite de la mise sur le marché de nouveaux produits et la bonne santé de l'entreprise. Par conséquent, la prise de décisions à cette étape ne peut être faite à la légère et doit être bien argumentée. Dans l'idéal, les méthodes de la recherche opérationnelle doivent être utilisées pour modéliser les problèmes stratégiques qui en relèvent. Comme en pratique les concepteurs ne disposent que d'estimations pour les données de départ, les modèles utilisés doivent tenir compte du fait que certaines données peuvent avoir, au moment de la mise en œuvre du système, des valeurs différentes de celles prévues au point de remettre en cause la solution retenue à l'étape de résolution. Même si la modélisation de l'incertitude est un sujet bien développé en recherche opérationnelle, très peu de travaux ont été consacrés à la conception des lignes de fabrication. Pour combler cette lacune, nous abordons cette problématique dans cette thèse. Plus concrètement, nous étudions la conception des lignes d'assemblage et d'usinage sous incertitudes en nous fixant les trois objectifs suivants :

1. Proposer des outils pour la modélisation de l'incertitude des données dans le contexte de la conception des lignes de fabrication.
2. Définir de nouveaux critères pour les problèmes d'optimisation étudiés qui ne correspondent pas forcément au critère d'optimalité.
3. Développer des méthodes de résolution capables de trouver des solutions qui garantissent la performance souhaitée de la ligne à concevoir.

Ce mémoire est organisé en cinq chapitres.

Dans le chapitre 1, nous donnons un aperçu historique de la modélisation de l'incertitude en recherche opérationnelle des premières approches jusqu'à l'état de l'art actuel. Une attention particulière est portée sur les deux approches suivantes : l'approche robuste et l'analyse de sensibilité. Une synthèse concernant les applications de ces deux approches pour les problèmes d'optimisation combinatoire est notamment présentée.

Dans le chapitre 2, nous décrivons le contexte de la conception des systèmes de fabrication et les problèmes d'optimisation combinatoires qui en relèvent. L'accent est mis

sur les lignes d'assemblage et les lignes d'usinage pour lesquelles les problèmes d'équilibrage sous incertitude sont formulés dans les Chapitres 3, 4, 5. Les lignes d'assemblage considérées sont constituées d'une chaîne de stations, où dans chaque station, les opérations d'assemblage s'exécutent de manière séquentielle. En revanche, les lignes d'usinage traitées sont composées des machines de transfert comportant un ou plusieurs boîtiers multibroches où les opérations s'exécutent simultanément. Un aperçu des méthodes déterministes existantes dans la littérature pour la conception de telles lignes est donné.

Dans le chapitre 3, nous étudions le problème de conception de lignes d'assemblage dont la version déterministe est connue sous le nom de SALBP-E (Simple Assembly Line Balancing Problem-Efficiency). Ce problème consiste à affecter un ensemble d'opérations à une séquence de stations en tenant compte des contraintes de précédence. Chaque opération nécessite un temps d'exécution dont la valeur est connue. L'objectif est de minimiser à la fois le nombre de stations utilisées et la somme totale des temps opératoires affectés à une station. Dans cette thèse, nous considérons le cas où les temps opératoires sont susceptibles de varier. Pour cette formulation du problème, nous menons, tout d'abord, l'analyse de sensibilité des solutions à ces variations et proposons d'utiliser le rayon de stabilité en tant que mesure de leur performance. Le rayon de stabilité d'une solution est défini comme l'amplitude maximale des perturbations des temps d'exécution d'opérations pour lesquelles la solution préserve une propriété particulière, par exemple, de faisabilité ou d'optimalité. Ensuite, le problème est posé comme un problème bicritère ayant pour premier but la minimisation de l'ancienne fonction objectif et pour deuxième but, la maximisation du rayon de stabilité de la solution. Enfin, nous proposons une approche pour la recherche des solutions non-dominées au sens de Pareto pour le problème ainsi posé avec les deux critères évoqués.

Dans le chapitre 4, nous étudions le problème de conception des lignes d'usinage à boîtiers multibroches. Comme pour le problème précédent, il s'agit d'affecter un ensemble d'opérations à des stations de travail. La particularité de ce problème réside dans le fait que les opérations affectées à une station sont regroupées en blocs de sorte que toutes les opérations se trouvant dans un bloc commun soient exécutées simultanément. Ceci est possible grâce à l'utilisation de boîtiers multibroches. Une autre différence par rapport au problème considéré dans le chapitre 3 vient de la présence de contraintes d'exclusion interdisant l'exécution dans un même bloc ou sur une même station des opérations technologiquement incompatibles. La fonction objectif est également différente et vise à minimiser le coût de la ligne calculé comme le coût total des stations et des boîtiers multibroches utilisés. Pour ce problème d'optimisation combinatoire, connu dans la littérature sous le nom de TLBP (Transfer Line Balancing Problem), nous admettons également que les opérations sont sujettes à des variations de temps d'exécution et proposons l'analyse

de sensibilité des solutions optimales, faisables et quasi-faisables. Ensuite, une nouvelle formulation est proposée, dans laquelle deux critères d'optimisation sont considérés : d'une part, la minimisation du coût de la ligne et, d'autre part, la maximisation du rayon de stabilité de la solution. Une méthode construisant des solutions non-dominées au sens de Pareto est développée pour ce problème bi-objectif.

Dans le chapitre 5, un autre problème de conception de lignes de fabrication, connu dans la littérature sous le nom de SALBP-1, est étudié. Ce problème consiste à affecter des opérations à des stations de travail lors de la conception d'une ligne d'assemblage en minimisant le nombre de stations nécessaires. Une analyse de sensibilité pour ce problème dans le cas où les temps d'exécution de certaines des opérations seraient susceptibles de changer a été présentée dans [Sotskov *et al.*, 2006]. Nous introduisons une nouvelle formulation de ce problème où les temps opératoires sont donnés par des intervalles de valeurs possibles. Pour résoudre ce nouveau problème, nous proposons une approche robuste basée sur la méthode par séparation et évaluation. La méthode proposée est évaluée sur un échantillon des instances recensées dans la littérature.

La conclusion générale présentée à la fin de ce mémoire fait le bilan des travaux effectués et des résultats obtenus. Les perspectives pour des recherches futures sont également évoquées.

Chapitre 1

Prise en compte de l'incertitude en recherche opérationnelle

Sommaire

1.1	Modélisation de l'incertitude	15
1.2	Approches robustes	17
1.3	Analyse de sensibilité	21
1.4	Conclusion	26

1.1 Modélisation de l'incertitude

Lors de la résolution d'un problème d'optimisation, les décideurs sont souvent confrontés à l'incertitude des données concernant un système réel ou son comportement. Intégrer cette information incertaine dans un modèle mathématique du problème ne fait qu'augmenter sa complexité, mais la négliger peut faire perdre toute crédibilité au modèle.

L'information imparfaite dans les applications réelles peut être présente, selon Sanlaville, sous les trois formes suivantes [Sanlaville, 2005] :

- **incertitude.** Une donnée est soumise à une incertitude si sa valeur ne peut être déterminée de façon précise qu'au moment exact où cette donnée intervient. Par exemple, en ordonnancement, la durée d'une tâche ne peut être connue de façon certaine que lorsque celle-ci est réellement exécutée.
- **aléa.** Une donnée est sujette aux aléas si la valeur qui lui est associée a priori est susceptible d'être modifiée par certains événements. Par exemple, le temps d'exécution d'une tâche peut être allongé par des micro-arrêts de la machine qui la réalise.
- **variation.** Une donnée est soumise à des variations si sa valeur nominale change

durant la phase d'approbation. Par exemple, si une tâche est réalisée par un opérateur, l'effet d'apprentissage peut raccourcir son temps d'exécution.

Pour faire face à l'incertitude des données, les approches suivantes peuvent être appliquées selon Mahjoub [Mahjoub, 2004] :

- **a priori**. Une solution pour le problème d'optimisation initial est préalablement calculée sur la base des données estimées. Une fois cette solution calculée, elle n'est plus remise en cause.
- **en-ligne**. Le problème ne peut être résolu que lorsque les données sont disponibles complètement ou partiellement.
- **partiellement en-ligne**. Les deux approches précédentes sont combinées. Ainsi les données disponibles a priori sont utilisées pour calculer une solution préalable, mais cette solution peut évoluer en fonction des perturbations des données.

Notons que dans [Davenport and Beck, 2000], les auteurs préfèrent parler d'approches prédictives, réactives, et prédictives réactives, respectivement.

Puisque les approches en-ligne se basent sur les données réelles, leur application est impossible pour de nombreux problèmes stratégiques, pour lesquels la prise de décisions ne peut être reportée au moment où toutes les données seront connues. Souvent en pratique, seules les approches a priori peuvent être appliquées dans ce cas. Par conséquent, il est important d'intégrer les perturbations possibles dans la modélisation des données de départ afin d'obtenir des solutions préalables pouvant être efficaces pour plusieurs scénarios possibles du futur. Ainsi dans la littérature, plusieurs approches ont été développées pour modéliser l'incertitude des données ou pour guider le décideur lors de la prise de décisions sous incertitudes :

- Les **approches probabilistes** considèrent les données incertaines comme variables aléatoires de lois connues.
- Les **approches par ensembles flous** supposent que chaque donnée incertaine appartient à un sous-ensemble flou $A \subseteq X$ avec le degré d'appartenance qui se mesure par la fonction $\mu_A : X \rightarrow [0, 1]$, où X est un ensemble universel.
- Les **approches robustes** associent à une donnée incertaine un ensemble, continu ou discret, de valeurs possibles, sans probabilité attachée.
- L'**analyse de sensibilité** introduit l'incertitude après la phase de résolution du problème. Ainsi à la différence des approches précédentes, il s'agit d'étudier le comportement des solutions déjà trouvées face aux variations éventuelles des données de départ et non pas de résoudre le problème posé. D'ailleurs, dans le cas général, l'évaluation de l'impact de telles variations sur une solution est possible sans que le problème initial soit résolu.

Notons qu'en pratique l'application des deux premières approches est souvent difficile,

puisque les connaissances a priori ne permettent pas toujours de déduire les lois de probabilité ou les degrés d'appartenance associés aux données, surtout si le problème posé est traité pour la première, et peut-être unique, fois. Pour cette raison, l'approche robuste et l'analyse de sensibilité sont de plus en plus utilisées pour intégrer l'incertitude dans les problèmes d'optimisation combinatoire. Dans ce qui suit, nous présentons ces approches de façon plus approfondie.

1.2 Approches robustes

Évoquée dès la fin des années 1960 [Gupta and Rosenhead, 1968], l'idée de robustesse suscite un intérêt croissant à la fois de la part des praticiens et des théoriciens en recherche opérationnelle. Reflétant initialement une préoccupation de flexibilité dans un contexte d'incertitude, ce concept paraît aujourd'hui s'adapter à un spectre beaucoup plus large de situations comme l'indique Bernard Roy dans [Billaut *et al.*, 2005, chapitre 2].

Dans la littérature concernant la robustesse, une donnée incertaine est principalement associée à un ensemble, continu ou discret, de valeurs possibles, sans probabilité attachée [Kouvelis and Yu, 1997]. Dans le cas continu, les ensembles sont souvent des intervalles, d'où la notion d'approche par intervalles. Dans le cas discret, quant à lui, on parle du modèle de scénarios. Quel que soit le cas considéré, le produit cartésien de ces ensembles définit les instances possibles d'un problème.

Après avoir modélisé les données incertaines sous forme d'instances possibles, le problème est donc de trouver une solution qui soit « bonne » pour toutes ces instances ou robuste. Pour la définition d'une solution robuste, nous utilisons celle donnée dans [Billaut *et al.*, 2005, chapitre 1] :

Définition 1.1. *Une solution est robuste si sa performance est peu sensible à l'incertitude des données et aux aléas.*

Suivant cette définition, une solution robuste est alors conçue pour limiter ou absorber les effets des données incertaines. En pratique, on considère souvent que la solution robuste est la meilleure solution dans le pire cas. La plupart des travaux concernant la robustesse utilisent le concept de distance entre la solution robuste calculée a priori et la solution optimale de l'instance réalisée effectivement suivant un certain critère (satisfaction de contraintes ou une exigence particulière).

Pour la recherche des solutions robustes, la plupart des auteurs utilisent les critères de la robustesse absolue (min-max), du regret maximal (min-max regret) ou bien du regret relatif [Kouvelis and Yu, 1997]. Soit pour un problème de minimisation, si \mathcal{S} est l'ensemble des solutions possibles, \mathcal{I} est l'ensemble des instances, $f(s, I)$ donne la valeur

de la solution $s \in \mathcal{S}$ pour l'instance $I \in \mathcal{I}$ et f_I^* la valeur optimale pour l'instance $I \in \mathcal{I}$, alors les trois critères sont définis comme suit :

- **robustesse absolue** :

$$\min_{s \in \mathcal{S}} \max_{I \in \mathcal{I}} f(s, I),$$

- **regret maximal** :

$$\min_{s \in \mathcal{S}} \max_{I \in \mathcal{I}} (f(s, I) - f_I^*),$$

- **regret relatif** :

$$\min_{s \in \mathcal{S}} \max_{I \in \mathcal{I}} \frac{f(s, I) - f_I^*}{f_I^*}.$$

Ces critères, et surtout le deuxième, ont été largement appliqués à différents problèmes d'optimisation comme le problème du sac-à-dos, le problème d'affectation, les problèmes du plus court chemin et de l'arbre couvrant de poids minimal, ou des problèmes d'ordonnement. Dans ce qui suit, nous présentons une synthèse de ces travaux.

Problème du sac-à-dos. Dans sa version déterministe classique, ce problème consiste à sélectionner, parmi un ensemble fini d'objets, un sous-ensemble d'objets à mettre dans un sac d'une capacité limitée. Chaque objet a un poids et une valeur et on cherche à maximiser la valeur totale des objets choisis. Ce problème, qui est \mathcal{NP} -difficile [Garey and Johnson, 1979], peut être résolu en temps pseudo-polynomial à l'aide de la programmation dynamique [Dantzig, 1957].

Supposons que la valeur de chaque objet n'est pas connue de façon certaine. Pour le cas où ces valeurs sont modélisées par un nombre constant de scénarios, Yu [Yu, 1996] a proposé un algorithme pseudo-polynomial et un algorithme de type séparation et évaluation pour la version max-min. Pour le cas où les données sont modélisées par intervalles, Averbakh [Averbakh, 2004] a démontré que la version min-max regret du problème de sac-à-dos est \mathcal{NP} -difficile au sens fort.

Problème d'affectation. Étant donné un graphe biparti complet caractérisé par le même nombre n de sommets dans chaque partie et des coûts associés à chaque arête, le problème d'affectation vise à construire une affectation ayant un coût minimum. Une affectation optimale peut être calculée pour le cas déterministe en temps $\mathcal{O}(n^3)$ avec l'algorithme de Balinski [Balinski, 1985].

En utilisant une réduction polynomiale à partir du problème de partition, Kouvelis et Yu [Kouvelis and Yu, 1997] ont démontré que les versions min-max du problème d'affectation sont \mathcal{NP} -difficiles, même pour deux scénarios de coûts. Toutefois, l'existence d'algorithmes pseudo-polynomiaux pour ces problèmes est une question ouverte. Dans le cas de la modélisation par intervalles, la version min-max regret est \mathcal{NP} -difficile au sens fort [Aissi *et al.*, 2005].

Problème du plus court chemin. Considérons un graphe orienté avec n sommets, où deux parmi eux correspondent aux sommets origine et destination. Chaque arc possède une longueur. Le problème du plus court chemin consiste donc à trouver un chemin de longueur minimale entre l'origine et la destination. Dans le cas déterministe, ce problème peut être résolu en $\mathcal{O}(n^2)$ en utilisant l'algorithme de Dijkstra [Dijkstra, 1959].

Kouvelis et Yu [Kouvelis and Yu, 1997] ont étudié la complexité des versions min-max du problème du plus court chemin pour le cas d'un nombre constant de scénarios. Ils ont démontré que ces versions sont \mathcal{NP} -difficiles, même avec deux scénarios. Plus tard, des algorithmes pseudo-polynomiaux ont été proposés pour ces versions dans [Yu and Yang, 1998].

Quant à la version min-max regret dans le cas de la modélisation par intervalles, elle est \mathcal{NP} -difficile au sens fort [Averbakh and Lebedev, 2004]. Pour cette version, un Programme Linéaire Mixte (MIP) a été développé et testé sur des instances de différentes tailles dans [Karaşan *et al.*, 2001]. Un algorithme plus efficace de type séparation et évaluation a été présenté dans [Montemanni *et al.*, 2004].

Problème de l'arbre couvrant de poids minimal. Étant donné un graphe connexe, avec n sommets, m arêtes et des poids associés à chaque arête, le problème est de trouver un sous-graphe connexe et sans cycle de poids total minimal. Un arbre couvrant optimal peut être obtenu, par exemple, en temps $\mathcal{O}(n^2 + m \log n)$ en utilisant l'algorithme de Prim [Prim, 1957] ou en temps $\mathcal{O}(m \log m)$ en utilisant l'algorithme de Kruskal [Kruskal, 1976].

Kouvelis et Yu [Kouvelis and Yu, 1997] ont démontré que les versions min-max (regret) du problème de l'arbre couvrant de poids minimal sont \mathcal{NP} -difficiles, même pour deux scénarios. Ils ont également conclu que des algorithmes pseudo-polynomiaux peuvent être proposés uniquement pour une classe restreinte de graphes : les grilles.

Averbakh et Lebedev [Averbakh and Lebedev, 2004] ont démontré que la version min-max regret du problème d'arbre couvrant de poids minimal est \mathcal{NP} -difficile au sens fort dans le cas de la modélisation par intervalles. Un modèle MIP pour cette version a été développé dans [Yaman *et al.*, 2001] où les auteurs ont également proposé quelques règles de pré-traitement. Un algorithme plus efficace de type séparation et évaluation a été présenté dans [Montemanni and Gambardella, 2005].

Problèmes d'ordonnancement. La plupart des travaux en ordonnancement portant sur des critères de robustesse traitent des problèmes simples où un ensemble de n tâches doit être exécuté séquentiellement, sans préemption, sur une machine. Chaque tâche i se caractérise par un temps d'exécution p_i et, de façon facultative, par une date d'échéance d_i à partir de laquelle la tâche est considérée en retard. En utilisant les notations de Graham [Graham *et al.*, 1979], les problèmes traités peuvent être représentés de

la manière suivante :

- $1||\sum C_i$ – minimisation de la somme des dates de fin des tâches,
- $1||\sum U_i$ – minimisation du nombre de tâches en retard,
- $1||L_{\max}$ – minimisation du retard algébrique maximum.

Ici C_i est la date de fin de la tâche i , U_i égal 1 si $C_i > d_i$ et 0 sinon, $L_{\max} = \max_i\{C_i - d_i\}$.

La version déterministe du problème $1||\sum C_i$ peut être résolu en temps $O(n \log n)$ en utilisant la règle SPT (Shortest Processing Time) [Smith, 1956]. Cependant, si les temps d'exécution des tâches sont incertains, ce problème est \mathcal{NP} -difficile pour les trois critères de robustesse, même avec deux scénarios de temps d'exécution des tâches [Yang and Yu, 2002]. Pour la version min-max regret, [Lebedev and Averbakh, 2006] ont obtenu le même résultat pour le cas de la modélisation par intervalles. Toutefois, l'existence d'algorithmes pseudo-polynomiaux pour ces problèmes reste une question ouverte.

Le problème déterministe $1||\sum U_i$ peut être résolu en temps $O(n \log n)$ par une variante de la règle EDD (Earliest Due Date) proposée par Moore [Moore, 1968]. La version min-max de ce problème est \mathcal{NP} -difficile si les temps d'exécution des tâches sont incertains, même pour deux scénarios [Aloulou and Della Croce, 2008]. Dans [Aissi *et al.*, 2011], les auteurs ont considéré la situation où les temps d'exécution des tâches sont déterministes, mais leurs dates d'échéance sont incertaines et modélisées par scénarios. Ils ont prouvé que la version min-max regret de ce problème est polynomiale dans le cas où le nombre de scénarios et le nombre de dates d'échéance sont constants. Mahjoub [Mahjoub, 2004] a étudié le problème de la minimisation du nombre de tâches en retard avec une période d'indisponibilité située au début d'une durée inconnue. Le critère employé était le min-max regret. Il a proposé un algorithme dont le temps d'exécution est de l'ordre $O(n^2)$ pour le cas où seuls deux scénarios suivants sont pris en compte : la machine est immédiatement disponible ou bien la durée d'indisponibilité est connue. Un autre algorithme a été proposé pour le cas général, où l'ensemble de scénarios est discret ou continu, dont le temps d'exécution est de l'ordre $O(n^4)$.

Dans [Kasperski, 2005], l'auteur a considéré le problème de la minimisation du retard algébrique maximum en présence des contraintes de précédence $1|prec|L_{\max}$ dont la version déterministe peut être résolue en temps $O(n^2)$ avec l'algorithme de Lawler [Lawler, 1973]. Pour la version min-max regret où les temps des tâches sont modélisés par intervalles, l'auteur a proposé un algorithme polynomial avec un temps d'exécution de l'ordre $O(n^4)$.

Cette brève analyse nous permet de conclure que la plupart des problèmes combinatoires classiques qui sont polynomiaux dans leur version déterministe deviennent \mathcal{NP} -difficiles sous incertitudes. En ce qui concerne les critères utilisés, nous pouvons constater que les critères min-max et min-max regret ne s'appuient que sur le pire cas et,

par conséquent, peuvent fournir des solutions non adaptées aux cas les plus probables de l'évolution du problème. Pour contourner ce désavantage, Bertsimas et Sim [Bertsimas and Sim, 2003], qui traitent des problèmes de programmation mathématique, ont proposé une approche différente. Une solution est dite robuste au sens de Bertsimas et Sim si sa valeur est optimale parmi les solutions qui sont réalisables lorsqu'au plus Γ_i paramètres dans chaque contrainte i sont autorisés à dévier de leur valeur nominale située au milieu de l'intervalle de valeurs possibles. Ainsi seul un sous-ensemble de paramètres incertains peut influencer négativement la solution. Le point fort de l'approche de Bertsimas et Sim est que la version robuste du problème reste polynomiale si le problème initial l'est. Cette approche a été appliquée à plusieurs problèmes d'optimisation en gestion de stocks [Bertsimas and Thiele, 2006], ordonnancement de projets [Hazır *et al.*, 2011], optimisation de portefeuille [Moon and Yao, 2011] et planification de production [Alem and Morabito, 2012]. Dans le chapitre 5 de ce mémoire, nous adaptions l'approche de Bertsimas et Sim au problème d'équilibrage d'une ligne d'assemblage.

1.3 Analyse de sensibilité

L'analyse de sensibilité consiste à étudier l'impact des variations des données initiales d'un problème sur ses solutions. Ainsi il ne s'agit pas de résoudre un problème d'optimisation, mais d'évaluer le comportement des solutions déjà trouvées face à des perturbations des données initiales. De cette façon, l'incertitude est introduite dans la phase après la résolution du problème.

L'analyse de sensibilité a été initialement utilisée pour évaluer l'influence des modifications d'un des paramètres des problèmes de programmation linéaire résolus à l'aide de l'algorithme du simplexe. Le but était d'analyser le comportement de la fonction objectif sans devoir résoudre le problème à nouveau. L'analyse de sensibilité pour des problèmes de l'optimisation combinatoire est apparue un peu plus tard. Les tentatives pour employer directement les méthodes développées pour les problèmes linéaires continus sur ces problèmes n'ont pas abouti à un résultat satisfaisant. Ainsi depuis les années 1970, de nombreux travaux ont été consacrés à l'étude de sensibilité pour les problèmes d'optimisation combinatoire. Le premier état de l'art a été publié par Geoffrion et Nauss en 1977 [Geoffrion and Nauss, 1977].

À l'heure actuelle, l'étude de sensibilité en optimisation combinatoire se décline en deux approches essentielles : l'**approche quantitative** et l'**approche qualitative**. La première approche cherche à évaluer diverses mesures de sensibilité pour chaque solution et à proposer des algorithmes efficaces pour leur calcul. La deuxième approche étudie les propriétés structurelles, combinatoires ou géométriques des solutions.

Les mesures de sensibilité les plus considérées dans l'approche quantitative sont : l'**intervalle de sensibilité** et le **rayon de stabilité**. Voici leurs définitions.

Si on note \mathcal{P} un problème d'optimisation, $c = (c_1, c_2, \dots, c_n) \in \mathbb{R}^n$ un vecteur des paramètres de ce problème, x une solution étudiée du problème \mathcal{P} qui possède une propriété α (notée $x \in \alpha(\mathcal{P})$), \mathcal{P}' une version modifiée du problème \mathcal{P} avec les paramètres perturbés $c' = (c'_1, c'_2, \dots, c'_n) \in \mathbb{R}^n$. Alors, on dit que $[\underline{c}_k, \bar{c}_k]$ est un **intervalle de sensibilité** (ou intervalle de tolérance) pour le paramètre c_k , $k \in \mathcal{N} = \{1, 2, \dots, n\}$, si la formule suivante se vérifie :

$$\forall c' \in \mathcal{C}(k) \quad (x \in \alpha(\mathcal{P}')),$$

où

$$\mathcal{C}(k) = \{c' \in \mathbb{R}^n : c'_k \in [\underline{c}_k, \bar{c}_k] \ \& \ c'_j = c_j, \ j \in \mathcal{N} \setminus \{k\}\}.$$

Par exemple, si α est la propriété d'optimalité, l'intervalle de sensibilité d'un paramètre est donc l'intervalle de valeurs dans lequel la solution x préserve son optimalité. Dans le cas où l'intervalle de sensibilité est de longueur maximale, ses bornes sont appelées respectivement la tolérance inférieure et la tolérance supérieure.

Si à la fois un ensemble $\mathcal{A} \subseteq \mathcal{N}$ de paramètres de \mathcal{P} est susceptible de voir ses valeurs perturbées, on parle d'une **région de sensibilité** à l'intérieur de laquelle la solution préserve la propriété étudiée. Cette région dans \mathbb{R}^n peut être vue comme l'intersection du sous-espace $\Pi_{\mathcal{N} \setminus \mathcal{A}}(c)$ de codimension $|\mathcal{N} \setminus \mathcal{A}|$ (orthogonal aux axes de $\mathcal{N} \setminus \mathcal{A}$) traversant le point $c \in \mathbb{R}^n$ et la boule de rayon maximal (d'où la notion de rayon de stabilité) centrée sur le point $c \in \mathbb{R}^n$ telle que la solution x préserve la propriété α quels que soient les paramètres du problème à l'intérieur de cette région.

Le **rayon de stabilité**, noté $\rho(x, c)$, est donc défini de la manière suivante :

$$\rho(x, c) = \sup\{\varepsilon > 0 : \forall c' \in \Pi_{\mathcal{N} \setminus \mathcal{A}}(c) \cap \mathcal{B}(c, \varepsilon) \quad (x \in \alpha(\mathcal{P}'))\},$$

où

$$\Pi_{\mathcal{N} \setminus \mathcal{A}}(c) = \{c' \in \mathbb{R}^n : c'_j = c_j, \ j \in \mathcal{N} \setminus \mathcal{A}\},$$

$$\mathcal{B}(c, \varepsilon) = \{c' \in \mathbb{R}^n : \|c - c'\| < \varepsilon\},$$

$$\|c - c'\| = \max\{|c_j - c'_j| : j \in \mathcal{N}\}.$$

Ici, $\mathcal{B}(c, \varepsilon)$ est la boule ouverte centrée sur $c \in \mathbb{R}^n$ de rayon ε , $\|\cdot\|$ est la distance de Chebychev entre deux vecteurs dans \mathbb{R}^n .

Bien évidemment, plus la valeur du rayon de stabilité est grande, moins la solution est sensible face aux perturbations éventuelles.

Si pour chaque paramètre c_k , $k \in \mathcal{A}$, l'intervalle de sensibilité a été individuellement déterminé, alors le lien entre ces intervalles et le rayon de stabilité peut être exprimé par l'inégalité suivante

$$\rho(x, c) \leq \min_{k \in \mathcal{A}} \min\{|c_k - \underline{c}_k|, |\bar{c}_k - c_k|\}.$$

Historiquement, les premiers travaux sur l'analyse de sensibilité pour les problèmes d'optimisation combinatoire ont été publiés par Roodman [Roodman, 1972; 1974] et par Piper et Zoltners [Piper and Zoltners, 1975; 1976]. Ils ont considéré le problème en variables 0 – 1 suivant :

$$\min cx, \quad \text{s.c. } x \in X \subseteq \{0, 1\}^n. \quad (1.1)$$

Une méthode d'énumération implicite (arborescente) et de type séparation et évaluation ont été tout d'abord employées pour fournir un ensemble de solutions complètes (réalisables), qui contient la solution optimale, et un ensemble de solutions partielles dominées. L'arborescence a été ensuite utilisée pour déterminer l'intervalle de sensibilité d'un paramètre lors de l'étude de l'optimalité des solutions.

La notion de rayon de stabilité a été introduite dans les travaux de Leontev [Leont'ev, 1975; Leontev, 1979] comme le seuil maximal de perturbations des paramètres qui n'implique pas l'apparition de nouvelles solutions optimales pour des problèmes perturbés par rapport au problème d'origine. Pour le problème (1.1), Leontev a proposé la formule de calcul du rayon de stabilité suivante :

$$\rho(X^*, c) = \min_{x' \in X \setminus X^*} \max_{x \in X^*} \frac{c(x' - x)}{\|x' - x\|_1}, \quad (1.2)$$

où X^* est l'ensemble des solutions optimales, $\|x\|_1 = \sum_{i \in \mathcal{N}} |x_i|$.

Des résultats semblables ont été obtenus par Gordeev et al. [Gordeev and Leont'ev, 1980] pour le problème combinatoire du goulot d'étranglement. Puis dans [Gordeev *et al.*, 1983], une modification de l'algorithme de type séparation et évaluation a été proposée afin de calculer plus efficacement (1.2) pour le problème du voyageur de commerce. Dans [Gordeev, 1989], l'auteur a démontré que la recherche du rayon de stabilité pour le problème du plus court chemin est \mathcal{NP} -difficile.

Après ces travaux pionniers, la recherche de l'intervalle de sensibilité ou du rayon de stabilité a fait l'objet de nombreuses études dans la littérature. Ci-dessous, nous présentons les résultats obtenus les plus importants. Pour d'autres références, nous invitons le lecteur à consulter la bibliographie de Greenberg [Greenberg, 1998].

Problème du sac-à-dos. Belgacem, Hifi et al. ont étudié la sensibilité d'une solution optimale du problème du sac-à-dos [Hifi *et al.*, 2005; 2008; Belgacem and Hifi, 2008a; 2008d] et l'une de ses variantes appelée dans la littérature anglo-saxonne « sharing knapsack problem » [Belgacem and Hifi, 2008c; 2008b] face aux perturbations des profits et

des poids du problème. Les résultats obtenus ont permis aux auteurs de proposer des algorithmes de calcul des intervalles de sensibilité de ces paramètres et, comme cela a été fait dans [Hifi *et al.*, 2008], d'adapter un algorithme de type séparation et évaluation pour calculer ces intervalles lors de la recherche d'une solution optimale.

Problème du voyageur de commerce. Dans [van der Poort, 1997; Libura *et al.*, 1998], les auteurs ont étudié différents aspects de la sensibilité pour le problème du voyageur de commerce (PVC). En particulier, les auteurs ont considéré le problème de recherche des k meilleures solutions de PVC en connaissant la solution optimale et son rayon de stabilité. Un algorithme polynomial pour ce problème a été présenté pour le cas $k = 2$, et il a été prouvé que ce problème devient \mathcal{NP} -difficile pour le cas $k > 2$. Les auteurs ont aussi montré comment utiliser la connaissance des k meilleures solutions pour calculer efficacement le rayon de stabilité de celle qui est optimale.

La complexité du calcul des intervalles de sensibilité et des rayons de stabilité pour les solutions optimales de nombreux problèmes d'optimisation combinatoire a été étudiée dans [Chakravarti and Wagelmans, 1998; Van Hoesel and Wagelmans, 1999; Ramaswamy *et al.*, 2000]. Les auteurs ont montré que le calcul de ces mesures n'est pas plus facile que le problème d'optimisation lui-même. Ce qui implique, par exemple, que le problème de recherche des intervalles de sensibilité ou du rayon de stabilité pour la solution optimale de PVC est \mathcal{NP} -difficile.

Problème du plus court chemin. Dans [Ramaswamy *et al.*, 2005], les auteurs ont considéré le problème du plus court chemin dans un graphe non-orienté avec m arêtes. Ils ont démontré que l'intervalle de sensibilité de la longueur d'une arête peut être calculé en $\mathcal{O}(m + k \log k)$, où k est le nombre d'arêtes dans le chemin optimal étudié.

Problème de l'arbre couvrant de poids minimal. Le problème d'arbre couvrant de poids minimal dans un graphe avec n sommets et m arêtes a été étudié dans [Shier and Witzgall, 1980]. Les auteurs ont démontré que l'intervalle de sensibilité pour le poids d'une arête peut être calculé en $\mathcal{O}(n^2)$. Puis, dans [Tarjan, 1982], ce résultat a été amélioré et un algorithme de complexité $\mathcal{O}(m\alpha(m, n))$ a été développé, où $\alpha(m, n)$ est la fonction inverse d'Ackermann [Tarjan, 1975]. Enfin, après 23 ans, Pettie [Pettie, 2005] en se basant sur la structure de données « split-findmin » a proposé un algorithme de complexité record $\mathcal{O}(m \log \alpha(m, n))$ pour ce problème.

Problèmes d'ordonnancement. Sotskov *et al.* [Sotskov, 1991; Kravchenko *et al.*, 1995; Bräsel *et al.*, 1996; Sotskov *et al.*, 1997; 1998a; 1998b; 2010] ont étudié le rayon de stabilité d'une solution optimale en ordonnancement sous l'incertitude sur les temps d'exécution des tâches. Leurs travaux s'appliquent à une large gamme de problèmes d'ordonnancement essentiellement de type job shop et open shop. Les résultats généraux présentés sont des conditions nécessaires et suffisantes pour l'existence d'un rayon de sta-

bilité strictement positif, ou pour que le rayon de stabilité soit infini (approche qualitative de sensibilité). Ils ont également présenté des formules générales pour le calcul du rayon de stabilité, mais à complexité exponentielle.

L'analyse de sensibilité en ordonnancement a aussi été traitée dans l'article [Hall and Posner, 2004]. Hall et Posner ont cherché à calculer la déviation relative de performance d'une solution optimale. Ainsi, par exemple, pour le problème de minimisation du makespan à deux machines parallèles $P2||C_{\max}$ (ici $C_{\max} = \max_i C_i$), ils ont démontré que la déviation relative est bornée par $8/7$ si une perturbation sur le temps d'exécution d'une tâche ne dépasse pas un certain seuil. Ce seuil calculable en $O(n \log n)$ dépend de la structure de la solution optimale. Lorsqu'il s'agit d'un problème polynomial, comme, par exemple, le problème de minimisation du temps total d'exécution pondéré $1||\sum_i w_i C_i$ pour une machine, il est possible de calculer les intervalles de sensibilité pour chaque poids w_i à l'intérieur duquel l'optimalité de la solution étudiée n'est pas altérée. Quant aux problèmes pseudo-polynomiaux, comme, par exemple, le problème de minimisation du nombre pondéré de tâches en retard $1||\sum_i w_i U_i$, Hall et Posner ont montré qu'on peut calculer le rayon de stabilité lors de la recherche d'une solution optimale au prix de calculs supplémentaires (conservant pseudo-polynomialité de l'algorithme).

Problèmes multiobjectifs. L'analyse de sensibilité a aussi été menée pour des problèmes multiobjectifs de programmation linéaire et quadratique en nombres entiers dans [Emelichev *et al.*, 2002; Lebedeva *et al.*, 2005; Emelichev and Gurevskii, 2009]. Les auteurs ont analysé le comportement de l'ensemble des solutions optimales au sens de Pareto face aux perturbations éventuelles des données initiales. Les auteurs ont étudié, par exemple, le rayon de stabilité de variations des données qui n'impliquent pas l'apparition de nouvelles solutions optimales au sens de Pareto. Les problèmes avec un rayon de stabilité strictement positif défini de telle façon sont appelés stables. Les auteurs ont proposé des formules de calcul du rayon de stabilité et ont présenté des conditions nécessaires et suffisantes de stabilité du problème. Ainsi, ils ont prouvé que les problèmes sont stables si et seulement si chaque solution optimale au sens de Pareto est optimale au sens de Pareto faible (optimum de Slater [Ehrgott, 2005]). De plus, ils ont proposé un opérateur, nommé l'opérateur de régularisation, qui peut être utilisé pour transformer un problème donné en une série de problèmes stables avec le même ensemble de solutions optimales au sens de Pareto que le problème d'origine.

Planification de la production. Par ailleurs, le rayon de stabilité a aussi été utilisé en tant que mesure de sensibilité pour certains autres problèmes appliqués de gestion de production. Ainsi, Rossi dans sa thèse [Rossi, 2003] a considéré le problème de répartition de la charge au sein d'un parc de machines d'un atelier. Le but était de répartir la charge de travail entre machines de façon à maximiser leurs taux d'occupation. L'atelier

considéré recevait des pièces à partir d'autres ateliers en amont. Les perturbations qui survenaient dans les ateliers en amont pouvait alors modifier la commande prévisionnelle. Rossi a défini le rayon de stabilité comme la variation maximale de la commande ou l'amplitude maximale des perturbations pour lesquelles la solution initiale restait équilibrée, autrement dit la charge des machines n'était pas affectée. Par la suite, le problème d'optimisation a été formulé comme un problème bicritère : le premier critère étant l'équilibre dans la répartition des charges, et le deuxième visant à maximiser le rayon de stabilité.

L'analyse de la littérature nous permet de conclure que l'analyse de sensibilité prend une place importante dans la recherche opérationnelle, puisque les mesures comme l'intervalle de sensibilité et le rayon de stabilité qu'elle utilise permettent de prédire le comportement des solutions face aux perturbations des données initiales. Comme chaque problème d'optimisation combinatoire nécessite une étude de sensibilité particulière, nous considérons, dans les chapitres 3 et 4, deux problèmes d'optimisation combinatoire importants, à savoir les problèmes de conception de lignes d'assemblage et d'usinage, respectivement. Pour ces problèmes, nous étudions tout d'abord le calcul du rayon de stabilité au sens de faisabilité et d'optimalité des solutions. Puis nous montrons comment trouver un compromis entre le coût de la ligne à minimiser et son rayon de stabilité à maximiser.

1.4 Conclusion

Dans ce chapitre, nous avons étudié la modélisation de l'incertitude des données dans les problèmes d'optimisation. Nous avons présenté les différentes approches utilisées dans la littérature pour prendre en compte l'incertitude. Puis, nous nous sommes focalisés sur les approches a priori et plus précisément sur les approches déterministes comme les approches robustes et l'analyse de sensibilité, pour lesquelles une revue de la littérature a été effectuée. Ces approches seront utilisées pour traiter l'incertitude des données lors de la conception de lignes de fabrication dans les chapitres 3, 4 et 5.

Le contexte et la problématique de la conception de lignes de fabrication sont présentés dans le chapitre suivant.

Chapitre 2

Conception de lignes de fabrication

Sommaire

2.1	Lignes de fabrication	27
2.2	Problématique de l'équilibrage de lignes	29
2.3	Équilibrage sous incertitudes	33
2.4	Conclusion	36

2.1 Lignes de fabrication

Les lignes de fabrication au sens large ont pour but principal de transformer des produits bruts entrants en produits finis sortants. Une telle ligne peut être vue comme un ensemble de stations disposées de manière séquentielle de sorte qu'il soit possible d'insérer un produit brut au début de la ligne et de récupérer un produit fini à la fin de la ligne. Ainsi, une unité de produit parcourt la ligne dans une seule direction et visite séquentiellement toutes les stations installées.

Lorsqu'une unité de produit arrive à une station, elle subit une suite d'opérations, effectuées par des ouvriers ou par des équipements automatiques associés à cette station. La durée totale des opérations sur une station représente la charge de la station. Deux types de lignes peuvent être distinguées par rapport à l'organisation du transfert de produits : lignes synchronisées et non synchronisées. Si la ligne est synchronisée, alors tous les produits se trouvant sur la ligne sont transportés simultanément à l'aide d'un dispositif de transfert commun. Le fonctionnement synchronisé de la ligne se base sur le temps de cycle (le temps qui sépare la sortie de deux produits finis). Dans ce cas, il est impératif que la charge de chaque station ne dépasse pas ce temps de cycle. Si cette condition est respectée, alors il est possible, au bout d'un temps de cycle, de faire avancer tous les produits se trouvant sur la ligne vers les stations suivantes, de charger

un nouveau produit brut sur la première station et de décharger un produit fini sur la dernière station. Dans les lignes non synchronisées, il n'y a pas cette contrainte grâce aux stocks tampons qui sont utilisés entre les stations.

L'architecture d'une ligne de fabrication influe directement sur sa fiabilité et son coût. Par conséquent, le choix de l'architecture d'une ligne parmi celles existantes est une décision importante. Voici une liste non exhaustive des architectures possibles :

- **Architecture linéaire homogène.** Les stations sont installées en série (Figure 2.1). Ce type de ligne se caractérise par un coût d'installation et d'exploitation relativement bas, mais il présente des inconvénients au niveau de la fiabilité, car si une station tombe en panne, toute la ligne est bloquée.

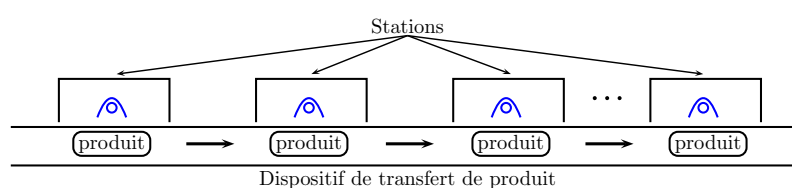


FIGURE 2.1 – Architecture linéaire homogène.

- **Lignes bilatérales.** Si le processus industriel permet l'exécution de certaines opérations sur deux faces opposées du produit, il est possible d'installer des stations des deux côtés de la ligne (Figure 2.2).

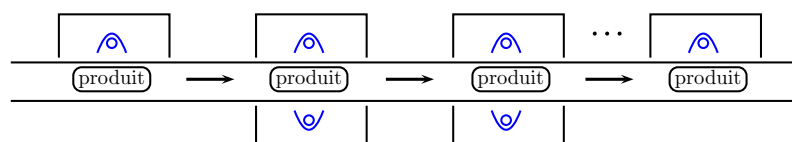


FIGURE 2.2 – Ligne de fabrication bilatérale.

- **Lignes à stations parallèles.** Dans le cas où il y aurait des opérations qui ont un temps d'exécution supérieur au temps de cycle, la mise en place de stations identiques parallèles sur le même poste de travail (Figure 2.3) permet de respecter le temps de cycle et d'améliorer la fiabilité de la ligne.

Quelque soit le type de la ligne de fabrication, sa conception préliminaire en phase d'avant-projet demande un ensemble d'approches qui nécessitent la résolution de plusieurs problèmes complexes [Askin and Standridge, 1993; Dolgui and Proth, 2006]. Les étapes générales suivantes résument ce processus :

- **Analyse du produit.** Cette phase correspond à la définition du produit à fabriquer. Elle permet d'avoir une description complète des opérations de fabrication nécessaires.

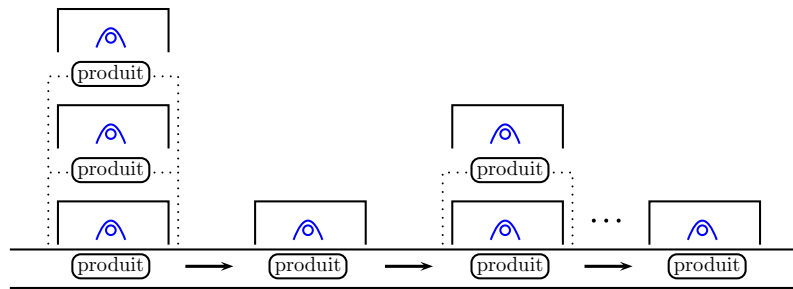


FIGURE 2.3 – Ligne de fabrication à stations parallèles.

- **Planification de processus.** Cette étape s'intéresse à la sélection de l'architecture de la ligne. En outre, c'est lors de cette phase que les contraintes technologiques sont explicitées. Par exemple, un ordre partiel entre les opérations ainsi que des contraintes d'inclusion et d'exclusion entre les opérations. Ceci nécessite une connaissance approfondie des spécifications du produit et des contraintes technologiques liées aux procédés de fabrication.
- **Configuration.** À ce stade, on étudie l'affectation des opérations aux stations de façon à ajuster la productivité de la ligne à la demande.
- **Analyse de flux.** La simulation à événements discrets peut être utilisée afin d'étudier le flux des produits en tenant compte à la fois des événements aléatoires externes et de la variabilité de la production.

La prise en compte simultanée de l'ensemble de ces étapes est en pratique une tâche très difficile. C'est pourquoi la démarche la plus souvent utilisée procède de façon hiérarchique en considérant les étapes les unes après les autres. Cependant, il est souvent nécessaire de revenir aux étapes antérieures lorsque les décisions qui y sont prises s'avèrent inadéquates.

Dans cette thèse, nous nous intéressons plus particulièrement à une étape de conception très importante, nommée ici « Configuration », car le coût et l'efficacité du système de fabrication conçu dépendent fortement de la qualité des résultats obtenus durant ce stade. Souvent, ce stade est associé au problème d'équilibrage de ligne que nous présentons dans la section suivante.

2.2 Problématique de l'équilibrage de lignes

D'une manière générale le problème d'équilibrage consiste à affecter les opérations aux stations tout en respectant les contraintes du problème, afin d'optimiser un ou plusieurs critères (coût de la ligne, nombre de stations, temps de cycle, etc.).

Historiquement, le premier problème d'équilibrage consistait à minimiser les pertes,

engendrées par les temps morts des opérateurs travaillant sur des lignes d'assemblage fabriquant un seul type de produit. Les hypothèses suivantes sont à la base de cette première formulation :

- L'ensemble d'opérations $V = \{1, 2, \dots, n\}$ est connu.
- Chaque opération $j \in V$ est caractérisée par son temps d'exécution déterministe t_j .
- Les stations sont identiques en termes de configuration et n'importe quelle opération peut être affectée à n'importe quelle station.
- Les opérations affectées à une station sont exécutées de manière séquentielle. La somme des temps d'exécution de toutes ces opérations représente la charge de cette station.
- Chaque opération doit être affectée à une seule station installée sur la ligne.
- Un seul type de produit est assemblé.
- La ligne est synchronisée, c'est-à-dire l'exécution d'opérations sur toutes les stations de la ligne est limitée par un temps de cycle objectif T_0 .
- Un ordre partiel d'exécution d'opérations nommé « contraintes de précédence » doit être respecté lors de la répartition des opérations. Cet ordre peut être vu comme un ensemble \mathcal{A} de paires (i, j) modélisant le fait que l'opération j ne peut pas être exécutée avant l'opération i .
- L'objectif est de minimiser le nombre de postes de travail nécessaires pour affecter toutes les opérations en respectant l'ensemble des contraintes de précédence et la contrainte de temps de cycle.

Salveson [Salveson, 1955] a été le premier à étudier ce type de problèmes. Il est facile de voir qu'il s'agit d'une généralisation du problème de bin packing (avec les contraintes de précédence), qui est connu pour être \mathcal{NP} -difficile [Garey and Johnson, 1979]. Plus tard, cette généralisation a été surnommée Simple Assembly Line Balancing Problem (SALBP) par Baybars [Baybars, 1986].

En utilisant les notations introduites ci-dessus, le modèle mathématique de SALBP peut être présenté comme suit :

$$\text{Minimiser } \sum_{k=1}^m y_k \tag{2.1}$$

sous les contraintes

$$\sum_{k=1}^m x_{jk} = 1, \quad \forall j \in V, \tag{2.2}$$

$$\sum_{k=1}^m kx_{ik} \leq \sum_{k=1}^m kx_{jk}, \quad \forall (i, j) \in \mathcal{A}, \tag{2.3}$$

$$\begin{aligned} \sum_{j \in V} t_j x_{jk} &\leq T_0 y_k, \quad \forall k \in \{1, 2, \dots, m\}, \\ y_k &\in \{0, 1\}, \quad \forall k \in \{1, 2, \dots, m\}, \\ x_{jk} &\in \{0, 1\}, \quad \forall j \in V, \quad \forall k \in \{1, 2, \dots, m\}, \end{aligned} \quad (2.4)$$

où m est le nombre maximal autorisé de stations sur la ligne, la variable y_k est égale à 1 si la station k est ouverte, 0 sinon, la variable binaire x_{jk} est égale à 1 si l'opération j est affectée à la station k , 0 sinon, (2.1) représente la fonction objectif, l'égalité (2.2) impose que chaque opération soit affectée à une seule station, les contraintes de précédence sont modélisées par les inégalités (2.3) et les inégalités (2.4) vérifient la contrainte de temps de cycle.

Hormis la formulation de Salveson qui a par la suite été désignée comme **SALBP-1**, trois autres problèmes d'équilibrage de lignes d'assemblage ont été distingués et placés dans la classe de SALBP (chacun comporte les hypothèses présentées ci-dessus sauf la fonction objectif qui est différente) :

- **SALBP-2**. Il s'agit de minimiser le temps de cycle, en affectant les opérations à une séquence de stations, dont le nombre est fixé.
- **SALBP-E** conjugue les objectifs de SALBP-1 et SALBP-2 et vise à maximiser l'efficacité de la ligne par la minimisation du produit : (nombre de stations) \times (temps de cycle).
- **SALBP-F**, quant à lui, est un problème de décision qui cherche à répondre à la question suivante : existe-t-il une manière de répartir l'ensemble d'opérations V sur m stations en satisfaisant les contraintes de précédence et en garantissant que la charge de toute station ne dépasse pas le temps de cycle T_0 ?

Le développement de l'industrie et des systèmes d'assemblage entraîna la popularité de ces problèmes également dans le milieu académique et, par conséquent, un certain nombre de modèles mathématiques, de méthodes et d'outils de résolution ont été développés pour les résoudre. Depuis les années 1960, de nombreuses études bibliographiques de ces problèmes ont été présentées, par exemple, [Ignall, 1965; Baybars, 1986; Talbot *et al.*, 1986; Ghosh and Gagnon, 1989; Erel and Sarin, 1998; Scholl, 1999; Rekiek *et al.*, 2002; Scholl and Becker, 2006; Guschinskaya and Dolgui, 2010].

Notons tout de même que dans la littérature l'attention a été portée plus aux problèmes SALBP-1 et SALBP-2 qu'à SALBP-E. Une des rares méthodes de résolution exactes pour SALBP-E a été proposée dans [Rosenblatt and Carlson, 1985]. Tout d'abord, cette méthode cherche la solution optimale pour le problème SALBP-1 avec le temps de cycle qui est égal au temps d'exécution de l'opération la plus longue (t_{\max}). Par conséquent, le nombre m de stations, obtenu de cette façon, peut être évidemment vu comme

une borne supérieure du nombre de stations pour le problème initial. Ainsi, le premier produit $m \cdot t_{\max}$ est déterminé. Puis, $m - 2$ problèmes SALBP-2 avec le nombre de stations correspondant aux éléments de l'ensemble $\{2, \dots, m - 1\}$ doivent être résolus de manière exacte. La résolution de chacun de ces problèmes fournit un produit respectif. Le produit minimal parmi les $m - 1$ résultats ainsi obtenus correspond à la valeur de la solution optimale.

La formulation de SALBP n'exprime pas toujours toute la réalité industrielle à cause des hypothèses simplificatrices sur lesquelles elle se base. Pour faire face à cette situation, plusieurs chercheurs ont essayé d'introduire des hypothèses plus générales dans le SALBP. N'importe quel problème de type ALBP ayant au moins une hypothèse différente par rapport à la formulation de base (SALBP) a été nommé dans la littérature Generalized Assembly Line Balancing Problem (GALBP) [Ghosh and Gagnon, 1989; Rekiek *et al.*, 2002; Becker and Scholl, 2006]. Parmi les hypothèses additionnelles, les suivantes sont les plus souvent prises en compte :

- **Temps inter-opérateur** (temps de setup). Un temps additionnel entre deux opérations successives d'une séquence peut être pris en compte dû au déplacement ou au changement d'outil exécutant les opérations [Andrés *et al.*, 2008; Scholl *et al.*, 2008; Essafi *et al.*, 2010b].
- **Opérations parallèles**. Des opérations différentes peuvent être exécutées simultanément sur la même station [Dolgui *et al.*, 2006a; Guschinskaya *et al.*, 2008; Guschinskaya and Dolgui, 2009; Delorme *et al.*, 2012].
- **Contraintes d'exclusion**. Elles viennent de l'impossibilité d'exécuter certaines opérations sur la même station [Buxey, 1974; Raouf and Tsui, 1982; Dolgui *et al.*, 2006c; 2008; Guschinskaya *et al.*, 2008].
- **Contraintes d'inclusion**. Elles peuvent traduire la nécessité d'exécuter certaines opérations dans la même station afin de respecter la tolérance de fabrication requise [Buxey, 1974; Deckro, 1989; Dolgui *et al.*, 2006a; Guschinskaya *et al.*, 2008; Guschinskaya and Dolgui, 2009].
- **Fonction objectif**. Il existe une classe de fonctions objectifs qui visent à optimiser à la fois les coûts liés à l'installation et à l'utilisation de la ligne. Ces coûts peuvent être engendrés par l'achat des équipements et l'installation des stations ou par les salaires des ouvriers assurant le fonctionnement de la ligne. La classe de ce type de problèmes d'équilibrage est connue dans la littérature sous le nom Cost-oriented Assembly Line Balancing Problem (CALBP) [Amen, 2000a; 2000b; 2001; 2006; Belmokhtar *et al.*, 2006; Dolgui *et al.*, 2006b; 2008; Essafi *et al.*, 2010a; 2010b; Guschinskaya and Dolgui, 2009].

Concernant les méthodes de résolution, il y en a beaucoup dans la littérature. D'une

manière générale, ces méthodes peuvent être divisées en deux classes, à savoir : les méthodes exactes et les méthodes approchées. Parmi les méthodes exactes, nous pouvons citer :

- Différentes *procédures par séparation et évaluation* comme FABLE [Johnson, 1988], EUREKA [Hoffmann, 1992], SALOME [Scholl and Klein, 1997] et d'autres plus récentes comme celles publiées dans [Liu *et al.*, 2008; Miralles *et al.*, 2008; Wu *et al.*, 2008; Dolgui and Ihnatsenka, 2009] ;
- Des méthodes de *programmation dynamique* [Jackson, 1956; Held *et al.*, 1963; Bard, 1989; Easton, 1990].

Malheureusement, la qualité des solutions qu'elles fournissent est souvent payée par un temps de résolution très important.

Les méthodes approchées, quant à elles, ne garantissent pas que la solution trouvée soit optimale, mais de manière générale, elles donnent de bons résultats, souvent plus ou moins proches de l'optimum. Elles sont utilisées pour des problèmes de grande taille ou lorsque le temps de calcul alloué est très limité. Différentes *heuristiques* ont été proposées, les plus connues sont celles de [Helgeson and Birnie, 1961; Arcus, 1965; Talbot *et al.*, 1986; Boctor, 1995]. Parmi les méthodes métaheuristiques les plus utilisées il y a :

- la *recherche Tabou* [Chiang, 1998; Pastor *et al.*, 2002; Lapierre *et al.*, 2006; Özcan and Toklu, 2009],
- le *recuit simulé* [McMullen and Frazier, 1998; Vilarinho and Simaria, 2002; Baykasoglu, 2006; Gourgand *et al.*, 2007; Özcan, 2010],
- les *colonies de fourmis* [Bautista and Pereira, 2002; McMullen and Tarasewich, 2003; Vilarinho and Simaria, 2006; McMullen and Tarasewich, 2006; Bautista and Pereira, 2007; Baykasoglu and Dereli, 2008; Simaria and Vilarinho, 2009; Sabuncuoglu *et al.*, 2009; Essafi *et al.*, 2010a],
- les *algorithmes génétiques* [Rubinovitz and Levitin, 1995; Suresh *et al.*, 1996; Sabuncuoglu *et al.*, 2000; Rekiek *et al.*, 2001; Simaria and Vilarinho, 2004; Rekiek and Delchambre, 2006; Levitin *et al.*, 2006; Tasan and Tunali, 2008; Kim *et al.*, 2009; Guschinskaya *et al.*, 2011].

2.3 Équilibrage sous incertitudes

En pratique, vue la complexité des systèmes de fabrication, il n'est pas toujours possible, voire quasi impossible, de connaître les données d'un problème d'une manière précise et certaine. Les concepteurs sont souvent confrontés à une situation où ils déterminent la configuration d'une ligne pour des données estimées et lorsqu'ils commencent sa mise en œuvre, ils se rendent compte que la configuration ainsi obtenue n'est pas correcte pour

les données réelles, ce qui peut entraîner une forte dégradation de la performance prévue.

La détermination d'une configuration avec de bonnes performances nécessite la connaissance préalable et précise des temps d'exécution des opérations. Or, les temps d'exécution des opérations ne sont que des estimations. Souvent, lors de la mise en œuvre d'une configuration, certaines opérations ont des temps d'exécution plus petits ou plus grands que les valeurs estimées pour lesquelles la configuration a été calculée.

Ces incertitudes sur les temps d'exécution des opérations sont principalement dues [Chiang and Urban, 2006] à l'instabilité des performances des opérateurs humains, au changement des caractéristiques du produit ou des stations, au changement de la composition matérielle du produit et aux micro-arrêts des stations. Tous ces facteurs sont difficiles à modéliser et peuvent intervenir à n'importe quel moment de la vie de la ligne. C'est pourquoi, tenir compte des incertitudes lors de la conception en avant-projet des lignes de fabrication est une tâche importante.

Pour faire face à l'incertitude des temps d'exécution des opérations lors de la conception en avant-projet des lignes de fabrication, certaines approches ont été proposées dans la littérature, les plus utilisées sont les approches probabilistes [Urban and Chiang, 2006; Chiang and Urban, 2006; Baykasoğlu and Özbakır, 2007; Ağpak and Gökçen, 2007; Özcan, 2010] et les approches par ensembles flous [Tsujiimura *et al.*, 1995; Gen *et al.*, 1996; Hop, 2006].

Dans les approches probabilistes, le plus souvent, les auteurs considèrent le cas où les temps d'exécution des opérations seraient des variables aléatoires de loi normale avec une espérance et une variance connues, c'est-à-dire $t_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$, $j \in V$. Ainsi, par exemple, pour le problème SALBP-1, la contrainte (2.4) sur le temps de cycle est prise en compte de la manière suivante :

$$\mathbb{P} \left(\sum_{j \in V} t_j x_{jk} \leq T_0 \right) \geq \alpha, \quad \forall k \in \{1, 2, \dots, m\}. \quad (2.4')$$

C'est-à-dire la probabilité que « la charge d'une station ne dépasse pas T_0 » doit être supérieure ou égale à un seuil de confiance α (en pratique, α est supérieur à 0.9).

Utilisant les propriétés de la variable aléatoire de loi normale, on peut déduire que $\mathcal{X}_k := \sum_{j \in V} t_j x_{jk}$ est aussi une variable aléatoire de loi normale telle que

$$\mathcal{X}_k \sim \mathcal{N} \left(\sum_{j \in V} \mu_j x_{jk}, \sum_{j \in V} \sigma_j^2 x_{jk} \right).$$

Après la transformation connue $\mathcal{Y}_k := \frac{\mathcal{X}_k - \sum_{j \in V} \mu_j x_{jk}}{\sqrt{\sum_{j \in V} \sigma_j^2 x_{jk}}}$, $\mathcal{Y}_k \sim \mathcal{N}(0, 1)$, l'inégalité (2.4') peut être écrite de la façon suivante :

$$\mathbb{P} \left(\mathcal{Y}_k \leq \frac{T_0 - \sum_{j \in V} \mu_j x_{jk}}{\sqrt{\sum_{j \in V} \sigma_j^2 x_{jk}}} \right) \geq \alpha, \quad \forall k \in \{1, 2, \dots, m\}.$$

D'où

$$\sum_{j \in V} \mu_j x_{jk} + \Phi^{-1}(\alpha) \sqrt{\sum_{j \in V} \sigma_j^2 x_{jk}} \leq T_0, \quad \forall k \in \{1, 2, \dots, m\}, \quad (2.4'')$$

où $\Phi(x) = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$ est la fonction de répartition de la loi normale centrée réduite.

Avec la contrainte de temps de cycle (2.4'') substituée à (2.4), le problème SALBP-1 reste déterministe et la plupart des méthodes développées auparavant sont applicables.

Quant aux approches par ensembles flous, les temps d'exécution des opérations sont modélisés par des ensembles avec des degrés d'appartenance associés (souvent triangulaires ou trapézoïdaux). Cependant, lors de l'affectation des opérations aux stations des méthodes de comparaison entre les temps d'exécution flous doivent être développées.

Toutefois, comme il était indiqué dans la section 1.4, l'application des approches probabilistes ou celles des ensembles flous en pratique peut s'avérer être une tâche difficile à cause de la connaissance insuffisante a priori sur les temps d'exécution des opérations pour déduire les lois de probabilité ou les degrés d'appartenance associés. Plus souvent, les concepteurs disposent d'une information moins riche, par exemple, un ensemble d'opérations dont le temps d'exécution est susceptible de varier, ou les temps d'exécution des opérations connus sous la forme d'intervalles sans loi de probabilité attachée.

Pour le cas où l'ensemble des opérations avec un temps variable serait connu, [Sotskov *et al.*, 2006] proposent d'étudier l'analyse de sensibilité du problème SALBP-1. Dans cet article, ils mettent l'accent sur l'étude des solutions optimales en proposant les conditions nécessaires et suffisantes de stabilité et la formule de calcul du rayon de stabilité. Cependant, ils ne montrent pas comment utiliser cette information lors de la conception de ce type de lignes de fabrication. Dans cette thèse, nous essayons de combler cette lacune. Ainsi, nous menons l'analyse de sensibilité pour d'autres problèmes d'équilibrage plus complexes comme SALBP-E (Chapitre 3) ou l'équilibrage d'une ligne d'usinage à boîtiers multibroches (Chapitre 4). En outre, nous élargissons l'approche de [Sotskov *et al.*, 2006] non seulement en étudiant les solutions optimales, mais aussi des solutions faisables et quasi-faisables (solutions qui respectent toutes les contraintes sauf les contraintes de temps de cycle). De plus, nous montrons que le rayon de stabilité est une mesure de performance appropriée et proposons des approches qui prennent en compte sa valeur lors de la configuration en avant-projet de ces lignes. Quant au cas où les temps sont donnés par des intervalles, à notre connaissance, il n'existe pas de travaux scientifiques publiés. C'est pourquoi, dans le Chapitre 5, nous tentons d'adapter l'approche de Bertsimas et Sim [Bertsimas and Sim, 2003] à l'équilibrage de lignes d'assemblage de type SALBP-1 et proposons une procédure par séparation et évaluation de recherche de solutions robustes.

2.4 Conclusion

Dans ce chapitre, nous avons présenté les lignes de fabrication et leur méthodologie de conception de façon générale. Nous avons introduit la problématique qui fait l'objet de notre étude et qui concerne leur conception en avant-projet. Un état de l'art non-exhaustif des principales méthodes de résolution des problèmes d'équilibrage des lignes de fabrication a été donné. Puis, nous nous sommes interrogés sur la problématique importante de prise en compte des incertitudes des temps d'exécution des opérations lors de la phase de conception en avant-projet de ces systèmes, qui est le sujet de cette thèse.

Chapitre 3

Équilibrage de lignes d'assemblage de type SALBP-E : analyse de sensibilité

Sommaire

3.1	Introduction	37
3.2	Définitions et propriétés de base	38
3.3	Exemple	43
3.4	Solutions faisables	44
3.5	Solutions quasi-faisables	46
3.6	Solutions optimales	47
3.7	Recherche du compromis	54
3.8	Résultats numériques	57
3.9	Conclusion	60

3.1 Introduction

Dans ce chapitre, nous considérons le problème d'équilibrage des lignes d'assemblage de type SALBP-E, qui consiste à affecter l'ensemble des opérations de fabrication aux stations tout en respectant les contraintes de précédence et les contraintes de temps de cycle, afin de minimiser l'expression suivante : (le nombre de stations) \times (le temps de travail sur la station la plus chargée).

Une attention particulière est portée ici sur la conception préliminaire de ce type de lignes d'assemblage en présence d'un ensemble d'opérations dont le temps d'exécution est susceptible de varier. Tout d'abord, comme dans l'article [Sotskov *et al.*, 2006], nous étudions des aspects théoriques de l'analyse de sensibilité des solutions pour ce problème

face aux variations des temps d'exécution des opérations. Nous proposons des conditions nécessaires et suffisantes de stabilité des solutions et les formules de calcul de leurs rayons de stabilité, respectivement. Nous montrons que le rayon de stabilité est une mesure de performance appropriée qui peut jouer un rôle important lors de la conception de ce type de lignes. Quelques exemples d'une telle utilisation sont également présentés.

Le reste du chapitre est organisé de la manière suivante. Dans la Section 3.2, les définitions et propriétés de base de l'analyse de sensibilité, ainsi que le modèle mathématique du SALBP-E, sont introduits. Un exemple illustratif est présenté dans la Section 3.3. Les sections 3.4, 3.5 et 3.6 s'intéressent à l'étude du rayon de stabilité respectivement pour des solutions faisables, quasi-faisables, et optimales. Deux procédures heuristiques qui cherchent un compromis entre les valeurs de la fonction objectif et du rayon de stabilité d'une solution faisable sont décrites dans la Section 3.7. Les résultats expérimentaux de l'utilisation de ces procédures sont présentés dans la Section 3.8. La conclusion est donnée dans la Section 3.9.

3.2 Définitions et propriétés de base

Nous considérons une ligne d'assemblage qui doit être conçue pour un produit donné. Les opérations nécessaires pour fabriquer le produit constituent un ensemble $V = \{1, 2, \dots, n\}$ associé au vecteur $t = (t_1, t_2, \dots, t_n) \in \mathbb{R}_+^n$ de temps d'exécution, où t_j est le temps d'exécution de l'opération $j \in V$ et \mathbb{R}_+ est l'ensemble des nombres réels positifs.

Les restrictions suivantes doivent être prises en compte lors de l'affectation des opérations aux stations :

- Des *contraintes de capacité* s'expriment par deux paramètres $c \in \mathbb{R}_+$ et $m \in \mathbb{N}$ qui introduisent le temps de travail maximal autorisé sur chaque station et le nombre maximal de stations dans la ligne, respectivement.
- Des *contraintes de précedence* introduisent une relation d'ordre partiel non-strict sur les opérations, et sont représentées par un graphe orienté acyclique $G = (V, \mathcal{A})$. Un arc (i, j) appartient à \mathcal{A} si et seulement si l'opération j doit être affectée à une station qui ne doit pas précéder la station à laquelle l'opération i est affectée. Néanmoins, cette relation autorise l'affectation des opérations i et j à la même station.

En utilisant les notations présentées ci-dessus, un modèle mathématique de SALBP-E peut être exprimé comme suit :

$$\text{Minimiser } \mathcal{T} \cdot \left(\sum_{k=1}^m y_k \right) \tag{3.1}$$

sous les contraintes

$$\sum_{k=1}^m x_{jk} = 1, \quad \forall j \in V, \quad (3.2)$$

$$\sum_{k=1}^m kx_{ik} \leq \sum_{k=1}^m kx_{jk}, \quad \forall (i, j) \in \mathcal{A}, \quad (3.3)$$

$$\sum_{j \in V} t_j x_{jk} \leq \mathcal{T} y_k, \quad \forall k \in \{1, 2, \dots, m\}, \quad (3.4)$$

$$y_k \in \{0, 1\}, \quad \forall k \in \{1, 2, \dots, m\},$$

$$x_{jk} \in \{0, 1\}, \quad \forall j \in V, \quad \forall k \in \{1, 2, \dots, m\},$$

où $\mathcal{T} \in [0, c]$ est une variable réelle estimant le temps de travail sur la station la plus chargée. La variable y_k est égale à 1 si la station k est ouverte, 0 sinon. La variable binaire x_{jk} est égale à 1 si l'opération j est affectée à la station k , 0 sinon. L'expression (3.1) représente la fonction objectif. L'égalité (3.2) impose que chaque opération soit affectée à une seule station. Les contraintes de précédence sont modélisées par les inégalités (3.3), et les inégalités (3.4) vérifient les contraintes de capacité.

Avant de considérer l'analyse de sensibilité de ce problème, quelques notions importantes doivent être introduites.

Définition 3.1. *Une affectation des opérations V aux stations est dite une solution faisable si les contraintes de précédence et les contraintes de capacité sont respectées.*

Dans la suite, l'ensemble des solutions faisables pour un vecteur donné $t \in \mathbb{R}_+^n$ est noté $\mathbf{S}_{\mathcal{F}}(t)$, où chaque solution s est caractérisée par la collection suivante $\{V_1^s, V_2^s, \dots, V_m^s\}$ de sous-ensembles disjoints non-vides de V tels que $V = V_1^s \cup V_2^s \cup \dots \cup V_m^s$. Ici V_k^s est l'ensemble des opérations affectées à la station $k \in W^s$ de la solution s , $W^s = \{1, 2, \dots, m^s\}$ est l'ensemble des numéros des stations de la solution s .

Ainsi, le modèle générique de SALBP-E peut être formulé comme suit :

$$\mathcal{Z}(s, t) := m^s \cdot c(s, t) \rightarrow \min_{s \in \mathbf{S}_{\mathcal{F}}(t)},$$

où $c(s, t) = \max\{t(V_k^s) : k \in W^s\}$ est le temps de travail sur la station la plus chargée de la solution s pour le vecteur t , $t(V_k^s) = \sum_{j \in V_k^s} t_j$ est le temps de travail de la station k . Les solutions faisables ayant la valeur minimale de la fonction objectif $\mathcal{Z}(s, t)$ sont appelées les solutions optimales, qui constituent l'ensemble $\mathbf{S}_{\mathcal{O}}(t)$. Il est évident que $\mathbf{S}_{\mathcal{O}}(t) \subseteq \mathbf{S}_{\mathcal{F}}(t)$.

Les deux propriétés suivantes sont des conclusions immédiates des définitions introduites ci-dessus.

Propriété 3.1. *Pour toute paire de solutions $s, s^0 \in \mathbf{S}_O(t) : \mathcal{Z}(s, t) = \mathcal{Z}(s^0, t)$.*

Propriété 3.2. *Pour tout $s \in \mathbf{S}_O(t)$ et tout $s^0 \in \mathbf{S}_F(t) \setminus \mathbf{S}_O(t) : \mathcal{Z}(s, t) < \mathcal{Z}(s^0, t)$.*

Dans cette étude, nous supposons que l'ensemble V contient deux types d'opérations [Sotskov *et al.*, 2006] :

- Opérations incertaines : leur temps d'exécution peut varier pendant l'exploitation de la ligne. L'ensemble de ces opérations est noté \tilde{V} .
- Opérations déterministes : leur temps d'exécution reste toujours le même et est fixé. Ces opérations constituent l'ensemble $V \setminus \tilde{V}$.

Sans perte de généralité, nous supposons que $\tilde{V} = \{1, 2, \dots, \tilde{n}\}$ et $V \setminus \tilde{V} = \{\tilde{n} + 1, \tilde{n} + 2, \dots, n\}$, où $0 < \tilde{n} \leq n$.

Les variations des temps d'exécution des opérations (VTEO) de \tilde{V} peuvent être représentées par le vecteur $\xi = (\xi_1, \xi_2, \dots, \xi_{\tilde{n}}, 0, 0, \dots, 0) \in \mathbb{R}^n$, où tout $\xi_j, j \in \tilde{V}$, peut être aussi bien positif que négatif. Par conséquent, à un moment donné de la vie de la ligne le vecteur des temps d'exécution peut être représenté par le vecteur *perturbé* suivant $t^* = (t_1 + \xi_1, t_2 + \xi_2, \dots, t_{\tilde{n}} + \xi_{\tilde{n}}, t_{\tilde{n}+1}, \dots, t_n)$.

Remarque 3.1. *Dans cette étude, il est supposé que $t_j^* = \max\{0, t_j + \xi_j\}, j \in \tilde{V}$.*

Notons que les VTEO ne modifient ni les contraintes de précédence ni le nombre de stations de la solution étudiée (optimale ou faisable). Cependant, elles peuvent provoquer la perte de l'optimalité et même de la faisabilité d'une solution. La faisabilité d'une solution peut être perdue si le temps de travail sur la station la plus chargée devient supérieur à c pour un vecteur perturbé t^* . Une solution optimale s trouvée pour le vecteur initial t peut perdre son optimalité, en restant faisable, pour un nouveau vecteur perturbé t^* s'il existe une solution $s^0 \in \mathbf{S}_F(t^*)$ telle que $\mathcal{Z}(s^0, t^*) < \mathcal{Z}(s, t^*)$.

Notons aussi que la solution s^0 respectant les contraintes de précédence pour laquelle $m^{s^0} \leq m$ et $c(s^0, t) > c$, peut devenir faisable et même optimale si $c(s^0, t^*) \leq c$ est vérifié pour un nouveau vecteur perturbé t^* . L'ensemble de ce type de solutions (appelées *quasi-faisables*) est noté $\mathbf{S}_{\tilde{F}}(t)$.

Pour modéliser les VTEO possibles, la distance de Chebychev entre deux vecteurs t et t' dans \mathbb{R}_+^n est définie de la manière suivante :

$$\|t - t'\| = \max\{|t_i - t'_i| : i \in V\}.$$

Cela permet d'introduire la notion de ε -voisinage de t dans \mathbb{R}_+^n , noté $\Omega(\varepsilon, t)$, avec :

$$\Omega(\varepsilon, t) = \{t' \in \Psi(t) : \|t - t'\| < \varepsilon\}, \quad \varepsilon > 0,$$

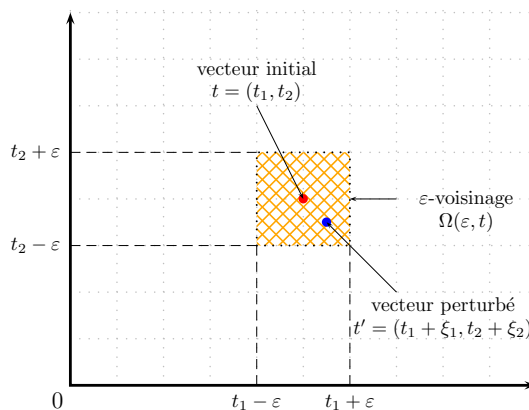


FIGURE 3.1 – Modélisation des perturbations.

où

$$\Psi(t) = \{t' \in \mathbb{R}_+^n : t'_j = t_j, j \in V \setminus \tilde{V}\}.$$

En introduisant $\mathcal{R} \in \{\mathcal{F}, \widehat{\mathcal{F}}, \mathcal{O}\}$, les définitions suivantes sont au centre de cette étude, où \mathcal{F} désigne la faisabilité, $\widehat{\mathcal{F}}$ désigne la quasi-faisabilité et \mathcal{O} désigne l'optimalité.

Définition 3.2. Une solution $s \in \mathbf{S}_{\mathcal{R}}(t)$ est dite \mathcal{R} -stable s'il existe un ε -voisinage $\Omega(\varepsilon, t)$ tel que pour tout $t' \in \Omega(\varepsilon, t)$, s reste dans $\mathbf{S}_{\mathcal{R}}(t')$, c'est-à-dire s est \mathcal{R} -stable si la condition suivante est vérifiée :

$$\Xi_{\mathcal{R}}(s, t) := \{\varepsilon > 0 : \forall t' \in \Omega(\varepsilon, t) \ (s \in \mathbf{S}_{\mathcal{R}}(t'))\} \neq \emptyset.$$

Définition 3.3. Le \mathcal{R} -rayon de stabilité $\rho_{\mathcal{R}}(s, t)$ de la solution $s \in \mathbf{S}_{\mathcal{R}}(t)$ est défini comme la plus petite borne supérieure de $\Xi_{\mathcal{R}}(s, t)$, c'est-à-dire

$$\rho_{\mathcal{R}}(s, t) = \sup \Xi_{\mathcal{R}}(s, t).$$

Remarque 3.2. Dans cette étude, il est supposé que $\sup \emptyset = 0$.

Un exemple à deux dimensions des notions introduites est donné dans la Figure 3.1.

Le \mathcal{R} -rayon de stabilité d'une solution $s \in \mathbf{S}_{\mathcal{R}}(t)$ peut être considéré comme le rayon maximal d'une boule ouverte centrée sur t dans $(\mathbb{R}_+^n, \|\cdot\|)$ telle que s reste dans $\mathbf{S}_{\mathcal{R}}(t')$ quel que soit le vecteur perturbé t' à l'intérieur de cette boule.

Il est possible de voir que s est \mathcal{R} -stable (resp. n'est pas \mathcal{R} -stable) si et seulement si $\rho_{\mathcal{R}}(s, t) > 0$ (resp. $\rho_{\mathcal{R}}(s, t) = 0$); et que $\rho_{\mathcal{O}}(s, t) \leq \rho_{\mathcal{F}}(s, t)$ est vérifié pour toute solution optimale s .

Au vu de la continuité de $t(\cdot)$, les deux propriétés suivantes sont respectées. Elles seront utilisées par la suite.

Propriété 3.3. Si $\mathcal{Z}(s, t) < \mathcal{Z}(s^0, t)$ est vérifiée pour les solutions s et s^0 , alors :

$$\exists \varepsilon > 0 \quad \forall t' \in \Omega(\varepsilon, t) \quad (\mathcal{Z}(s, t') < \mathcal{Z}(s^0, t')).$$

Propriété 3.4. Pour toute solution s , nous avons :

$$\forall \varepsilon > 0 \quad \forall t' \in \Omega(\varepsilon, t) \quad \forall k \in \widetilde{W}^s \quad \left(t(V_k^s) - \varepsilon |\widetilde{V}_k^s| < t'(V_k^s) < t(V_k^s) + \varepsilon |\widetilde{V}_k^s| \right).$$

Ici $\widetilde{W}^s = \{k \in W^s : \widetilde{V}_k^s \neq \emptyset\}$, $\widetilde{V}_k^s = V_k^s \cap \widetilde{V}$, $t'(V_k^s) = \sum_{j \in V_k^s} t'_j$.

On pose

$$K(s, t) = \{k \in W^s : t(V_k^s) = c(s, t)\}. \quad (3.5)$$

Autrement dit, $K(s, t)$ est l'ensemble des stations les plus chargées de la solution s pour un vecteur t . Évidemment, $K(s, t)$ n'est pas vide. Pour toute solution faisable s telle que $W^s \neq K(s, t)$ le paramètre suivant est introduit :

$$\Delta(s) = \min \left\{ \frac{t(V_k^s) - t(V_l^s)}{|\widetilde{V}_k^s| + |\widetilde{V}_l^s|} : k \in K(s, t), l \in W^s \setminus K(s, t) \right\}. \quad (3.6)$$

Notons que pour tout $k \in K(s, t)$ et tout $l \in W^s \setminus K(s, t)$, $t(V_k^s) - t(V_l^s) > 0$. C'est pourquoi, comme $\widetilde{W}^s \neq \emptyset$, il existe au moins une paire d'indices $k^* \in K(s, t)$ et $l^* \in W^s \setminus K(s, t)$ telle que $|\widetilde{V}_{k^*}^s| + |\widetilde{V}_{l^*}^s| \neq 0$ et, par conséquent, $\Delta(s)$ est un nombre positif fini. Il est supposé que $\Delta(s) = +\infty$ dans le cas où $W^s = K(s, t)$.

Propriété 3.5. Pour toute solution faisable s , la formule suivante est vérifiée :

$$\forall t' \in \Omega(\Delta(s), t) \quad \left(c(s, t') = \max_{k \in K(s, t)} t'(V_k^s) \right). \quad (3.7)$$

Démonstration. Notons que si $W^s = K(s, t)$ la formule (3.7) est immédiate. Par conséquent, nous supposons que $W^s \neq K(s, t)$. En utilisant la définition de $\Delta(s)$, nous avons :

$$\forall k \in K(s, t) \quad \forall l \in W^s \setminus K(s, t) \quad \left(\Delta(s)(|\widetilde{V}_k^s| + |\widetilde{V}_l^s|) \leq t(V_k^s) - t(V_l^s) \right). \quad (3.8)$$

D'où, en utilisant la Propriété 3.4, nous dérivons :

$$\begin{aligned} \min_{k \in K(s, t)} t'(V_k^s) &\stackrel{\text{Propriété 3.4}}{\geq} \min_{k \in K(s, t)} \left\{ t(V_k^s) - \Delta(s) |\widetilde{V}_k^s| \right\} \\ &\stackrel{(3.8)}{\geq} \max_{l \in W^s \setminus K(s, t)} \left\{ t(V_l^s) + \Delta(s) |\widetilde{V}_l^s| \right\} \stackrel{\text{Propriété 3.4}}{\geq} \max_{l \in W^s \setminus K(s, t)} t'(V_l^s), \end{aligned}$$

où au moins la première ou la troisième inégalité est stricte, ce qui implique (3.7). \square

Cette propriété montre que si $k \notin K(s, t)$, alors $t'(V_k^s) < c(s, t')$ pour tout vecteur t' appartenant au $\Delta(s)$ -voisinage. En conséquence, la stabilité de la solution étudiée est liée à l'affectation des opérations aux stations appartenant à l'ensemble $K(s, t)$.

3.3 Exemple

Dans cette section, un exemple est présenté afin d'illustrer les notations introduites.

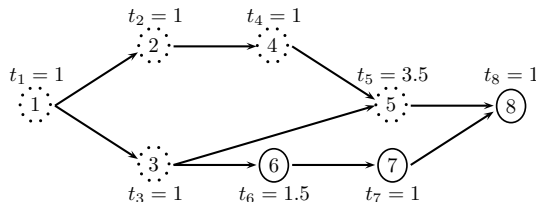


FIGURE 3.2 – Contraintes de précédence.

Soit $V = \{1, 2, \dots, 8\}$, $\tilde{V} = \{1, 2, \dots, 5\}$, $t = (1, 1, 1, 1, 3.5, 1.5, 1, 1)$, $c = 5$, $m = 5$. Les contraintes de précédence sont données par le graphe orienté acyclique représenté dans la Figure 3.2, où les opérations appartenant à \tilde{V} sont en pointillé.

Soit s^1 une solution faisable telle que $V_1^{s^1} = \{1, 2, 3, 4\}$, $V_2^{s^1} = \{5\}$, $V_3^{s^1} = \{6, 7, 8\}$ (voir Figure 3.3). Nous pouvons calculer $t(V_1^{s^1}) = 4$, $t(V_2^{s^1}) = 3.5$ et $t(V_3^{s^1}) = 3.5$. Par conséquent, $c(s^1, t) = \max\{t(V_1^{s^1}), t(V_2^{s^1}), t(V_3^{s^1})\} = 4$, c'est-à-dire $\mathcal{Z}(s^1, t) = 12$.

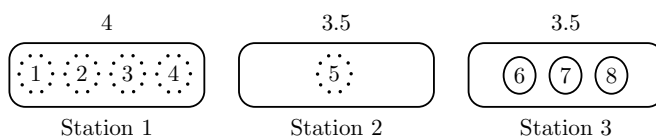


FIGURE 3.3 – Solution s^1 .

Soit s^2 une autre solution faisable telle que $V_1^{s^2} = \{1, 3, 6\}$, $V_2^{s^2} = \{2, 4, 7\}$, $V_3^{s^2} = \{5, 8\}$ (voir Figure 3.4). Par conséquent, $t(V_1^{s^2}) = 3.5$, $t(V_2^{s^2}) = 3$, $t(V_3^{s^2}) = 4.5$. De manière similaire, nous obtenons $c(s^2, t) = \max\{t(V_1^{s^2}), t(V_2^{s^2}), t(V_3^{s^2})\} = 4.5$, $\mathcal{Z}(s^2, t) = 13.5$.

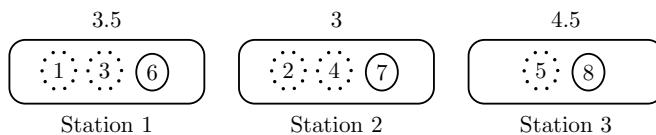


FIGURE 3.4 – Solution s^2 .

En utilisant (3.5), $K(s^1, t) = \{1\}$ et $K(s^2, t) = \{3\}$, et conformément à (3.6) :

$$\Delta(s^1) = \min \left\{ \frac{4 - 3.5}{4 + 1}, \frac{4 - 3.5}{4 + 0} \right\} = \frac{1}{10},$$

$$\Delta(s^2) = \min \left\{ \frac{4.5 - 3.5}{1 + 2}, \frac{4.5 - 3}{1 + 2} \right\} = \frac{1}{3}.$$

Cela signifie que dans le cadre de $\frac{1}{10}$ -voisinage et $\frac{1}{3}$ -voisinage de t les première et troisième stations restent les plus chargées pour s^1 and s^2 , respectivement.

Dans la section suivante, le comportement des solutions faisables sous VTEO est étudié.

3.4 Solutions faisables

Rappelons que la solution $s \in \mathbf{S}_{\mathcal{F}}(t)$ peut perdre sa faisabilité pour un nouveau vecteur perturbé $t^* \in \mathbb{R}_+^n$, c'est-à-dire $s \notin \mathbf{S}_{\mathcal{F}}(t^*)$, seulement si le temps d'exécution sur la station la plus chargée devient supérieur à c , c'est-à-dire si $c(s, t^*) > c$.

Le théorème suivant présente la formule de calcul du \mathcal{F} -rayon de stabilité, défini comme le seuil limite des VTEO tel que la solution faisable étudiée préserve sa faisabilité.

Théorème 3.1. *Le \mathcal{F} -rayon de stabilité $\rho_{\mathcal{F}}(s, t)$ de $s \in \mathbf{S}_{\mathcal{F}}(t)$ est calculé de la manière suivante :*

$$\rho_{\mathcal{F}}(s, t) = \min_{k \in \widetilde{W}^s} \frac{c - t(V_k^s)}{|\widetilde{V}_k^s|}. \quad (3.9)$$

Démonstration. Pour simplifier la démonstration, les notations suivantes sont introduites : ρ et φ sont les parties gauche et droite de l'égalité (3.9), respectivement. Il est évident que φ est un nombre non-négatif, car $s \in \mathbf{S}_{\mathcal{F}}(t)$.

Pour prouver la formule (3.9), nous montrons successivement que les inégalités $\rho \geq \varphi$ et $\rho \leq \varphi$ sont vérifiées.

Tout d'abord, nous prouvons que $\rho \geq \varphi$. Pour ce faire, il est suffisant de vérifier que

$$\forall t' \in \Omega(\varphi, t) \quad (s \in \mathbf{S}_{\mathcal{F}}(t')). \quad (3.10)$$

Si $\varphi = 0$, l'inégalité $\rho \geq \varphi$ est évidente. Soit $\varphi > 0$, $t' \in \Omega(\varphi, t)$. Vu la définition de φ nous avons :

$$\forall k \in \widetilde{W}^s \quad (\varphi |\widetilde{V}_k^s| \leq c - t(V_k^s)).$$

D'où, en utilisant la Propriété 3.4, nous obtenons :

$$c - t'(V_k^s) > c - (t(V_k^s) + \varphi |\widetilde{V}_k^s|) \geq 0, \quad k \in \widetilde{W}^s.$$

Par conséquent, en sachant que $c \geq t'(V_k^s)$, $k \in W^s \setminus \widetilde{W}^s$, nous concluons que

$$\forall t' \in \Omega(\varphi, t) \quad \forall k \in W^s \quad (c \geq t'(V_k^s)).$$

Autrement dit, $c \geq c(s, t')$, $t' \in \Omega(\varphi, t)$, et donc la formule (3.10) est vérifiée.

Maintenant, montrons que $\rho \leq \varphi$, ce qui équivaut à prouver que :

$$\forall \varepsilon > \varphi \quad \exists t^* \in \Omega(\varepsilon, t) \quad (s \notin \mathbf{S}_{\mathcal{F}}(t^*)). \quad (3.11)$$

Pour prouver la formule (3.11), la définition de φ est utilisée. Par définition, il existe $k^* \in \widetilde{W}^s$ tel que $\varphi|\widetilde{V}_{k^*}^s| = c - t(V_{k^*}^s)$. Alors, en supposant que $\varepsilon > \varphi$, $t^* \in \Omega(\varepsilon, t)$, où

$$t_j^* = \begin{cases} t_j + \delta, & \text{si } j \in \widetilde{V}_{k^*}^s, \\ t_j & \text{sinon,} \end{cases}$$

$$\varphi < \delta < \varepsilon,$$

nous obtenons :

$$c - t^*(V_{k^*}^s) = c - (t(V_{k^*}^s) + \delta|\widetilde{V}_{k^*}^s|) = (\varphi - \delta)|\widetilde{V}_{k^*}^s| < 0.$$

Il s'en suit que $c(s, t^*) > c$, c'est-à-dire $s \notin \mathbf{S}_{\mathcal{F}}(t^*)$, et donc (3.11) est vérifié. \square

Le théorème 3.1 implique les conséquences suivantes.

Conséquence 3.1. *La solution $s \in \mathbf{S}_{\mathcal{F}}(t)$ n'est pas \mathcal{F} -stable ssi la condition suivante est vérifiée :*

$$\exists k \in \widetilde{W}^s \quad (c = t(V_k^s)).$$

Démonstration. Suffisance. Soit $k \in \widetilde{W}^s$ tel que $c = t(V_k^s)$. Alors, d'après (3.9), nous avons $\rho_{\mathcal{F}}(s, t) = 0$ et s n'est donc pas \mathcal{F} -stable.

Nécessité. Soit $s \in \mathbf{S}_{\mathcal{F}}(t)$ une solution qui n'est pas \mathcal{F} -stable, alors $\rho_{\mathcal{F}}(s, t) = 0$. D'où, en utilisant (3.9), nous concluons qu'il existe $k \in \widetilde{W}^s$ tel que $c = t(V_k^s)$. \square

Conséquence 3.2. *Le problème de la recherche du \mathcal{F} -rayon de stabilité de $s \in \mathbf{S}_{\mathcal{F}}(t)$ est polynomial.*

Démonstration. Un algorithme de recherche de $\rho_{\mathcal{F}}(s, t)$ consiste à analyser de manière séquentielle les stations de s . Il peut être décrit comme suit : pour la station courante $k \in W^s$, si $\widetilde{V}_k^s \neq \emptyset$ calculer l'expression $\varphi(k) = \frac{c-t(V_k^s)}{|\widetilde{V}_k^s|}$ (cela se calcule en $\mathcal{O}(|V_k^s|)$), sinon analyser la station suivante. Continuer jusqu'à ce que toutes les stations soient analysées ou qu'une station $k \in \widetilde{W}^s$ telle que $\varphi(k) = 0$ soit trouvée. Dans ce dernier cas, $\rho_{\mathcal{F}}(s, t) = 0$, sinon $\rho_{\mathcal{F}}(s, t) = \min_{k \in \widetilde{W}^s} \varphi(k)$. Donc, dans le pire des cas, où $\widetilde{W}^s = W^s$ et aucun $\varphi(k)$, $k \in W^s$, n'est égal à 0, cet algorithme se réalise en temps polynomial : $\mathcal{O}(\sum_{k=1}^{m^s} |V_k^s|) = \mathcal{O}(|V|) = \mathcal{O}(n)$. \square

Notons que le \mathcal{F} -rayon de stabilité peut être considéré comme une mesure de performance des solutions faisables. Ainsi, plus la valeur du \mathcal{F} -rayon de stabilité est grande, moins la solution faisable est sensible aux VTEO.

La section suivante porte sur l'évaluation du $\widehat{\mathcal{F}}$ -rayon de stabilité pour les solutions quasi-faisables.

3.5 Solutions quasi-faisables

Rappelons que pour toute solution quasi-faisable $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$, l'inégalité $c(s, t) > c$ est vérifiée. Mais, une telle solution peut devenir faisable. Cela peut arriver si pour un nouveau vecteur perturbé $t^* \in \mathbb{R}_+^n$, $c(s, t^*) \leq c$. Cependant, si pour une solution quasi-faisable, il existe une station sans aucune opération incertaine et la charge de cette station est supérieure à c , alors la solution reste toujours quasi-faisable.

Dans ce qui suit, l'ensemble des stations de la solution s dont la charge est supérieure à c est noté \widehat{W}^s .

Théorème 3.2. *Pour le $\widehat{\mathcal{F}}$ -rayon de stabilité $\rho_{\widehat{\mathcal{F}}}(s, t)$ de la solution $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$, la borne inférieure suivante est vérifiée :*

$$\rho_{\widehat{\mathcal{F}}}(s, t) \geq \max_{k \in \widehat{W}^s} \frac{t(V_k^s) - c}{|\widetilde{V}_k^s|}. \quad (3.12)$$

Démonstration. Comme pour le Théorème 3.1, les notations supplémentaires sont introduites : ρ et φ sont les parties gauche et droite de (3.12), respectivement. Il est facile de voir que φ peut être égal à $+\infty$. Cela peut arriver s'il existe un indice $k \in \widehat{W}^s$ tel que $\widetilde{V}_k^s = \emptyset$. Ceci signifie que la solution s ne deviendra jamais faisable. Dans le cas contraire, φ est un nombre positif fini, car $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$. C'est pourquoi, par la suite, nous supposons que $\widetilde{V}_k^s \neq \emptyset$, pour tout $k \in \widehat{W}^s$.

Pour prouver la formule (3.12), nous montrons que l'inégalité $\rho \geq \varphi$ est vérifiée. Pour ce faire, il est suffisant de vérifier que

$$\forall t' \in \Omega(\varphi, t) \quad (s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t')). \quad (3.13)$$

Si $\varphi = 0$, l'inégalité $\rho \geq \varphi$ est directe. Soit $\varphi > 0$, $t' \in \Omega(\varphi, t)$. Par définition de φ , nous avons :

$$\exists k^* \in \widehat{W}^s \quad (\varphi |\widetilde{V}_{k^*}^s| = t(V_{k^*}^s) - c).$$

D'où, en utilisant la Propriété 3.4, nous obtenons :

$$t'(V_{k^*}^s) - c > (t(V_{k^*}^s) - \varphi |\widetilde{V}_{k^*}^s|) - c = 0.$$

Il s'en suit que $c(s, t') > c$, $t' \in \Omega(\varphi, t)$, c'est-à-dire la formule (3.13) est vérifiée. \square

Le théorème 3.2 implique les conséquences suivantes.

Conséquence 3.3. *Toute solution quasi-faisable est $\widehat{\mathcal{F}}$ -stable.*

Démonstration. Puisque pour toute solution $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$, les inégalités suivantes $c > t(V_k^s)$, $k \in \widehat{W}^s$ sont vérifiées, alors nous avons $\rho_{\widehat{\mathcal{F}}}(s, t) > 0$, d'après (3.12). \square

La conséquence 3.3 signifie qu'il existe $\varepsilon^* > 0$ tel que, quel que soit le vecteur perturbé t' dans $\Omega(\varepsilon^*, t)$, aucune solution quasi-faisable ne devient faisable. Par conséquent, la valeur du $\widehat{\mathcal{F}}$ -rayon de stabilité des solutions quasi-faisables est toujours positive.

Conséquence 3.4. *Le problème de la recherche de la borne inférieure du $\widehat{\mathcal{F}}$ -rayon de stabilité de la solution $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$ est polynomial.*

Démonstration. L'algorithme de recherche de $\rho_{\widehat{\mathcal{F}}}(s, t)$ est similaire à celui présenté dans la Conséquence 3.2. Il consiste à analyser de manière successive les stations de la solution s . Pour chaque station $k \in W^s$, l'expression $\varphi(k) = \frac{t(V_k^s) - c}{|V_k^s|}$ est calculée, si $t(V_k^s) > c$ (se calcule en $\mathcal{O}(|V_k^s|)$), sinon la station suivante est analysée. On continue jusqu'à ce que toutes les stations soient analysées ou qu'une station $k \in \widehat{W}^s$, telle que $\widetilde{V}_k^s = \emptyset$, soit trouvée. Dans ce cas, $\rho_{\widehat{\mathcal{F}}}(s, t) = +\infty$, sinon $\max_{k \in \widehat{W}^s} \varphi(k)$ est pris en tant que borne inférieure. Ainsi, dans le pire des cas, où $\widehat{W}^s = W^s$ et aucun $\varphi(k)$, $k \in W^s$ n'est égal à $+\infty$, cet algorithme se réalise en temps polynomial : $\mathcal{O}(\sum_{k=1}^{m^s} |V_k^s|) = \mathcal{O}(|V|) = \mathcal{O}(n)$. \square

3.6 Solutions optimales

Ici la stabilité des solutions optimales sous VTEO est étudiée. Le but est de trouver les conditions d'existence d'un voisinage de t tel que, quel que soit le vecteur perturbé t' dans ce voisinage, la solution en question reste optimale, c'est-à-dire elle appartient à $\mathbf{S}_{\mathcal{O}}(t')$.

Il existe deux cas, où la solution optimale s peut perdre son optimalité pour un nouveau vecteur perturbé t^* :

- s perd sa faisabilité (ce cas est étudié dans la Section 3.4) ;
- s reste faisable, mais il existe $s^0 \in \mathbf{S}_{\mathcal{F}}(t^*)$ tel que $\mathcal{Z}(s^0, t^*) < \mathcal{Z}(s, t^*)$.

En suivant les Propriétés 3.2 et 3.3 ainsi que la Conséquence 3.3, le deuxième cas est seulement possible si s^0 est également une solution optimale du problème initial, c'est-à-dire $s^0 \in \mathbf{S}_{\mathcal{O}}(t)$.

Introduisons la notation suivante :

$$\Upsilon(s, t) = \{\widetilde{V}_k^s : k \in K(s, t)\},$$

où $\Upsilon(s, t)$ représente l'ensemble qui contient les ensembles d'opérations incertaines affectés aux stations les plus chargées. Il est évident que $\Upsilon(s, t) \neq \emptyset$, mais il peut contenir l'ensemble vide comme élément.

En utilisant l'inégalité évidente $m^s \cdot c(s, t) \geq t(V)$, $s \in \mathbf{S}_{\mathcal{F}}(t)$, nous avons $m_{\min} \geq \lceil \frac{t(V)}{c} \rceil$, où $m_{\min} = \min\{m^s : s \in \mathbf{S}_{\mathcal{F}}(t)\}$.

Remarque 3.3. Pour simplifier la description des résultats obtenus, nous supposons que $m < 2m_{\min}$.

Afin de faciliter la présentation de la recherche de condition de stabilité des solutions optimales, nous la divisons en trois parties représentées par les lemmes suivants.

Lemme 3.1. Si pour $s, s^0 \in \mathbf{S}_{\mathcal{O}}(t)$ l'inclusion $\Upsilon(s, t) \subseteq \Upsilon(s^0, t)$ n'est pas vérifiée, alors s n'est pas \mathcal{O} -stable.

Démonstration. Si l'inclusion $\Upsilon(s, t) \subseteq \Upsilon(s^0, t)$ ne se vérifie pas, alors il existe au moins un indice $f \in K(s, t)$ tel que l'ensemble \tilde{V}_f^s n'appartient pas à $\Upsilon(s^0, t)$.

Deux cas sont possibles.

Cas 1 : $\tilde{V}_f^s = \emptyset$. Cela signifie que, pour tout $k \in K(s^0, t)$, nous avons $\tilde{V}_k^{s^0} \neq \emptyset$.

Alors, si nous introduisons t_j^* de la manière suivante :

$$t_j^* = \begin{cases} t_j - \delta, & \text{si } j \in \tilde{V}, \\ t_j & \text{sinon,} \end{cases}$$

où $0 < \delta < \varepsilon < \min\{\Delta(s), \Delta(s^0)\}$, nous obtenons que $t^* \in \Omega(\varepsilon, t)$.

Conformément aux Propriétés 3.1 et 3.5, cela nous donne que

$$\begin{aligned} m^s \cdot c(s, t^*) &\stackrel{\text{Propriété 3.5}}{=} m^s \cdot \max_{k \in K(s, t)} \sum_{j \in V_k^s} t_j^* = m^s \cdot \max \left\{ \sum_{j \in V_f^s} t_j^*, \max_{k \in K(s, t) \setminus \{f\}} \sum_{j \in V_k^s} t_j^* \right\} \\ &\stackrel{\tilde{V}_f^s = \emptyset}{=} m^s \cdot \max \left\{ \sum_{j \in V_f^s} t_j, \max_{k \in K(s, t) \setminus \{f\}} \left\{ \sum_{j \in V_k^s} t_j - \delta |\tilde{V}_k^s| \right\} \right\} \\ &= m^s \cdot \max \left\{ c(s, t), \max_{k \in K(s, t) \setminus \{f\}} \left\{ c(s, t) - \delta |\tilde{V}_k^s| \right\} \right\} \stackrel{\text{Propriété 3.1}}{=} m^{s^0} \cdot c(s^0, t) \\ &= m^{s^0} \cdot \max_{k \in K(s^0, t)} \sum_{j \in V_k^{s^0}} t_j \stackrel{\tilde{V}_k^{s^0} \neq \emptyset, k \in K(s^0, t)}{>} m^{s^0} \cdot \max_{k \in K(s^0, t)} \left\{ \sum_{j \in V_k^{s^0}} t_j - \delta |\tilde{V}_k^{s^0}| \right\} \\ &= m^{s^0} \cdot \max_{k \in K(s^0, t)} \sum_{j \in V_k^{s^0}} t_j^* \stackrel{\text{Propriété 3.5}}{=} m^{s^0} \cdot c(s^0, t). \end{aligned}$$

Cas 2 : $\tilde{V}_f^s \neq \emptyset$. Pour une opération arbitraire $p \in \tilde{V}_f^s$, nous considérons deux sous-cas :

• **Sous-cas 2.1 :** Il existe un ensemble $\tilde{V}_g^{s^0} \in \Upsilon(s^0, t)$, $g \in K(s^0, t)$ tel que $p \in \tilde{V}_g^{s^0}$. Comme $\tilde{V}_f^s \neq \tilde{V}_g^{s^0}$, de nouveau deux sous-cas sont possibles :

- **Sous-sous-cas 2.1.1** : Il existe $s \in \tilde{V}_f^s$ tel que $s \notin \tilde{V}_g^{s^0}$. Alors, en introduisant t_j^*

$$t_j^* = \begin{cases} t_j + \delta, & \text{si } j \in \{p, s\}, \\ t_j & \text{sinon,} \end{cases}$$

où $0 < \delta < \varepsilon$, nous obtenons que $t^* \in \Omega(\varepsilon, t)$. En utilisant les Propriété 3.1 et la Remarque 3.3, nous obtenons :

$$\begin{aligned} m^s \cdot c(s, t^*) &= m^s \cdot \max_{i \in W^s} \sum_{j \in V_i^s} t_j^* = m^s \cdot \max \left\{ \sum_{j \in V_f^s} t_j^*, \max_{i \in W^s \setminus \{f\}} \sum_{j \in V_i^s} t_j^* \right\} \\ &= m^s \cdot \max \left\{ \sum_{j \in V_f^s} t_j + 2\delta, \max_{i \in W^s \setminus \{f\}} \sum_{j \in V_i^s} t_j \right\} \underset{f \in K(s, t)}{=} m^s \cdot (c(s, t) + 2\delta) \\ &\underset{\text{Propriété 3.1}}{=} m^{s^0} \cdot c(s^0, t) + 2m^s \delta \underset{\text{Remarque 3.3}}{\geq} m^{s^0} \cdot c(s^0, t) + m^{s^0} \delta \\ &\underset{g \in K(s^0, t)}{=} m^{s^0} \cdot \max \left\{ \sum_{j \in V_g^{s^0}} t_j + \delta, \max_{i \in W^{s^0} \setminus \{g\}} \sum_{j \in V_i^{s^0}} t_j \right\} \\ &= m^{s^0} \cdot \max \left\{ \sum_{j \in V_g^{s^0}} t_j^*, \max_{i \in W^{s^0} \setminus \{g\}} \sum_{j \in V_i^{s^0}} t_j^* \right\} = m^{s^0} \cdot c(s^0, t^*). \end{aligned}$$

- **Sous-sous-cas 2.1.2** : Il existe $s \in \tilde{V}_g^{s^0}$ tel que $s \notin \tilde{V}_f^s$. Alors, en introduisant

$$t_j^* = \begin{cases} t_j + \delta, & \text{si } i = p, \\ t_j - \delta, & \text{si } i = s, \\ t_j & \text{sinon,} \end{cases}$$

où $0 < \delta < \varepsilon$, nous dérivons que $t^* \in \Omega(\varepsilon, t)$ et, en utilisant la Propriété 3.1, nous avons :

$$\begin{aligned} m^s \cdot c(s, t^*) &= m^s \cdot \max_{i \in W^s} \sum_{j \in V_i^s} t_j^* = m^s \cdot \max \left\{ \sum_{j \in V_f^s} t_j^*, \max_{i \in W^s \setminus \{f\}} \sum_{j \in V_i^s} t_j^* \right\} = \\ &= m^s \cdot \max \left\{ \sum_{j \in V_f^s} t_j + \delta, \max_{i \in W^s \setminus \{f\}} \sum_{j \in V_i^s} t_j^* \right\} \underset{f \in K(s, t)}{=} m^s \cdot c(s, t) + m^s \delta \\ &\underset{\text{Propriété 3.1}}{=} m^{s^0} \cdot c(s^0, t) + m^s \delta > m^{s^0} \cdot c(s^0, t) = m^{s^0} \cdot \max \left\{ \sum_{j \in V_g^{s^0}} t_j, \max_{i \in W^{s^0} \setminus \{g\}} \sum_{j \in V_i^{s^0}} t_j \right\} \end{aligned}$$

$$= m^{s^0} \cdot \max \left\{ \sum_{j \in V_g^{s^0}} t_j^*, \max_{i \in W^{s^0} \setminus \{g\}} \sum_{j \in V_i^{s^0}} t_j^* \right\} = m^{s^0} \cdot c(s^0, t^*).$$

• **Sous-cas 2.2** : Aucun ensemble dans $\Upsilon(s^0, t)$ ne contient l'opération p . Alors, en introduisant

$$t_j^* = \begin{cases} t_j + \delta, & \text{si } j = p, \\ t_j & \text{sinon,} \end{cases}$$

où $0 < \delta < \varepsilon$, nous obtenons que $t^* \in \Omega(\varepsilon, t)$ et, en utilisant la Propriété 3.1, nous dérivons le même résultat que dans le Sous-sous-cas 2.1.2.

En résumant tous les cas décrits ci-dessus, nous pouvons conclure que

$$\forall \varepsilon < \min\{\Delta(s), \Delta(s^0)\} \quad \exists t^* \in \Omega(\varepsilon, t) \quad (\mathcal{Z}(s, t^*) > \mathcal{Z}(s^0, t^*)),$$

c'est-à-dire que

$$\forall \varepsilon < \min\{\Delta(s), \Delta(s^0)\} \quad \exists t^* \in \Omega(\varepsilon, t) \quad (s \notin \mathbf{S}_O(t^*)).$$

D'où $\Xi_O(s, t) = \emptyset$ et, conformément à la Définition 3.2, s n'est pas \mathcal{O} -stable. \square

Lemme 3.2. *Si pour $s, s^0 \in \mathbf{S}_O(t)$ l'inclusion $\Upsilon(s, t) \subseteq \Upsilon(s^0, t)$ et l'égalité $m^s = m^{s^0}$ sont vérifiées, alors*

$$\forall \varepsilon < \min\{\Delta(s), \Delta(s^0)\} \quad \forall t' \in \Omega(\varepsilon, t) \quad (\mathcal{Z}(s, t') \leq \mathcal{Z}(s^0, t')). \quad (3.14)$$

Démonstration. Pour tout $t' \in \Omega(\varepsilon, t)$, où $\varepsilon < \min\{\Delta(s), \Delta(s^0)\}$, en prenant en compte les Propriétés 3.1 et 3.5 et la définition des VTEO, nous dérivons :

$$\begin{aligned} c(s, t') &\stackrel{\text{Propriété 3.5}}{=} \max_{k \in K(s, t)} \sum_{j \in V_k^s} t'_j = \max_{k \in K(s, t)} \left\{ \sum_{j \in V_k^s} t_j + \sum_{j \in \tilde{V}_k^s} \xi_j \right\} = \max_{U \in \Upsilon(s, t)} \left\{ c(s, t) + \sum_{j \in U} \xi_j \right\} \\ &\stackrel{\text{Propriété 3.1}}{=} \max_{U \in \Upsilon(s, t)} \left\{ c(s^0, t) + \sum_{j \in U} \xi_j \right\} \stackrel{\Upsilon(s, t) \subseteq \Upsilon(s^0, t)}{\leq} \max_{U \in \Upsilon(s^0, t)} \left\{ c(s^0, t) + \sum_{j \in U} \xi_j \right\} \\ &= \max_{k \in K(s^0, t)} \sum_{j \in V_k^{s^0}} t'_j \stackrel{\text{Propriété 3.5}}{=} c(s^0, t'). \end{aligned}$$

Cela implique (3.14). \square

Lemme 3.3. *Soit $s, s^0 \in \mathbf{S}_O(t)$, $\Upsilon(s, t) \subseteq \Upsilon(s^0, t)$ et $m^s \neq m^{s^0}$. Alors*

- si $\emptyset \notin \Upsilon(s^0, t)$, s n'est pas \mathcal{O} -stable ;
- si $\emptyset \in \Upsilon(s^0, t)$, alors

- si $\Upsilon(s, t) \neq \{\emptyset\}$ et $m^s > m^{s^0}$, s n'est pas \mathcal{O} -stable ;
- sinon la formule (3.14) est vérifiée.

Démonstration. Considérons les deux cas suivants.

Cas 1 : $\emptyset \notin \Upsilon(s^0, t)$. En tenant compte de l'inclusion $\Upsilon(s, t) \subseteq \Upsilon(s^0, t)$, nous pouvons conclure que $\emptyset \notin \Upsilon(s, t)$ et, étant donné que par définition, $\Upsilon(s, t)$ et $\Upsilon(s^0, t)$ ne sont pas vides, alors ils contiennent des ensembles d'opérations incertaines.

• **Sous-cas 1.1 :** $m^s > m^{s^0}$. Selon les conclusions du Cas 1, il existe un indice $f \in K(s, t)$, tel que $\tilde{V}_f^s \neq \emptyset$, et un indice $g \in K(s^0, t)$, tel que $\tilde{V}_f^s = \tilde{V}_g^{s^0}$. En utilisant ceci, nous introduisons

$$t_j^* = \begin{cases} t_j + \delta, & \text{si } j \in \tilde{V}_f^s, \\ t_j & \text{sinon,} \end{cases}$$

où $0 < \delta < \varepsilon$. Par conséquent, $t^* \in \Omega(\varepsilon, t)$, et, en tenant compte de l'inégalité $m^s > m^{s^0}$ et de la Propriété 3.1, nous obtenons :

$$\begin{aligned} m^s \cdot c(s, t^*) &= m^s \cdot \max_{i \in W^s} \sum_{j \in V_i^s} t_j^* = m^s \cdot \max \left\{ \sum_{j \in V_f^s} t_j^*, \max_{i \in W^s \setminus \{f\}} \sum_{j \in V_i^s} t_j^* \right\} \\ &= m^s \cdot \max \left\{ \sum_{j \in V_f^s} t_j + \delta |\tilde{V}_f^s|, \max_{i \in W^s \setminus \{f\}} \sum_{j \in V_i^s} t_j \right\} \stackrel{f \in K(s, t)}{=} m^s \cdot c(s, t) + m^s \delta |\tilde{V}_f^s| \\ &\stackrel{\text{Propriété 3.1, } \tilde{V}_f^s = \tilde{V}_g^{s^0}}{=} m^{s^0} \cdot c(s^0, t) + m^s \delta |\tilde{V}_g^{s^0}| \stackrel{m^s > m^{s^0}}{\geq} m^{s^0} \cdot c(s^0, t) + m^{s^0} \delta |\tilde{V}_g^{s^0}| \\ &= m^{s^0} \cdot \max \left\{ \sum_{j \in V_g^{s^0}} t_j^*, \max_{i \in W^{s^0} \setminus \{g\}} \sum_{j \in V_i^{s^0}} t_j^* \right\} = m^{s^0} \cdot c(s^0, t^*). \end{aligned}$$

• **Sous-cas 1.2 :** $m^s < m^{s^0}$. En mettant

$$t_j^* = \begin{cases} t_j - \frac{\delta}{|\tilde{V}_k^{s^0}|}, & \text{si } j \in \tilde{V}_k^{s^0}, k \in K(s^0, t), \\ t_j & \text{sinon,} \end{cases}$$

où $0 < \delta < \varepsilon < \min\{\Delta(s), \Delta(s^0)\}$, nous obtenons que $t^* \in \Omega(\varepsilon, t)$ et, en prenant en compte la Propriété 3.5, nous dérivons :

$$\begin{aligned} c(s, t^*) &= \max_{k \in K(s, t)} \sum_{j \in V_k^s} t_j^* = \max_{k \in K(s, t)} \left\{ \sum_{j \in V_k^s \setminus \tilde{V}_k^s} t_j + \sum_{j \in \tilde{V}_k^s} t_j^* \right\} \\ &= \max_{k \in K(s, t)} \left\{ \sum_{j \in V_k^s \setminus \tilde{V}_k^s} t_j + \sum_{j \in \tilde{V}_k^s} \left(t_j - \frac{\delta}{|\tilde{V}_k^s|} \right) \right\} = \max_{k \in K(s, t)} \left\{ \sum_{j \in V_k^s} t_j - \delta \right\} = c(s, t) - \delta. \end{aligned}$$

De manière similaire, nous obtenons $c(s^0, t^*) = c(s^0, t) - \delta$. C'est pourquoi, au vu de la Propriété 3.1, nous avons :

$$m^s \cdot c(s, t^*) = m^s \cdot c(s, t) - m^s \delta = m^{s^0} \cdot c(s^0, t) - m^s \delta > m^{s^0} \cdot c(s^0, t) - m^{s^0} \delta = m^{s^0} \cdot c(s^0, t^*).$$

En résumant ces deux sous-cas, nous pouvons conclure que

$$\forall \varepsilon < \min\{\Delta(s), \Delta(s^0)\} \exists t^* \in \Omega(\varepsilon, t) \quad (\mathcal{Z}(s^0, t^*) < \mathcal{Z}(s, t^*)), \quad (3.15)$$

Autrement dit, $\Xi_{\mathcal{O}}(s, t) = \emptyset$ et, conformément à la Définition 3.2, s n'est pas \mathcal{O} -stable.

Cas 2 : $\emptyset \in \Upsilon(s^0, t)$. Considérons les deux sous-cas suivants.

• **Sous-cas 2.1 :** $\Upsilon(s, t) \neq \{\emptyset\}$ et $m^s > m^{s^0}$. Comme $\Upsilon(s, t) \neq \{\emptyset\}$ et d'après l'inclusion $\Upsilon(s, t) \subseteq \Upsilon(s^0, t)$, il existe des indices $f \in K(s, t)$ et $g \in K(s^0, t)$ tels que $\tilde{V}_f^s = \tilde{V}_g^{s^0} \neq \emptyset$. Par conséquent, en répétant les mêmes calculs comme dans le Sous-cas 1.1, nous aboutissons à la formule (3.15), c'est-à-dire s n'est pas \mathcal{O} -stable.

• **Sous-cas 2.2 :** $\Upsilon(s, t) = \{\emptyset\}$ et $m^s < m^{s^0}$. Alors, conformément aux Propriétés 3.1 et 3.5 ainsi qu'à la définition des VTEO, pour tout $t' \in \Omega(\varepsilon, t)$, où $\varepsilon < \min\{\Delta(s), \Delta(s^0)\}$, nous pouvons obtenir :

$$\begin{aligned} m^s \cdot c(s, t') &\stackrel{\text{Propriété 3.5}}{=} m^s \cdot \max_{k \in K(s, t)} \sum_{j \in V_k^s} t'_j = m^s \cdot \max_{k \in K(s, t)} \left\{ \sum_{j \in V_k^s} t_j + \sum_{j \in \tilde{V}_k^s} \xi_j \right\} \\ &= \max_{U \in \Upsilon(s, t)} \left\{ m^s \cdot c(s, t) + m^s \cdot \sum_{j \in U} \xi_j \right\} \stackrel{\text{Propriété 3.1}}{=} \max_{U \in \Upsilon(s, t)} \left\{ m^{s^0} \cdot c(s^0, t) + m^s \cdot \sum_{j \in U} \xi_j \right\} \\ &= m^{s^0} \cdot c(s^0, t) + m^s \cdot \max_{U \in \Upsilon(s, t)} \sum_{j \in U} \xi_j = \mathcal{I}. \end{aligned}$$

Si $\Upsilon(s, t) = \{\emptyset\}$, alors $\max_{U \in \Upsilon(s, t)} \sum_{j \in U} \xi_j = 0$ et nous obtenons :

$$\begin{aligned} \mathcal{I} &= m^{s^0} \cdot c(s^0, t) + m^{s^0} \cdot \max_{U \in \Upsilon(s, t)} \sum_{j \in U} \xi_j \stackrel{\text{Propriété 3.1}}{\leq} m^{s^0} \cdot c(s^0, t) + m^{s^0} \cdot \max_{U \in \Upsilon(s^0, t)} \sum_{j \in U} \xi_j \\ &= m^{s^0} \cdot \max_{U \in \Upsilon(s^0, t)} \left\{ c(s^0, t) + \sum_{j \in U} \xi_j \right\} = m^{s^0} \cdot c(s^0, t'). \end{aligned}$$

Si $\Upsilon(s, t) \neq \{\emptyset\}$, alors $m^s < m^{s^0}$ et nous avons :

$$\begin{aligned} \mathcal{I} &\stackrel{\text{Propriété 3.1}}{\leq} m^{s^0} \cdot c(s^0, t) + m^s \cdot \max_{U \in \Upsilon(s^0, t)} \sum_{j \in U} \xi_j \\ &\left[\text{Puisque } \emptyset \in \Upsilon(s^0, t), \text{ alors } \max_{U \in \Upsilon(s^0, t)} \sum_{j \in U} \xi_j \geq 0 \right] \end{aligned}$$

$$\underbrace{\leq}_{m^s < m^{s^0}} m^{s^0} \cdot c(s^0, t) + m^{s^0} \cdot \max_{U \in \Upsilon(s^0, t)} \sum_{j \in U} \xi_j = m^{s^0} \cdot \max_{U \in \Upsilon(s^0, t)} \left\{ c(s^0, t) + \sum_{j \in U} \xi_j \right\} = m^{s^0} \cdot c(s^0, t').$$

Autrement dit, nous avons la formule (3.14). \square

Les résultats des Lemmes 3.1-3.3 et les Conséquences 3.1 et 3.3 impliquent le théorème suivant.

Théorème 3.3. *La solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ n'est pas \mathcal{O} -stable ssi au moins une des conditions suivantes est vérifiée :*

- il existe $k \in \widetilde{W}^s$ tel que $c = t(V_k^s)$,
- il existe $s^0 \in \mathbf{S}_{\mathcal{O}}(t)$ tel que la suite est vérifiée :

$$\Upsilon(s, t) \subseteq \Upsilon(s^0, t) \Rightarrow (m^s \neq m^{s^0} \& (\emptyset \in \Upsilon(s^0, t) \Rightarrow (\Upsilon(s, t) \neq \{\emptyset\} \& m^s > m^{s^0}))). \quad (3.17)$$

Montrons à présent que le résultat obtenu s'applique aussi aux deux cas particuliers du SALBP-E, (à savoir SALBP-1 et SALBP-2) définis comme suit :

SALBP-1 : $\mathcal{Z}(s, t) := m^s \rightarrow \min_{s \in \mathbf{S}_{\mathcal{F}}(t)}, c(s, t) \leq c$, m n'est pas donné,

SALBP-2 : $\mathcal{Z}(s, t) := c(s, t) \rightarrow \min_{s \in \mathbf{S}_{\mathcal{F}}(t)}, c$ n'est pas donné, $m^s = m$.

Le Théorème 3.3 implique les conséquences suivantes.

Conséquence 3.5. [Sotskov *et al.*, 2006] *La solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ de SALBP-1 n'est pas \mathcal{O} -stable ssi il existe $k \in \widetilde{W}^s$ tel que $c = t(V_k^s)$.*

Conséquence 3.6. *La solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ de SALBP-2 n'est pas \mathcal{O} -stable ssi il existe $s^0 \in \mathbf{S}_{\mathcal{O}}(t)$ tel que $\Upsilon(s, t) \subseteq \Upsilon(s^0, t)$ n'est pas vérifié.*

Notons $\mathcal{O} - \text{Stab}(s, t)$ le problème de décision suivant : « La solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ est-elle \mathcal{O} -stable ? ». D'après le Théorème 3.3 et la Conséquence 3.6, pour résoudre $\mathcal{O} - \text{Stab}(s, t)$, il est impératif de connaître l'ensemble des solutions optimales pour le problème d'optimisation initial. Par conséquent, comme SALBP-E et SALBP-2 sont \mathcal{NP} -difficiles, on peut penser que $\mathcal{O} - \text{Stab}(s, t)$ n'est pas polynomial pour SALBP-2 et SALBP-E. Néanmoins, la réponse à cette question est loin d'être évidente.

Par conséquent, le problème de recherche du \mathcal{O} -rayon de stabilité de la solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ pour SALBP-2 et SALBP-E semble aussi ne pas être polynomial. Cependant, une borne supérieure utile du \mathcal{O} -rayon de stabilité de la solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ peut être calculée en temps polynomial pour SALBP-E, en utilisant le Théorème 3.1, car $\rho_{\mathcal{O}}(s, t) \leq \rho_{\mathcal{F}}(s, t)$.

Conséquence 3.7. *Pour le \mathcal{O} -rayon de stabilité $\rho_{\mathcal{O}}(s, t)$ de la solution $s \in \mathbf{S}_{\mathcal{O}}(s, t)$ de SALBP-E, la borne supérieure suivante peut être calculée :*

$$\rho_{\mathcal{O}}(s, t) \leq \min_{k \in \widetilde{W}^s} \frac{c - t(V_k^s)}{|\widetilde{V}_k^s|}.$$

Toutefois, le problème $\mathcal{O} - \text{Stab}(s, t)$ est polynomial pour SALBP-1 comme cela indique la Conséquence 3.5. Un algorithme similaire à celui présenté dans la Section 3.4 peut être appliqué pour sa résolution.

Remarquons que les solutions optimales de SALBP-1 préservent leur optimalité sous VTEO si et seulement si elles restent faisables et aucune solution quasi-faisable avec un nombre de stations moindre ne devient faisable. C'est pourquoi les Théorèmes 3.1 et 3.2 impliquent la conséquence suivante qui a été déjà démontrée dans [Sotskov *et al.*, 2006].

Conséquence 3.8. [Sotskov *et al.*, 2006] *Le \mathcal{O} -rayon de stabilité $\rho_{\mathcal{O}}(s, t)$ de la solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ pour SALBP-1 est calculé comme suit :*

$$\rho_{\mathcal{O}}(s, t) = \min \left\{ \rho_{\mathcal{F}}(s, t), \min_{s' \in \mathbf{S}_{\mathcal{F}}(t)} \left\{ \rho_{\mathcal{F}}(s', t) : m^{s'} < m^s \right\} \right\}.$$

3.7 Recherche du compromis

Quand il n'existe pas de solution faisable ayant à la fois la valeur minimale de la fonction objectif et la valeur maximale du \mathcal{F} -rayon de stabilité, il est nécessaire de trouver un compromis entre ces deux objectifs antagonistes lors de la prise de décision.

Le concept de l'optimalité au sens de Pareto [Ehrgott, 2005] peut alors être utilisé. Dans l'optimisation multiobjectif, les solutions pour lesquelles il est impossible d'améliorer l'un des objectifs sans aggraver l'un des autres sont appelées efficaces ou non-dominées au sens de Pareto.

Pour présenter formellement l'ensemble des solutions non-dominées \mathcal{SND} , la relation binaire entre deux solutions faisables quelconques s et s' exprimant la règle de dominance au sens de Pareto est introduite comme suit :

$$s \succ s' \iff \mathcal{Z}(s, t) \leq \mathcal{Z}(s', t) \ \& \ \rho_{\mathcal{F}}(s, t) \geq \rho_{\mathcal{F}}(s', t),$$

où au moins l'une des deux inégalités est stricte. Dans le cas où $s \succ s'$ nous disons que s domine s' ou s' est dominé par s .

Ainsi, nous pouvons définir l'ensemble \mathcal{SND} de l'ensemble \mathcal{S} des solutions faisables de la manière suivante :

$$\mathcal{SND} = \{s \in \mathcal{S} : \nexists s' \in \mathcal{S} \ (s' \succ s)\}.$$

L'objectif de cette section est de proposer une approche de construction de \mathcal{SND} .

Dans notre cas, la recherche de l'ensemble \mathcal{SND} est un problème difficile. C'est pourquoi on s'oriente vers le développement de méthodes de construction d'approximations de \mathcal{SND} . Pour ce faire, une procédure heuristique multi-start peut être utilisée. Chaque itération de cette procédure consiste en deux phases : la construction d'une solution faisable

(en utilisant l'heuristique $\mathcal{H}(c)$ présentée ci-dessous) et la comparaison de cette solution avec des solutions déjà construites. Les deux phases se répètent de manière interchangeable jusqu'à ce que le critère d'arrêt soit satisfait.

Au début de la procédure heuristique, un intervalle entier $[c_{\min}, c_{\max}]$ définit l'ensemble de $c_{\max} - c_{\min} + 1$ valeurs admissibles pour c . Les valeurs de c sont modifiées de manière séquentielle durant la phase de construction. Au départ, c reçoit la valeur c_{\max} , qui est ensuite décrémentée après chaque $\frac{T_{\max}}{c_{\max} - c_{\min} + 1}$ secondes, où T_{\max} est le temps de résolution maximum autorisé.

Heuristique $\mathcal{H}(c)$

Tenant compte de la valeur courante de c , l'heuristique $\mathcal{H}(c)$ construit une solution faisable en affectant autant d'opérations que possible à la station courante. Initialement, la solution contient seulement une station vide. L'heuristique affecte des opérations à cette station jusqu'à ce qu'aucune opération ne puisse être affectée à cause des contraintes existantes. Ensuite, une nouvelle station est ouverte, elle devient la station courante et des opérations peuvent lui être affectées.

Cela continue jusqu'à ce que toutes les opérations soient affectées ou qu'il soit impossible d'ouvrir une nouvelle station sans dépasser le nombre de stations autorisé m . Dans ce dernier cas, la solution est reconstruite.

Soient s la solution en cours de construction, k l'indice de la station courante, et soit V_k^s l'ensemble des opérations affectées à la station k de la solution s . Pour choisir une opération à affecter à la station k , une liste $\mathcal{CL}(k, c)$ d'opérations candidates est générée. La liste $\mathcal{CL}(k, c)$ contient toutes les opérations qui peuvent être affectées à la station k . Cette liste est construite comme suit : l'ensemble des opérations non-affectées est examiné et l'opération j est placée dans $\mathcal{CL}(k, c)$, si les deux conditions suivantes sont vérifiées :

- tous ses prédécesseurs ont déjà été affectés ;
- $t(V_k^s \cup \{j\}) \leq c$.

Si $\mathcal{CL}(k, c) = \emptyset$, alors il n'y a plus d'opérations qui peuvent être affectées à la station courante. Ainsi, une nouvelle station doit être ouverte, si c'est possible, et $\mathcal{CL}(k + 1, c)$ est construit. Sinon ($\mathcal{CL}(k, c) \neq \emptyset$), une opération j est choisie au hasard dans $\mathcal{CL}(k, c)$ et affectée à V_k^s , $\mathcal{CL}(k, c)$ est reconstruit.

Notons que moins il y a d'opérations incertaines affectées par station, plus nous avons de chances d'avoir une plus grande valeur du \mathcal{F} -rayon de stabilité. Par conséquent, il est souhaitable d'affecter le moins possible d'opérations incertaines par station afin d'améliorer la valeur du \mathcal{F} -rayon de stabilité. Pour cela, l'opération j est choisie au hasard dans $\mathcal{CL}(k, c)$ seulement dans les cas où des opérations incertaines n'ont pas encore été affectées.

tées à la station courante ($\tilde{V}_k^s = \emptyset$) ou que $\mathcal{CL}(k, c)$ se compose uniquement d'opérations incertaines ($\mathcal{CL}(k, c) \cap \tilde{V} = \mathcal{CL}(k, c)$), sinon j est choisi dans $\mathcal{CL}(k, c) \setminus \tilde{V}$.

Afin de distinguer ces deux stratégies, la notation $\mathcal{H}_1(c)$ est utilisée pour la version de base de $\mathcal{H}(c)$ (présenté par l'Algorithme 1) et $\mathcal{H}_2(c)$ pour la deuxième version.

Algorithme 1 : Heuristique $\mathcal{H}(c)$

```

1  $i \leftarrow 0, k \leftarrow 1, V_k^s \leftarrow \emptyset, stop \leftarrow \text{faux}$ 
2 répéter
3   Construire une liste  $\mathcal{CL}(k, c)$ 
4   si  $\mathcal{CL}(k, c) \neq \emptyset$  alors
5     Choisir au hasard une opération  $j$  de  $\mathcal{CL}(k, c)$ 
6      $V_k^s \leftarrow V_k^s \cup \{j\}, i \leftarrow i + 1$ 
7   sinon si  $i = n$  alors
8      $stop \leftarrow \text{vrai}$  /* Une solution faisable est construite */
9   sinon si  $k = m_{\max}$  alors
10     $i \leftarrow 0, k \leftarrow 1, V_k^s \leftarrow \emptyset$  /* La solution construite est infaisable,
11      construire une autre solution */
12  sinon
13     $k \leftarrow k + 1, V_k^s \leftarrow \emptyset$  /* Une nouvelle station est ouverte */
13 jusqu'à  $stop = \text{vrai}$ 
14 retourner  $s$ 

```

Construction de $\mathcal{SN}\mathcal{D}$

L'ensemble $\mathcal{SN}\mathcal{D}$ est construit itérativement (Algorithme 2). Au départ, les ensembles \mathcal{S} et $\mathcal{SN}\mathcal{D}$ sont vides. \mathcal{S} reçoit de manière séquentielle toutes les solutions faisables construites par l'heuristique $\mathcal{H}(c)$, tandis que les solutions non-dominées constituent $\mathcal{SN}\mathcal{D}$, qui est actualisé à chaque fois qu'une nouvelle solution faisable est obtenue.

Chaque nouvelle solution faisable s est alors comparée aux solutions dans $\mathcal{SN}\mathcal{D}$ en appliquant les trois propositions suivantes [Ehr Gott, 2005].

Proposition 3.1. *S'il existe $s' \in \mathcal{S}$ tel que $s' \succ s$, alors*

- *il existe $s'' \in \mathcal{SN}\mathcal{D}$ tel que $s'' \succ s$;*
- *aucune solution dans $\mathcal{SN}\mathcal{D}$ n'est dominée par s .*

Proposition 3.2. *S'il existe $s' \in \mathcal{SN}\mathcal{D}$ tel que $s \succ s'$, alors $s \in \mathcal{SN}\mathcal{D}$.*

Il est facile de voir que si la condition de la Proposition 3.2 est vérifiée, alors la solution s' doit être exclue de $\mathcal{SN}\mathcal{D}$.

Proposition 3.3. *Si aucune solution de $\mathcal{SN}\mathcal{D}$ ne domine s , alors $s \in \mathcal{SN}\mathcal{D}$.*

Les Propositions 3.1, 3.2 et 3.3 montrent qu'il est nécessaire de comparer une nouvelle solution construite s seulement avec les solutions courantes de $\mathcal{SN}\mathcal{D}$, car si s est dominée, elle est nécessairement dominée par l'une des solutions de $\mathcal{SN}\mathcal{D}$, sinon s appartient à $\mathcal{SN}\mathcal{D}$.

Ainsi, en connaissant une nouvelle solution construite s , la séquence d'actions suivante est effectuée. Si $\mathcal{SN}\mathcal{D} = \emptyset$, alors s est la première solution faisable construite et, par conséquent, elle est placée dans $\mathcal{SN}\mathcal{D}$. Sinon, deux cas sont possibles : il existe une solution $s' \in \mathcal{SN}\mathcal{D}$ telle que $s' \succ s$, alors $\mathcal{SN}\mathcal{D}$ reste inchangé d'après la Proposition 3.1 ; sinon toutes les solutions dans $\mathcal{SN}\mathcal{D}$ dominées par s sont exclues et s est placé dans $\mathcal{SN}\mathcal{D}$ vu les Propositions 3.2 et 3.3.

3.8 Résultats numériques

Les expérimentations ont été effectuées sur un ordinateur ayant un processeur Pentium-IV cadencé à 3 GHz et 2.5 GB de RAM. Les algorithmes proposés ont été réalisés en utilisant le langage C++ et évalués sur 15 instances connues du SALBP (voir [http : //www.assembly - line - balancing.de](http://www.assembly-line-balancing.de)) présentées dans le Tableau 3.1.

En plus des notations introduites précédemment, le Tableau 3.1 indique :

- $t_{\min} = \min\{t_j : j \in V\}$ est le temps d'exécution minimal ;
- $t_{\max} = \max\{t_j : j \in V\}$ est le temps d'exécution maximal ;
- $t_{\text{sum}} = \sum_{j=1}^n t_j$ est la somme des temps d'exécution de toutes les opérations données ;
- $OS = \frac{2|\mathcal{A}|}{|V|(|V|-1)}$ est la densité des contraintes de précédence du graphe $G = (V, \mathcal{A})$.

Pour chaque instance 33% des opérations ont été choisies au hasard comme incertaines. Le temps de calcul disponible T_{\max} a été limité à 300 sec pour les instances où $c_{\max} - c_{\min} \leq 50$; à 600 sec pour les instances où $50 < c_{\max} - c_{\min} \leq 100$; et à 900 sec pour les autres.

Les Tableaux 3.2 et 3.3 montrent les performances des deux approches proposées (la première parmi elles utilise l'heuristique $\mathcal{H}_1(c)$, tandis que la deuxième utilise $\mathcal{H}_2(c)$). Les notations suivantes sont utilisées :

- # est la cardinalité de l'ensemble correspondant ;
- $\mathcal{SN}\mathcal{D}_1$ et $\mathcal{SN}\mathcal{D}_2$ sont les ensembles de solutions non-dominées trouvées avec la première et la deuxième approche, respectivement ;

Algorithme 2 : Construction de solutions non-dominées

```

1  $k \leftarrow 0, \mathcal{SND} \leftarrow \emptyset$ 
2 répéter
3    $T_{\text{cur}} \leftarrow 0$ 
4   répéter
5      $s \leftarrow \mathcal{H}(c_{\text{max}} - k)$  /* Génération d'une nouvelle solution  $s$  en
        utilisant l'heuristique  $\mathcal{H}$  */
6     si  $\mathcal{SND} \neq \emptyset$  alors
7        $flag \leftarrow \text{faux}$  /* Tant que  $s$  n'est pas dominée */
8       pour chaque  $s' \in \mathcal{SND}$  faire
9         si  $s \succ s'$  alors
10           $\mathcal{SND} \leftarrow \mathcal{SND} \setminus \{s'\}$ 
11          sinon si  $s' \succ s$  ou  $s' = s$  alors
12             $flag \leftarrow \text{vrai}$ , quitter la boucle /*  $s$  est dominée ou  $s$  et
                 $s'$  sont identiques */
13          si  $flag = \text{faux}$  alors
14             $\mathcal{SND} \leftarrow \mathcal{SND} \cup \{s\}$ 
15          sinon
16             $\mathcal{SND} \leftarrow \mathcal{SND} \cup \{s\}$ 
17  jusqu'à  $T_{\text{cur}} > \frac{T_{\text{max}}}{c_{\text{max}} - c_{\text{min}} + 1}$ 
18   $k \leftarrow k + 1$ 
19 jusqu'à  $k > c_{\text{max}} - c_{\text{min}}$ 

```

- \mathcal{Z}_{min} , \mathcal{Z}_{moy} , et \mathcal{Z}_{max} sont respectivement les valeurs minimale, moyenne, maximale de la fonction objectif des solutions non-dominées ;
- ρ_{min} , ρ_{moy} , et ρ_{max} donnent respectivement les valeurs minimale, moyenne et maximale de leur \mathcal{F} -rayon de stabilité.

Le Tableau 3.4 présente la comparaison entre deux approches développées. Ici $\mathcal{SND}_{\text{mut}}$ est l'ensemble des solutions non-dominées trouvées par les deux approches. Autrement dit,

$$\mathcal{SND}_{\text{mut}} = \{s \in \mathcal{SND}_1 \cup \mathcal{SND}_2 : \nexists s' \in \mathcal{SND}_1 \cup \mathcal{SND}_2 \ (s' \succ s)\}.$$

$\mathcal{S}_{\text{mut}-1}$ et $\mathcal{S}_{\text{mut}-2}$ sont les ensembles des solutions de $\mathcal{SND}_{\text{mut}}$ trouvées seulement par la première ou par la deuxième approche, respectivement, c'est-à-dire $\mathcal{S}_{\text{mut}-1} = \mathcal{SND}_{\text{mut}} \cap \mathcal{SND}_1$ et $\mathcal{S}_{\text{mut}-2} = \mathcal{SND}_{\text{mut}} \cap \mathcal{SND}_2$. Θ_1 et Θ_2 sont les pourcentages des solutions de \mathcal{SND}_1 et de \mathcal{SND}_2 , respectivement, qui appartiennent à $\mathcal{SND}_{\text{mut}}$, c'est-à-dire $\Theta_1 =$

TABLEAU 3.1 – Instances

Nom	n	t_{\min}	t_{\max}	t_{sum}	c_{\min}	c_{\max}	m	OS	T_{\max}
Buxey	29	1	25	324	27	54	13	50.74	300
Gunther	35	1	40	483	41	81	14	59.5	300
Lutz2	89	1	10	485	11	21	49	77.55	300
Mitchell	21	1	13	105	14	39	8	70.95	300
Roszieg	25	1	13	125	14	32	10	71.67	300
Sawyer	30	1	25	324	25	75	14	44.83	300
Wee-Mag	75	2	27	1499	28	56	63	22.67	300
Barthol2	148	1	83	4234	84	170	51	25.8	600
Lutz3	89	1	74	1644	75	150	23	77.55	600
Warnecke	58	7	53	1548	54	111	31	59.1	600
Barthold	148	3	383	5634	403	805	14	25.8	900
Heskia	28	1	108	1024	138	342	8	22.49	900
Kilbridge	45	3	55	552	56	184	10	44.55	900
Mukherje	94	8	171	4208	176	351	25	44.8	900
Tonge	70	1	156	3510	160	572	23	59.42	900

$$\frac{\#\mathcal{S}_{\text{mut}-1}}{\#\mathcal{SN}\mathcal{D}_1} \cdot 100\% \text{ et } \Theta_2 = \frac{\#\mathcal{S}_{\text{mut}-2}}{\#\mathcal{SN}\mathcal{D}_2} \cdot 100\%.$$

En analysant les résultats obtenus, nous pouvons voir que la première approche a fourni des résultats légèrement meilleurs que la deuxième approche dans la majorité des cas. Par exemple, la première approche a trouvé 7.27 solutions non-dominées en moyenne, où 70.83% d'entre elles font partie de $\mathcal{SN}\mathcal{D}_{\text{mut}}$, alors que la deuxième approche a trouvé 6.93 solutions non-dominées en moyenne, où 66.02% d'entre elles font partie de $\mathcal{SN}\mathcal{D}_{\text{mut}}$.

Néanmoins, pour trois instances testées, la deuxième approche fournit des solutions substantiellement meilleures que la première approche (voir les instances de Barthol2, Warnecke et Mukherje dans le Tableau 3.4). Et, comme on pourrait s'y attendre, bien que la deuxième approche ait trouvé en moyenne moins de solutions non-dominées que la première, elle a fourni des résultats légèrement meilleurs à l'égard du \mathcal{F} -rayon de stabilité. En effet, la valeur moyenne du \mathcal{F} -rayon de stabilité est égale à 3.09, pour la deuxième approche, et 3.02, pour la première.

Il faut aussi noter le caractère complémentaire de ces deux approches pour certaines instances. Par exemple, pour les instances de Barthold et Tonge chaque approche a trouvé approximativement la moitié de $\mathcal{SN}\mathcal{D}_{\text{mut}}$ sans aucune solution commune. De même, pour les instances Sawyer et Wee-Mag, la même quantité de solutions constituant l'ensemble

TABLEAU 3.2 – Première approche

Nom	$\#\mathcal{SN}\mathcal{D}_1$	\mathcal{Z}_{\min}	\mathcal{Z}_{moy}	\mathcal{Z}_{\max}	ρ_{\min}	ρ_{moy}	ρ_{\max}
Buxey	5	282	304.2	343	2	3.8	5.67
Gunther	12	432	482	539	0	3.72	7.5
Lutz2	4	483	513.75	540	0	0.54	1
Mitchell	3	70	90.33	105	2.5	3.17	4
Roszieg	6	96	109.33	119	0	0.92	2
Sawyer	5	256	276.8	329	2.5	4.3	6.5
Wee-Mag	9	1504	1659.33	1952	0	3.17	5.5
Barthol2	7	4160	4301.14	4592	0	1.1	2
Lutz3	9	1540	1657.89	1890	0	1.63	2.75
Warnecke	6	1485	1537.17	1648	0.5	3.56	7
Barthold	9	4935	5160	5628	0.58	1.21	1.6
Heskia	9	682	843.11	1012	0.5	6.86	10
Kilbridge	7	368	452.71	534	0	3	4
Mukherje	8	3924	4096.13	4466	0	2.08	3.4
Tonge	10	3012	3166.5	3549	3	6.18	9

$\mathcal{SN}\mathcal{D}_{\text{mut}}$ a été trouvée par chaque approche, mais avec un pourcentage relativement petit de solutions communes trouvées par les deux approches.

Un résultat intéressant a été obtenu pour les instances de Mitchell et Roszieg, où exactement le même ensemble $\mathcal{SN}\mathcal{D}$ a été trouvé par les deux approches.

Les performances des deux approches pour l'instance de Gunther sont représentées dans la Figure 3.5. Supposons que le concepteur cherche une configuration de la ligne avec la fonction objectif se situant dans l'intervalle $[445, 460]$. Deux solutions s^1 et s^2 se trouvent dans cet intervalle : $\mathcal{Z}(s^1, t) = 448$, $\rho_{\mathcal{F}}(s^1, t) = 1$ et $\mathcal{Z}(s^2, t) = 450$, $\rho_{\mathcal{F}}(s^2, t) = 2.67$. Parmi elles, la deuxième solution pourrait être préférée, car sa valeur du \mathcal{F} -rayon de stabilité est sensiblement plus grande que celle de s^1 , même si la valeur de sa fonction objectif est légèrement plus petite.

3.9 Conclusion

Dans ce chapitre, le problème d'équilibrage des lignes d'assemblage de type SALBP-E a été considéré sous VTEO. Les conditions de stabilité et les formules du rayon de stabilité pour des solutions faisables, quasi-faisables, et optimales ont été proposées. Leur

TABLEAU 3.3 – Deuxième approche

Nom	$\#\mathcal{SN}\mathcal{D}_2$	\mathcal{Z}_{\min}	\mathcal{Z}_{moy}	\mathcal{Z}_{\max}	ρ_{\min}	ρ_{moy}	ρ_{\max}
Buxey	3	287	311.67	328	3	4	5
Gunther	11	432	483.91	560	0	3.44	7
Lutz2	3	483	517.67	570	0	0.5	1
Mitchell	3	70	90.33	105	2.5	3.17	4
Roszieg	6	96	109.33	119	0	0.92	2
Sawyer	5	256	284.6	318	2.5	4.6	6
Wee-Mag	6	1519	1612	1705	1	3.41	5
Barthol2	7	4150	4220.43	4292	0.33	1.05	2
Lutz3	4	1548	1569.75	1599	0	1.27	2
Warnecke	9	1485	1546.11	1648	1.5	4.27	7
Barthold	7	4935	4989.43	5120	0.3	0.96	1.67
Heskia	11	682	902.73	1080	0.33	5.92	8.88
Kilbridge	6	368	461.17	621	0	2.77	4
Mukherje	14	3924	4076.36	4410	0.43	3.02	5.75
Tonge	9	3018	3473.88	4202	1.66	7.1	10.67

complexité de calcul a également été examinée. Un algorithme polynomial de calcul du rayon de stabilité pour les solutions faisables a été suggéré.

Notons que des résultats pour SALBP-1 et SALBP-2 peuvent être dérivés des résultats obtenus dans ce chapitre pour SALBP-E.

Le concept de l'optimalité au sens de Pareto a été utilisé afin de chercher un compromis entre la valeur de la fonction objectif à minimiser et la valeur du rayon de stabilité à maximiser. Pour trouver les solutions non-dominées au sens de Pareto, deux approches ont été développées. Ces approches ont montré un comportement complémentaire lors des expérimentations numériques. Elles sont suffisamment efficaces pour être appliquées lors de la conception des lignes d'assemblage, en aidant les concepteurs à choisir parmi des solutions de bonne qualité celle qui est la plus stable à mettre en place.

Les résultats présentés dans ce chapitre ont donné lieu à trois communications aux conférences ETFA'2009 [Gurevsky *et al.*, 2009], ROADEF 2010 [Gurevsky *et al.*, 2010b] et URPDM 2010 [Gurevsky *et al.*, 2010c] ainsi qu'à un article soumis à la revue « Annals of Operations Research » [Gurevsky *et al.*, xxxxa].

TABLEAU 3.4 – Comparaison des approches

Nom	$\#\mathcal{SND}_{\text{mut}}$	$\#\mathcal{S}_{\text{mut}-1}$	$\Theta_1, \%$	$\#\mathcal{S}_{\text{mut}-2}$	$\Theta_2, \%$
Buxey	5	5	100	1	33.33
Gunther	13	12	100	4	36.36
Lutz2	4	4	100	2	66.67
Mitchell	3	3	100	3	100
Roszieg	6	6	100	6	100
Sawyer	7	5	100	5	100
Wee-Mag	8	5	55.56	5	83.33
Barthol2	7	1	14.29	7	100
Lutz3	10	8	88.89	2	50
Warnecke	9	2	33.33	8	88.88
Barthold	5	2	22.22	3	42.86
Heskia	9	9	100	0	0
Kilbridge	7	6	85.71	2	33.33
Mukherje	14	1	12.5	14	100
Tonge	10	5	50	5	55.56

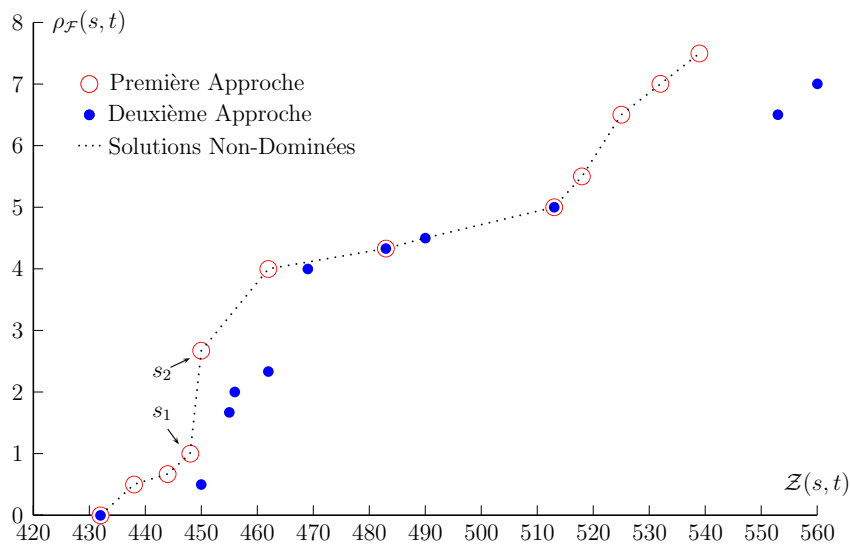


FIGURE 3.5 – Comparaison de deux approches pour l'instance de Gunther.

Chapitre 4

Équilibrage de lignes d'usinage à boîtiers multibroches : analyse de sensibilité

Sommaire

4.1	Introduction	63
4.2	Définitions et propriétés de base	65
4.3	Solutions faisables	68
4.4	Solutions quasi-faisables	72
4.5	Solutions optimales	78
4.6	Recherche du compromis	80
4.7	Résultats numériques	81
4.8	Conclusion	83

4.1 Introduction

Dans ce chapitre, nous considérons le cas de lignes d'usinage dans lesquelles chaque station est munie d'au moins un boîtier (Figures 4.1 et 4.2). Ces boîtiers peuvent comporter un ou plusieurs outils. Dans ce dernier cas, il s'agit de boîtiers multibroches qui sont utilisés pour effectuer simultanément un ensemble d'opérations. Les boîtiers de chaque station sont activés séquentiellement, dans un ordre fixe. L'activation des boîtiers n'est pas nécessairement synchronisée entre les différentes stations. L'ordre de déclenchement des boîtiers et l'affectation des outils à chaque boîtier sont à définir lors de la conception de la ligne.

Les lignes d'usinage étudiées appartiennent à la classe des lignes spécialisées, c'est-à-dire qu'une telle ligne est généralement dédiée à la fabrication de masse d'un type ou d'une famille de pièces proches. Par conséquent, sa configuration doit être choisie en analysant minutieusement la pièce à fabriquer.

À l'étape de la configuration de telles lignes, le nombre de stations ainsi que leur composition en termes de boîtiers (leur nombre et l'ordre d'activation) doivent être déterminés, mais aussi les opérations affectées à chaque boîtier que nous appelons blocs d'opérations.

Le problème d'équilibrage de telles lignes consiste alors à répartir les opérations d'usinage à des boîtiers d'usinage et à des stations de telle sorte que toutes les contraintes techniques et technologiques recensées soient respectées et que le coût de la ligne soit le plus petit possible. Le nombre de stations à équiper et le nombre de boîtiers à installer définissent principalement le coût de la ligne, que l'on vise à minimiser lors de la répartition des opérations.

Ce problème d'optimisation, nommé TLBP (Transfer Line Balancing Problem), a été déjà étudié dans [Dolgui *et al.*, 2006a; Guschinskaya *et al.*, 2008; Guschinskaya and Dolgui, 2009], où différentes méthodes exactes et approchées ont été développées.

Comme dans le Chapitre 3, une attention particulière est portée ici sur la conception préliminaire de ce type de lignes d'usinage en présence d'un ensemble d'opérations dont le temps d'exécution est susceptible de varier. Tout d'abord, nous étudions des aspects théoriques de l'analyse de sensibilité des solutions de ce problème face aux variations des temps d'exécution des opérations. Nous proposons des conditions nécessaires et suffisantes de stabilité des solutions et les formules de calcul de leurs rayons de stabilité, respectivement. Nous montrons que le rayon de stabilité peut être vu comme une mesure de performance appropriée des solutions et peut jouer un rôle important lors de la conception en avant-projet de ce type de lignes. Quelques expérimentations d'une telle utilisation à l'aide d'une heuristique sont également présentées.

Le reste du chapitre est organisé de la manière suivante. Dans la Section 4.2, les définitions et propriétés de base de l'analyse de sensibilité sont introduites. Les sections 4.3, 4.4 et 4.5 sont dédiées au calcul du rayon de stabilité pour des solutions faisables, quasi-faisables et optimales, respectivement. Une procédure heuristique qui cherche un compromis entre les valeurs de la fonction objectif et du rayon de stabilité de la solution faisable est décrite dans la Section 4.6. Les résultats expérimentaux de l'utilisation de cette procédure sur des instances connues constituent la Section 4.7. Les remarques finales et les conclusions sont données dans la Section 4.8.

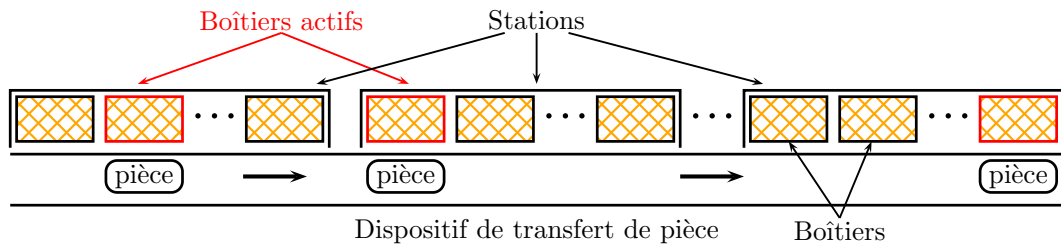


FIGURE 4.1 – Configuration d'une ligne d'usinage à boîtiers multibroches.

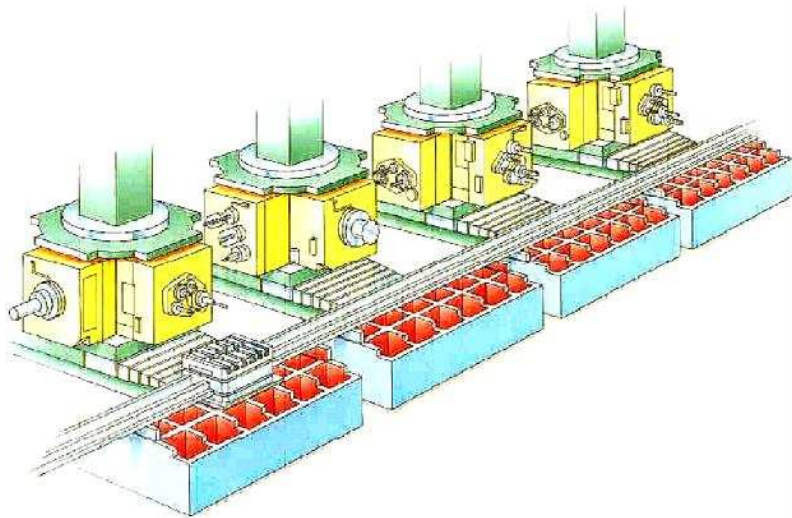


FIGURE 4.2 – Lignes d'usinage à boîtiers multibroches.

4.2 Définitions et propriétés de base

Les opérations nécessaires pour fabriquer la pièce constituent l'ensemble $V = \{1, 2, \dots, n\}$ associé au vecteur $t = (t_1, t_2, \dots, t_n) \in \mathbb{R}_+^n$, où t_j est le temps d'exécution de l'opération $j \in V$ et \mathbb{R}_+ est l'ensemble des nombres réels positifs.

Nous appelons bloc un ensemble d'opérations assignées à un boîtier multibroche et, par abus de langage, le boîtier lui-même.

Les restrictions suivantes doivent être prises en compte lors de l'affectation des opérations à des blocs et de ces blocs aux stations :

- Des *contraintes de capacité* sont caractérisées par le temps de cycle T_0 et le nombre maximal b_0 de blocs par station.
- Des *contraintes d'exclusion* définissent des groupes d'opérations qui ne peuvent pas être affectées ensemble à la même station (resp. au même bloc) à cause d'incompatibilités technologiques. Ces contraintes sont représentées par la famille E^w (resp.

E^b) de paires d'opérations de V telles que toutes les deux opérations de la même paire $e \in E^w$ ($e \in E^b$) ne peuvent pas être affectés à la même station (resp. au même bloc).

- Des *contraintes de précedence* introduisent une relation d'ordre partiel non-strict sur les opérations, et sont représentées par un graphe orienté acyclique $G = (V, \mathcal{A})$. Un arc (i, j) appartient à \mathcal{A} si et seulement si l'opération j doit être affectée à un bloc qui ne précède pas le bloc où l'opération i est affectée. Notons que cette relation partielle autorise l'affectation des opérations i et j au même bloc.

Définition 4.1. Une affectation des opérations V à des blocs et de ces blocs aux stations est appelée une solution faisable si toutes les contraintes définies ci-dessus sont respectées.

Dans la suite, l'ensemble des solutions faisables pour un vecteur $t \in \mathbb{R}_+^n$ est noté $\mathbf{S}_{\mathcal{F}}(t)$, où chaque solution s est caractérisée par la collection suivante :

$$\{\{V_{11}, V_{12}, \dots, V_{1|B_1|}\}, \dots, \{V_{|W^s|1}, V_{|W^s|2}, \dots, V_{|W^s|, |B_{|W^s|}|}\}\}$$

de sous-ensembles disjoints non-vides de V tels que $V = \bigcup_{k \in W^s} \bigcup_{l \in B_k} V_{kl}$. Ici nous utilisons les notations suivantes :

- W^s est l'ensemble des stations,
- B_k , $k \in W^s$, est l'ensemble des blocs de la station k ($|B_k| \leq b_0$),
- V_{kl} , $l \in B_k$, $k \in W^s$, est l'ensemble des opérations affectées au bloc l de la station k .

En utilisant les notations introduites, les contraintes de capacité peuvent être exprimées comme suit :

$$t_w(V_k) := \sum_{l \in B_k} t_b(V_{kl}) \leq T_0, \quad k \in W^s,$$

$$t_b(V_{kl}) := \max\{t_j : j \in V_{kl}\},$$

où $V_k = \bigcup_{l \in B_k} V_{kl}$ est l'ensemble des opérations affectées à la station k , $t_w(V_k)$ est le temps de travail de la station k , $t_b(V_{kl})$ est le temps de travail du bloc l de la station k .

Ainsi, le modèle générique de TLBP peut être formulé comme suit :

$$\mathcal{Z}(s, t) := C_1 |W^s| + C_2 \sum_{k \in W^s} |B_k| \rightarrow \min_{s \in \mathbf{S}_{\mathcal{F}}(t)},$$

où C_1 et C_2 sont respectivement les coûts d'une station et d'un bloc. Les solutions faisables avec la valeur minimum de $\mathcal{Z}(s, t)$ sont appelées solutions optimales et constituent l'ensemble $\mathbf{S}_{\mathcal{O}}(t)$. Il est évident que $\mathbf{S}_{\mathcal{O}}(t) \subseteq \mathbf{S}_{\mathcal{F}}(t)$.

Nous supposons de nouveau que l'ensemble V contient deux types d'opérations :

- Opérations incertaines : leur temps d'exécution peut varier au cours de la vie de la ligne. L'ensemble de ces opérations est noté \tilde{V} .
- Opérations déterministes : leur temps d'exécution reste le même au cours de la vie de la ligne. Ces opérations constituent l'ensemble $V \setminus \tilde{V}$.

Sans perte de généralité, nous supposons que $\tilde{V} = \{1, 2, \dots, \tilde{n}\}$ et $V \setminus \tilde{V} = \{\tilde{n} + 1, \tilde{n} + 2, \dots, n\}$, où $0 < \tilde{n} \leq n$.

Les variations des temps d'exécution des opérations (VTEO) de V peuvent être représentées par le vecteur $\xi = (\xi_1, \xi_2, \dots, \xi_{\tilde{n}}, 0, 0, \dots, 0) \in \mathbb{R}^n$, où tout ξ_j , $j \in \tilde{V}$, peut être aussi bien positif que négatif. Par conséquent, à un moment donné de la vie de la ligne le vecteur des temps d'exécution peut être représenté par le vecteur *perturbé* suivant $t^* = (t_1 + \xi_1, t_2 + \xi_2, \dots, t_{\tilde{n}} + \xi_{\tilde{n}}, t_{\tilde{n}+1}, \dots, t_n)$.

Remarque 4.1. Dans cette étude, il est supposé que $t_j^* = \max\{0, t_j + \xi_j\}$, $j \in \tilde{V}$.

Notons que les VTEO ne modifient ni les contraintes de précédence, ni les contraintes d'exclusion, ni le nombre de stations ou de blocs de la solution étudiée (optimale ou faisable). Cependant, elles peuvent provoquer la perte de l'optimalité et même de la faisabilité d'une solution. La faisabilité d'une solution peut être perdue si le temps de travail sur la station la plus chargée devient supérieur à T_0 pour un nouveau vecteur perturbé t^* . Une solution optimale s trouvée pour le vecteur initial t peut perdre son optimalité pour un certain nouveau vecteur perturbé t^* s'il existe une solution $s^0 \in \mathbf{S}_{\mathcal{F}}(t^*)$ telle que $\mathcal{Z}(s^0, t^*) < \mathcal{Z}(s, t^*)$.

Notons aussi que la solution s^0 respectant les contraintes de précédence et d'exclusion ayant une station $k \in W^{s^0}$ telle que $t_w(V_k) > T_0$ peut devenir faisable et même optimale si $t_w^*(V_k) \leq T_0$ est vérifié pour un nouveau vecteur perturbé t^* et tout $k \in W^{s^0}$. L'ensemble de ces solutions (appelées *quasi-faisables*) est noté $\mathbf{S}_{\tilde{\mathcal{F}}}(t)$.

Lors de l'étude de l'analyse de sensibilité des solutions du TLBP, la même définition du rayon de stabilité introduite dans la Section 3.2 est utilisée dans ce Chapitre.

Au vue de la continuité de $t(\cdot)$, la propriété suivante est facilement vérifiée. Elle sera utilisée par la suite.

Propriété 4.1. Pour toute solution s , la formule suivante se vérifie :

$$\forall \varepsilon > 0 \quad \forall t' \in \Omega(\varepsilon, t) \quad \forall k \in \tilde{W}^s \quad \forall l \in \tilde{B}_k \quad \left(t_b(\tilde{V}_{kl}) - \varepsilon < t'_b(\tilde{V}_{kl}) < t_b(\tilde{V}_{kl}) + \varepsilon \right).$$

Ici, $\tilde{W}^s = \{k \in W^s : \tilde{V}_k \neq \emptyset\}$, $\tilde{V}_k = V_k \cap \tilde{V}$, $\tilde{B}_k = \{l \in B_k : \tilde{V}_{kl} \neq \emptyset\}$, $\tilde{V}_{kl} = V_{kl} \cap \tilde{V}$, $t'_b(V_{kl}) = \max\{t'_j : j \in V_{kl}\}$.

Dans la section suivante, le comportement des solutions faisables sous VTEO est étudié.

4.3 Solutions faisables

Rappelons que la solution $s \in \mathbf{S}_{\mathcal{F}}(t)$ peut perdre sa faisabilité pour le nouveau vecteur perturbé $t^* \in \mathbb{R}_+^n$, c'est-à-dire $s \notin \mathbf{S}_{\mathcal{F}}(t^*)$, seulement si le temps d'exécution sur une station devient supérieur à T_0 , c'est-à-dire s'il existe $k \in \widetilde{W}^s$ tel que $t_w^*(V_k) > T_0$.

Le temps d'un bloc est égal au temps de son opération la plus longue (voir la Figure 4.3).

Pour tout $s \in \mathbf{S}_{\mathcal{F}}(t)$ et $k \in \widetilde{W}^s$, les notations suivantes sont introduites :

- $\Delta_{kl} = t_b(V_{kl}) - t_b(\widetilde{V}_{kl})$, $l \in \widetilde{B}_k$, est la différence entre le temps du bloc l et le temps d'exécution maximal parmi ses opérations incertaines,
- $\widetilde{B}_{k0} = \{l \in \widetilde{B}_k : \Delta_{kl} = 0\}$ est l'ensemble des blocs pour lesquels le temps d'exécution maximal parmi les opérations incertaines est égal au temps du bloc correspondant,
- $\widetilde{B}_{k>} = \{l \in \widetilde{B}_k : \Delta_{kl} > 0\}$ est l'ensemble des blocs pour lesquels le temps d'exécution maximal parmi les opérations incertaines est plus petit que le temps du bloc correspondant.

Il est facile de voir que $\widetilde{B}_k = \widetilde{B}_{k0} \cup \widetilde{B}_{k>} \neq \emptyset$, $\widetilde{B}_{k0} \cap \widetilde{B}_{k>} = \emptyset$.

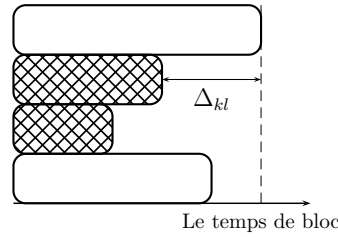


FIGURE 4.3 – Le bloc l de la station k de la solution s : Δ_{kl} .

Un exemple illustratif est donné dans la Figure 4.3. Ici un bloc avec quatre opérations est montré. Chaque opération est représentée par un rectangle horizontal dont la longueur correspond à son temps d'exécution. Les opérations incertaines sont hachurées.

Lemme 4.1. *Si pour $s \in \mathbf{S}_{\mathcal{F}}(t)$ il existe $k \in \widetilde{W}^s$ tel que $\widetilde{B}_{k>} \neq \emptyset$, alors*

$$\forall l \in \widetilde{B}_{k>} \quad \forall t' \in \Omega(\Delta_{kl}, t) \quad (t'_b(V_{kl}) = t_b(V_{kl})), \quad (4.1)$$

$$\forall l \in \widetilde{B}_{k>} \quad \forall \varepsilon \geq \Delta_{kl} \quad \forall t' \in \Omega(\varepsilon, t) \quad (t'_b(V_{kl}) \leq t_b(V_{kl}) + \varepsilon - \Delta_{kl}). \quad (4.2)$$

Démonstration. Par définition,

$$t'_b(V_{kl}) = \max\{t'_b(\widetilde{V}_{kl}), t_b(V_{kl})\}.$$

En tenant compte de la Propriété 4.1, nous avons :

$$t'_b(\widetilde{V}_{kl}) < t_b(\widetilde{V}_{kl}) + \Delta_{kl} = t_b(V_{kl}), \quad t' \in \Omega(\Delta_{kl}, t),$$

$$t'_b(\tilde{V}_{kl}) < t_b(\tilde{V}_{kl}) + \varepsilon = t_b(V_{kl}) + \varepsilon - \Delta_{kl}, \quad t' \in \Omega(\varepsilon, t),$$

d'où la validité des formules (4.1) et (4.2). \square

Le théorème suivant montre que le calcul du \mathcal{F} -rayon de stabilité peut être réduit à un problème d'optimisation biniveau. Nous utilisons la notation suivante :

$$\mathcal{I}_n = \begin{cases} \{1, \dots, n\}, & \text{si } n \in \mathbb{N}, \\ \emptyset, & \text{si } n = 0. \end{cases}$$

Théorème 4.1. *Le \mathcal{F} -rayon de stabilité de $s \in \mathbf{S}_{\mathcal{F}}(t)$ est calculé comme suit :*

$$\rho_{\mathcal{F}}(s, t) = \min_{k \in \tilde{W}^s} \varphi(k), \quad (4.3)$$

où

$$\begin{aligned} \varphi(k) &= \Delta_{kl_{j_k^*}} + \frac{T_0 - t_w(V_k) - \Theta(j_k^*)}{|\tilde{B}_{k0}| + j_k^*}, \\ j_k^* &= \operatorname{argmax}\{\Theta(j) : \Theta(j) \leq T_0 - t_w(V_k), j \in \{0\} \cup \mathcal{I}_{|\tilde{B}_{k>}|}\}, \\ \Theta(j) &= (|\tilde{B}_{k0}| + j)\Delta_{kl_j} - \sum_{i \in \mathcal{I}_j} \Delta_{kl_i}, \\ 0 &= \Delta_{kl_0} \leq \dots \leq \Delta_{kl_{|\tilde{B}_{k>}|}}, \quad l_i \in \tilde{B}_{k>}, \quad i \in \mathcal{I}_{|\tilde{B}_{k>}|}. \end{aligned}$$

Démonstration. Pour simplifier la démonstration, les notations ρ et φ utilisées désignent les parties gauche et droite de l'égalité (4.3), respectivement. Notons que φ est un nombre fini grâce au fait que $s \in \mathbf{S}_{\mathcal{F}}(t)$ et à la définition de j_k^* , qui implique aussi les inégalités suivantes :

$$|\tilde{B}_{k0}| + j_k^* > 0, \quad k \in \tilde{W}^s. \quad (4.4)$$

Notons que la fonction non-décroissante $\Theta(\cdot)$ a une représentation récursive :

$$\Theta(j+1) = \Theta(j) + (|\tilde{B}_{k0}| + j)(\Delta_{kl_{j+1}} - \Delta_{kl_j}), \quad \Theta(0) = 0, \quad (4.5)$$

qui sera utilisée ultérieurement.

Afin de prouver la formule (4.3), nous montrons successivement que les inégalités $\rho \geq \varphi$ et $\rho \leq \varphi$ sont vérifiées.

Tout d'abord, nous prouvons que $\rho \geq \varphi$. Pour cela, il suffit de vérifier que

$$\forall t' \in \Omega(\varphi, t) \quad (s \in \mathbf{S}_{\mathcal{F}}(t')). \quad (4.6)$$

Si $\varphi = 0$, l'inégalité $\rho \geq \varphi$ est immédiate. Alors, nous supposons par la suite que $\varphi > 0$.

En utilisant les définitions de $\varphi(k)$ et j_k^* , ainsi que la représentation (4.5) de $\Theta(\cdot)$, il est facile de voir que pour tout $k \in \widetilde{W}^s$, nous avons :

$$\forall i \in \mathcal{I}_{j_k^*} \quad (\Delta_{kl_i} \leq \varphi(k)), \quad (4.7)$$

$$\forall i \in \mathcal{I}_{|\widetilde{B}_{k>}|} \setminus \mathcal{I}_{j_k^*} \quad (\varphi(k) < \Delta_{kl_i}). \quad (4.8)$$

D'où, vu la Propriété 4.1, le Lemme 4.1, et l'expression (4.4), pour tout $t' \in \Omega(\varphi(k), t)$, $k \in \widetilde{W}^s$, nous dérivons :

$$\begin{aligned} t'_w(V_k) &= \sum_{l \in B_k} t'_b(V_{kl}) = \sum_{l \in B_k \setminus \widetilde{B}_k} t_b(V_{kl}) + \sum_{l \in \widetilde{B}_{k0}} t'_b(V_{kl}) + \sum_{l \in \widetilde{B}_{k>}} t'_b(V_{kl}) \leq \\ &\sum_{l \in B_k \setminus \widetilde{B}_k} t_b(V_{kl}) + \sum_{l \in \widetilde{B}_{k0}} (t_b(V_{kl}) + \varphi(k)) + \sum_{i \in \mathcal{I}_{j_k^*}} (t_b(V_{kl_i}) + \varphi(k) - \Delta_{kl_i}) + \\ &\sum_{i \in \mathcal{I}_{|\widetilde{B}_{k>}|} \setminus \mathcal{I}_{j_k^*}} t_b(V_{kl_i}) = t_w(V_k) + \varphi(k)(|\widetilde{B}_{k0}| + j_k^*) - \sum_{i \in \mathcal{I}_{j_k^*}} \Delta_{kl_i} = T_0. \end{aligned}$$

Par conséquent, la formule suivante est prouvée :

$$\forall k \in \widetilde{W}^s \quad \forall t' \in \Omega(\varphi(k), t) \quad (t'_w(V_k) \leq T_0).$$

En se basant sur ce résultat et en prenant en compte les inégalités suivantes :

$$\forall \varepsilon > 0 \quad \forall k \in W^s \setminus \widetilde{W}^s \quad \forall t' \in \Omega(\varepsilon, t) \quad (t'_w(V_k) = t_w(V_k) \leq T_0),$$

nous concluons que

$$\forall k \in W^s \quad \forall t' \in \Omega(\varphi, t) \quad (t'_w(V_k) \leq T_0)$$

ce qui implique (4.6).

Maintenant, montrons que $\rho \leq \varphi$, ce qui équivaut à prouver que

$$\forall \varepsilon > \varphi \quad \exists t^* \in \Omega(\varepsilon, t) \quad (s \notin \mathbf{S}_{\mathcal{F}}(t^*)). \quad (4.9)$$

Pour prouver la formule (4.9), la définition de φ est utilisée. Par définition, il existe $k^* \in \widetilde{W}^s$ tel que $\varphi = \varphi(k^*)$. Alors, en supposant que $\varepsilon > \varphi$, $t^* \in \Omega(\varepsilon, t)$, où

$$t_j^* = \begin{cases} t_j + \delta, & \text{si } j \in \widetilde{V}_{k^*}, \\ t_j & \text{sinon,} \end{cases}$$

$$\varphi < \delta < \min\{\varepsilon, \xi(k^*)\},$$

$$\xi(k^*) = \min_{i \in \mathcal{I}_{|\widetilde{B}_{k^*>}|} \setminus \mathcal{I}_{j_{k^*}^*}} \Delta_{k^*l_i},$$

et en considérant les formules (4.4), (4.7) et (4.8), nous obtenons :

$$\begin{aligned}
 t_w^*(V_{k^*}) &= \sum_{l \in B_{k^*}} t_b^*(V_{k^*l}) = \sum_{l \in B_{k^*} \setminus \tilde{B}_{k^*}} t_b(V_{k^*l}) + \sum_{l \in \tilde{B}_{k^*0}} t_b^*(V_{k^*l}) + \sum_{l \in \tilde{B}_{k^*>}} t_b^*(V_{k^*l}) = \\
 &\sum_{l \in B_{k^*} \setminus \tilde{B}_{k^*}} t_b(V_{k^*l}) + \sum_{l \in \tilde{B}_{k^*0}} (t_b(V_{k^*l}) + \delta) + \sum_{i \in \mathcal{I}_{j_{k^*}^*}} (t_b(V_{k^*l_i}) + \delta - \Delta_{k^*l_i}) + \\
 &\sum_{i \in \mathcal{I}_{|\tilde{B}_{k^*>}|} \setminus \mathcal{I}_{j_{k^*}^*}} t_b(V_{k^*l_i}) > t_w(V_{k^*}) + \varphi(k^*)(|\tilde{B}_{k^*0}| + j_{k^*}^*) - \sum_{i \in \mathcal{I}_{j_{k^*}^*}} \Delta_{k^*l_i} = T_0
 \end{aligned}$$

Il s'en suit que $s \notin \mathbf{S}_{\mathcal{F}}(t^*)$, et, par conséquent, (4.9) est vérifiée. \square

Le Théorème 4.1 implique les conséquences suivantes.

Conséquence 4.1. *La solution $s \in \mathbf{S}_{\mathcal{F}}(t)$ n'est pas \mathcal{F} -stable ssi la formule suivante est vérifiée :*

$$\exists k \in \tilde{W}^s \quad (t_w(V_k) = T_0 \quad \& \quad \tilde{B}_{k0} \neq \emptyset). \quad (4.10)$$

Démonstration. Suffisance. Soit $k \in \tilde{W}^s$ une station qui satisfait la condition (4.10). Alors, en utilisant la définition de j_k^* , nous obtenons $j_k^* = 0$ et, par conséquent, $\varphi(k) = 0$. La dernière égalité, conformément à (4.3), implique que $\rho_{\mathcal{F}}(s, t) = 0$ et, s n'est donc pas \mathcal{F} -stable.

Nécessité. Supposons, au contraire, que s n'est pas \mathcal{F} -stable, tandis que

$$\forall k \in \tilde{W}^s \quad (t_w(V_k) < T_0 \vee \tilde{B}_{k0} = \emptyset) \quad (4.11)$$

est vérifiée. Considérons deux cas possibles pour un $k \in \tilde{W}^s$ arbitraire.

Cas 1. $t_w(V_k) = T_0$. Alors, d'après (4.11), $\tilde{B}_{k0} = \emptyset$ est vérifié. Cela implique $j_k^* > 0$ du fait de (4.4). D'où, $\varphi(k) > 0$.

Cas 2. $t_w(V_k) < T_0$. Dans ce cas, en utilisant (4.4), nous concluons encore une fois que $\varphi(k) > 0$.

Ainsi, en résumant ces deux cas, nous avons $\rho_{\mathcal{F}}(s, t) > 0$ du fait de (4.3). Cela implique que s est \mathcal{F} -stable, ce qui contredit la supposition. \square

Conséquence 4.2. *Le \mathcal{F} -rayon de stabilité de la solution $s \in \mathbf{S}_{\mathcal{F}}(t)$ peut être calculé en $\mathcal{O}(n + \sum_{k \in \tilde{W}^s} |\tilde{B}_{k>}| \log |\tilde{B}_{k>}|)$.*

Démonstration. Un algorithme de calcul de $\rho_{\mathcal{F}}(s, t)$ consiste à analyser de manière séquentielle les stations de s . Ainsi, pour chaque bloc l de la station courante $k \in W^s$, les caractéristiques suivantes sont calculées : le temps du bloc $t_b(V_{kl})$, et en plus Δ_{kl} , si $l \in \tilde{B}_k$ (cela se calcule en $\mathcal{O}(|V_{kl}|)$). Dans le cas où tous les blocs de la station courante k

sont examinés et aucune opération incertaine n'est trouvée, la station suivante est analysée, sinon $\varphi(k)$ est calculé. Pour calculer $\varphi(k)$, les valeurs Δ_{kl} , $l \in \widetilde{B}_{k>}$, sont triées dans l'ordre non-décroissant (cela se calcule en $\mathcal{O}(|\widetilde{B}_{k>}| \log |\widetilde{B}_{k>}|)$) et l'indice j_k^* se trouve au plus fort en $\mathcal{O}(|\widetilde{B}_{k>}|)$, du fait de la représentation (4.5) de $\Theta(\cdot)$. Cette séquence de calculs est valide seulement dans le cas où $\widetilde{B}_{k>} \neq \emptyset$, sinon $\varphi(k) = \frac{T_0 - t_w(V_k)}{|\widetilde{B}_{k0}|}$. Cela continue jusqu'à ce que toutes les stations soient analysées ou que'une station $k \in \widetilde{W}^s$, telle que $\varphi(k) = 0$, soit trouvée. Dans le dernier cas, $\rho_{\mathcal{F}}(s, t) = 0$, sinon $\rho_{\mathcal{F}}(s, t) = \min_{k \in \widetilde{W}^s} \varphi(k)$. Ainsi, dans le pire des cas, où aucun $\varphi(k)$, $k \in \widetilde{W}^s$, n'est égal à 0, et aucun $\widetilde{B}_{k>}$, $k \in \widetilde{W}^s$, n'est vide, cet algorithme se réalise en $\mathcal{O}(\sum_{k \in \widetilde{W}^s} \sum_{l \in B_k} |V_{kl}| + \sum_{k \in \widetilde{W}^s} (|\widetilde{B}_{k>}| \log |\widetilde{B}_{k>}| + |\widetilde{B}_{k>}|)) = \mathcal{O}(n + \sum_{k \in \widetilde{W}^s} |\widetilde{B}_{k>}| \log |\widetilde{B}_{k>}|)$. \square

En tenant compte des Conséquences 4.1 et 4.2, ainsi que de la représentation (4.5), le schéma général du calcul du \mathcal{F} -rayon de stabilité est représenté par l'Algorithme 3, où la variable binaire *flag* est utilisée afin de contrôler si une nouvelle itération est nécessaire ou non.

Il est facile de voir que si les conditions suivantes sont vérifiées :

- une opération par bloc est autorisée,
- le nombre de blocs par station n'est pas limité,
- il n'existe aucune contrainte d'exclusion,
- $C_2 = 0$,

alors le TLBP étudié se réduit au SALBP-1. Dans ce cas, $\widetilde{B}_{k>}$ n'est pas vide, et par conséquent, $j_k^* = 0$ pour tout $k \in \widetilde{W}^s$. De plus, $\widetilde{B}_{k0} = \widetilde{V}_k$. Ainsi, le Théorème 4.1 et la Conséquence 4.1 nous amènent aux propositions suivantes qui ont déjà été démontrées dans [Sotkov *et al.*, 2006].

Conséquence 4.3. [Sotkov *et al.*, 2006] *Le \mathcal{F} -rayon de stabilité de la solution $s \in \mathbf{S}_{\mathcal{F}}(t)$ pour SALBP-1 est calculé comme suit :*

$$\rho_{\mathcal{F}}(s, t) = \min_{k \in \widetilde{W}^s} \frac{T_0 - t_w(V_k)}{|\widetilde{V}_k|}.$$

Conséquence 4.4. [Sotkov *et al.*, 2006] *La solution $s \in \mathbf{S}_{\mathcal{F}}(t)$ de SALBP-1 n'est pas \mathcal{F} -stable ssi il existe $k \in \widetilde{W}$ tel que $T_0 = t_w(V_k)$.*

La section suivante porte sur l'évaluation du $\widehat{\mathcal{F}}$ -rayon de stabilité pour les solutions quasi-faisables.

4.4 Solutions quasi-faisables

Rappelons que pour toute solution $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$, il existe une station $k \in W^s$ telle que $t_w(V_k) > T_0$. Néanmoins, une solution quasi-faisable peut devenir faisable, si pour un

Algorithme 3 : Le calcul du \mathcal{F} -rayon de stabilité

```

1  flag ← vrai /* Tant que s est  $\mathcal{F}$ -stable                               */
2  pour chaque  $k \in \widetilde{W}^s$  faire
3  | si  $t_w(V_k) = T_0$  et  $\widetilde{B}_{k0} \neq \emptyset$  alors
4  | |  $\rho_{\mathcal{F}}(s, t) \leftarrow 0$ , flag ← faux /* s n'est pas  $\mathcal{F}$ -stable       */
5  | | quitter la boucle
6  | sinon si  $\widetilde{B}_{k>} = \emptyset$  alors
7  | |  $\varphi(k) \leftarrow \frac{T_0 - t_w(V_k)}{|\widetilde{B}_{k0}|}$ 
8  | sinon
9  | | Trier  $\Delta_{kl}$ ,  $l \in \widetilde{B}_{k>} : 0 = \Delta_{kl_0} < \Delta_{kl_1} \leq \dots \leq \Delta_{kl_{|\widetilde{B}_{k>}|}}$ 
10 | |  $j \leftarrow 0$ ,  $\varphi(k) \leftarrow 0$ ,  $\Theta \leftarrow 0$ 
11 | | tant que  $j < |\widetilde{B}_{k>}|$  faire
12 | | | si  $\Theta + (|\widetilde{B}_{k0}| + j)(\Delta_{kl_{j+1}} - \Delta_{kl_j}) \leq T_0 - t_w(V_k)$  alors
13 | | | |  $\Theta \leftarrow \Theta + (|\widetilde{B}_{k0}| + j)(\Delta_{kl_{j+1}} - \Delta_{kl_j})$ ,  $\varphi(k) \leftarrow \Delta_{kl_{j+1}}$ ,  $j \leftarrow j + 1$ 
14 | | | sinon
15 | | | | quitter la boucle
16 | | |  $\varphi(k) \leftarrow \varphi(k) + \frac{T_0 - t_w(V_k) - \Theta}{|\widetilde{B}_{k0}| + j}$ 
17 si flag = vrai alors
18 |  $\rho_{\mathcal{F}}(s, t) \leftarrow \min\{\varphi(k) : k \in \widetilde{W}^s\}$ 

```

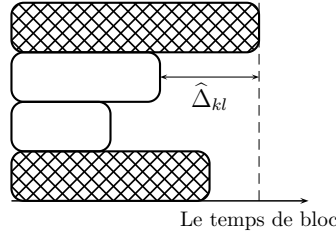
nouveau vecteur perturbé $t^* \in \mathbb{R}_+^n$, $t_w^*(V_k) \leq T_0$ est vérifié, pour tout $k \in W^s$. Cependant, si pour une solution quasi-faisable il existe une station sans aucune opération incertaine dont le temps d'exécution est supérieur à T_0 , alors cette solution restera toujours quasi-faisable.

Dans ce qui suit, l'ensemble des stations de la solution s dont le temps d'exécution est supérieur à T_0 est noté \widehat{W}^s .

Pour tout $s \in \mathbf{S}_{\mathcal{F}}(t)$ et tout $k \in \widehat{W}^s$, les notations suivantes sont introduites :

- $\widehat{\Delta}_{kl} = t_b(\widetilde{V}_{kl}) - t_b(V_{kl} \setminus \widetilde{V}_{kl})$, $l \in \widetilde{B}_k$, est la différence entre le temps d'exécution maximal parmi les opérations incertaines et le temps maximal parmi les opérations déterministes affectées au bloc l ,
- $\widehat{B}_k = \{l \in \widetilde{B}_k : \widehat{\Delta}_{kl} > 0\}$ est l'ensemble des blocs où le temps d'exécution maximal parmi les opérations incertaines est plus grand que le temps d'exécution maximal parmi les opérations déterministes.

Un exemple illustratif est donné dans la Figure 4.4. Ici, un bloc avec quatre opérations est montré. Chaque opération est représentée par un rectangle horizontal dont la longueur


 FIGURE 4.4 – Le bloc l de la station k de la solution s : $\widehat{\Delta}_{kl}$.

correspond à son temps d'exécution. Les opérations incertaines sont hachurées.

Dans ce qui suit, la propriété et le lemme suivants sont utilisés.

Propriété 4.2. Si, pour $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$ et $k \in \widehat{W}^s$, l'inégalité $\widetilde{B}_k \setminus \widehat{B}_k \neq \emptyset$ est vérifiée, alors

$$\forall \varepsilon > 0 \quad \forall t' \in \Omega(\varepsilon, t) \quad \forall l \in \widetilde{B}_k \setminus \widehat{B}_k \quad (t'_b(V_{kl}) \geq t_b(V_{kl})).$$

Lemme 4.2. Si, pour $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$, il existe $k \in \widehat{W}^s$ tel que $\widehat{B}_k \neq \emptyset$, alors

$$\forall l \in \widehat{B}_k \quad \forall \varepsilon > 0 \quad \forall t' \in \Omega(\varepsilon, t) \quad (t'_b(V_{kl}) \geq t_b(V_{kl}) - \widehat{\Delta}_{kl}), \quad (4.12)$$

$$\forall l \in \widehat{B}_k \quad \forall \varepsilon < \widehat{\Delta}_{kl} \quad \forall t' \in \Omega(\varepsilon, t) \quad (t'_b(V_{kl}) > t_b(V_{kl}) - \varepsilon). \quad (4.13)$$

Démonstration. Par définition,

$$t'_b(V_{kl}) = \max\{t'_b(\widetilde{V}_{kl}), t_b(V_{kl} \setminus \widetilde{V}_{kl})\}.$$

En prenant en compte les égalités évidentes

$$t_b(\widetilde{V}_{kl}) = t_b(V_{kl}), \quad l \in \widehat{B}_k,$$

et la Propriété 4.1, nous obtenons :

$$t'_b(V_{kl}) \geq t_b(V_{kl} \setminus \widetilde{V}_{kl}) = t_b(\widetilde{V}_{kl}) - \widehat{\Delta}_{kl} = t_b(V_{kl}) - \widehat{\Delta}_{kl}, \quad t' \in \Omega(\varepsilon, t),$$

$$t'_b(\widetilde{V}_{kl}) > t_b(\widetilde{V}_{kl}) - \varepsilon > t_b(\widetilde{V}_{kl}) - \widehat{\Delta}_{kl} = t_b(V_{kl} \setminus \widetilde{V}_{kl}), \quad t' \in \Omega(\varepsilon, t),$$

ce qui implique la validité des formules (4.12) et (4.13). \square

Théorème 4.2. Le $\widehat{\mathcal{F}}$ -rayon de stabilité de $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$ est calculé comme suit :

$$\rho_{\widehat{\mathcal{F}}}(s, t) = \max_{k \in \widehat{W}^s} \widehat{\varphi}(k), \quad (4.14)$$

où

$$\widehat{\varphi}(k) = \widehat{\Delta}_{kl_{j_k^*}} + \frac{t_w(V_k) - T_0 - \widehat{\Theta}(j_k^*)}{|\widehat{B}_k| - j_k^*},$$

$$\begin{aligned}
 j_k^* &= \operatorname{argmax}\{\widehat{\Theta}(j) : \widehat{\Theta}(j) \leq t_w(V_k) - T_0, j \in \{0\} \cup \mathcal{I}_{|\widehat{B}_k|}\}, \\
 \widehat{\Theta}(j) &= (|\widehat{B}_k| - j)\widehat{\Delta}_{kl_j} + \sum_{i \in \mathcal{I}_j} \widehat{\Delta}_{kl_i}, \\
 0 &= \widehat{\Delta}_{kl_0} \leq \dots \leq \widehat{\Delta}_{kl_{|\widehat{B}_k|}}, \quad l_i \in \widehat{B}_k, \quad i \in \mathcal{I}_{|\widehat{B}_k|}.
 \end{aligned}$$

Démonstration. Comme pour le Théorème 4.1, les notations supplémentaires suivantes sont introduites : $\widehat{\rho}$ et $\widehat{\varphi}$ sont respectivement les parties gauche et droite de (4.14). Il est facile de voir que $\widehat{\varphi}$ peut être égal à $+\infty$. Ceci arrive si $|\widehat{B}_k| = j_k^*$ pour un indice $k \in \widehat{W}^s$ et signifie que la solution s ne deviendra jamais faisable. Dans le cas contraire, $\widehat{\varphi}$ est un nombre positif fini du fait de $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$ et de la définition de j_k^* . Alors, par la suite, nous supposons que

$$|\widehat{B}_k| - j_k^* > 0, \quad k \in \widehat{W}^s. \quad (4.15)$$

Notons que la fonction non-décroissante $\widehat{\Theta}(\cdot)$ a une représentation récursive :

$$\widehat{\Theta}(j+1) = \widehat{\Theta}(j) + (|\widehat{B}_k| - j)(\widehat{\Delta}_{kl_{j+1}} - \widehat{\Delta}_{kl_j}), \quad \widehat{\Theta}(0) = 0, \quad (4.16)$$

qui sera utilisée par la suite.

Pour prouver la formule (4.14), nous montrons successivement que les inégalités $\widehat{\rho} \geq \widehat{\varphi}$ et $\widehat{\rho} \leq \widehat{\varphi}$ sont vérifiées.

Tout d'abord, prouvons que $\widehat{\rho} \geq \widehat{\varphi}$. Pour cela, il est suffisant de vérifier que

$$\forall t' \in \Omega(\widehat{\varphi}, t) \quad (s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t')). \quad (4.17)$$

Si $\widehat{\varphi} = 0$, l'inégalité $\widehat{\rho} \geq \widehat{\varphi}$ est immédiate. Alors, par la suite nous supposons que $\widehat{\varphi} > 0$.

En utilisant les définitions de $\widehat{\varphi}(k)$ et j_k^* , ainsi que la représentation (4.16) de $\widehat{\Theta}(\cdot)$, il est facile de voir que pour tout $k \in \widehat{W}^s$, nous avons :

$$\forall i \in \mathcal{I}_{j_k^*} \quad (\widehat{\Delta}_{kl_i} \leq \widehat{\varphi}(k)), \quad (4.18)$$

$$\forall i \in \mathcal{I}_{|\widehat{B}_k|} \setminus \mathcal{I}_{j_k^*} \quad (\widehat{\varphi}(k) < \widehat{\Delta}_{kl_i}), \quad (4.19)$$

et il existe $k^* \in \widehat{W}^s$, tel que $\widehat{\varphi}(k^*) = \widehat{\varphi}$.

D'où, vu la Propriété 4.2, le Lemme 4.2, et l'expression (4.15), pour tout $t' \in \Omega(\widehat{\varphi}, t)$, nous dérivons :

$$\begin{aligned}
 t'_w(V_{k^*}) &= \sum_{l \in B_{k^*}} t'_b(V_{k^*l}) = \sum_{l \in B_{k^*} \setminus \widetilde{B}_{k^*}} t_b(V_{k^*l}) + \sum_{l \in \widetilde{B}_{k^*} \setminus \widehat{B}_{k^*}} t'_b(V_{k^*l}) + \sum_{l \in \widehat{B}_{k^*}} t'_b(V_{k^*l}) > \\
 &\sum_{l \in B_{k^*} \setminus \widetilde{B}_{k^*}} t_b(V_{k^*l}) + \sum_{l \in \widetilde{B}_{k^*} \setminus \widehat{B}_{k^*}} t_b(V_{k^*l}) + \sum_{i \in \mathcal{I}_{j_{k^*}^*}} (t_b(V_{k^*l_i}) - \widehat{\Delta}_{k^*l_i}) +
 \end{aligned}$$

$$\sum_{i \in \mathcal{I}_{|\widehat{B}_{k^*}|} \setminus \mathcal{I}_{j_{k^*}^*}} (t_b(V_{k^*l_i}) - \widehat{\varphi}(k^*)) = t_w(V_{k^*}) - \sum_{i \in \mathcal{I}_{j_{k^*}^*}} \widehat{\Delta}_{k^*l_i} - \widehat{\varphi}(k^*)(|\widehat{B}_{k^*}| - j_{k^*}^*) = T_0.$$

Donc, la formule suivante est prouvée :

$$\exists k^* \in \widehat{W}^s \quad \forall t' \in \Omega(\widehat{\varphi}, t) \quad (t'_w(V_{k^*}) > T_0),$$

ce qui implique (4.17).

Maintenant montrons que $\widehat{\rho} \leq \widehat{\varphi}$, ce qui équivaut à prouver la formule suivante :

$$\forall \varepsilon > \widehat{\varphi} \quad \exists t^* \in \Omega(\varepsilon, t) \quad (s \in \mathbf{S}_{\mathcal{F}}(t^*)). \quad (4.20)$$

Par définition de $\widehat{\varphi}$, pour tout $k \in \widehat{W}^s$, nous avons $\widehat{\varphi} \geq \widehat{\varphi}(k)$. Alors, en supposant que $\varepsilon > \widehat{\varphi}$, $t^* \in \Omega(\varepsilon, t)$, où

$$t_j^* = \begin{cases} t_j - \delta, & \text{si } j \in \widetilde{V}_{kl}, l \in \widehat{B}_k, k \in \widehat{W}^s, \\ t_j & \text{sinon,} \end{cases}$$

$$\widehat{\varphi} < \delta < \min\{\varepsilon, \xi\},$$

$$\xi = \min_{k \in \widehat{W}^s} \min_{i \in \mathcal{I}_{|\widehat{B}_k|} \setminus \mathcal{I}_{j_k^*}} \widehat{\Delta}_{kl_i},$$

et en tenant compte de (4.15), (4.18) et (4.19), nous obtenons :

$$\begin{aligned} t_w^*(V_k) &= \sum_{l \in B_k} t_b^*(V_{kl}) = \sum_{l \in B_k \setminus \widetilde{B}_k} t_b(V_{kl}) + \sum_{l \in \widetilde{B}_k \setminus \widehat{B}_k} t_b(V_{kl}) + \sum_{l \in \widehat{B}_k} t_b^*(V_{kl}) = \\ &\sum_{l \in B_k \setminus \widetilde{B}_k} t_b(V_{kl}) + \sum_{l \in \widetilde{B}_k \setminus \widehat{B}_k} t_b(V_{kl}) + \sum_{i \in \mathcal{I}_{j_k^*}} (t_b(V_{kl_i}) - \widehat{\Delta}_{kl_i}) + \\ &\sum_{i \in \mathcal{I}_{|\widehat{B}_k|} \setminus \mathcal{I}_{j_k^*}} (t_b(V_{kl_i}) - \delta) > t_w(V_k) - \widehat{\varphi}(k)(|\widehat{B}_k| - j_k^*) - \sum_{i \in \mathcal{I}_{j_k^*}} \widehat{\Delta}_{kl_i} = T_0, \quad k \in \widehat{W}^s. \end{aligned}$$

Ensuite, compte tenu les inégalités évidentes suivantes $t_w^*(V_k) \leq T_0$, $k \in W^s \setminus \widehat{W}^s$, nous concluons que

$$\forall \varepsilon > \widehat{\varphi} \quad \exists t^* \in \Omega(\varepsilon, t) \quad \forall k \in W^s \quad (t_w^*(V_k) \leq T_0).$$

Ainsi, la formule (4.20) est vérifiée. \square

Le Théorème 4.2 implique les conséquences suivantes.

Conséquence 4.5. *Chaque solution quasi-faisable est $\widehat{\mathcal{F}}$ -stable.*

Démonstration. Comme pour tout $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$, les inégalités $t_w(V_k) > T_0$, $k \in \widehat{W}^s$, sont vérifiées, alors d'après la définition de j_k^* , l'égalité $\widehat{\Theta}(j_k^*) = t_w^*(V_k) - T_0$ est possible seulement pour $j_k^* > 0$. Alors, en tenant compte de la définition de $\widehat{\varphi}(k)$, nous avons $\widehat{\varphi}(k) > 0$, $k \in \widehat{W}^s$, ce qui implique $\rho_{\widehat{\mathcal{F}}}(s, t) > 0$ du fait de (4.14). \square

Conséquence 4.6. *Le $\widehat{\mathcal{F}}$ -rayon de stabilité de la solution $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$ peut être calculé en $\mathcal{O}(n + \sum_{k \in \widehat{W}^s} |\widehat{B}_k| \log |\widehat{B}_k|)$.*

Démonstration. Une approche similaire à celle proposée pour la démonstration de la Conséquence 4.2 est utilisée. Un algorithme de recherche de $\rho_{\widehat{\mathcal{F}}}(s, t)$ consiste à analyser de manière séquentielle les stations de la solution s . Ainsi, pour chaque bloc l de la station courante $k \in W^s$, les caractéristiques suivantes sont calculées : le temps de travail du bloc $t_b(V_{kl})$ et en plus $\widehat{\Delta}_{kl}$, si $l \in \widetilde{B}_k$ (cela se calcule en $\mathcal{O}(|V_{kl}|)$). Quand tous les blocs de la station courante k sont examinés et son temps de travail ne dépasse pas T_0 , la station suivante est analysée, sinon $\widehat{\varphi}(k)$ est calculé. Pour calculer $\widehat{\varphi}(k)$, les valeurs $\widehat{\Delta}_{kl}$, $l \in \widehat{B}_k$, sont triées dans l'ordre non-décroissant (cela se calcule en $\mathcal{O}(|\widehat{B}_k| \log |\widehat{B}_k|)$), et l'indice j_k^* se trouve dans le pire des cas en $\mathcal{O}(|\widehat{B}_k|)$ du fait de la représentation (4.16) de $\widehat{\Theta}(\cdot)$. Cette séquence de calculs est valide seulement dans le cas où $\widehat{B}_k \neq \emptyset$, sinon $\widehat{\varphi}(k) = +\infty$. L'algorithme continue jusqu'à ce que toutes les stations soient analysées ou qu'une station $k \in \widehat{W}^s$ tel que $\widehat{\varphi}(k) = +\infty$ soit trouvée. Dans ce dernier cas, $\rho_{\widehat{\mathcal{F}}}(s, t) = +\infty$, sinon $\rho_{\widehat{\mathcal{F}}}(s, t) = \max_{k \in \widehat{W}^s} \widehat{\varphi}(k)$. Donc, dans le pire des cas lorsqu'aucun $\widehat{\varphi}(k)$, $k \in \widehat{W}^s$ n'est égal à $+\infty$ et aucun \widehat{B}_k , $k \in \widehat{W}^s$ n'est vide, cet algorithme se termine en $\mathcal{O}(\sum_{k \in W^s} \sum_{l \in B_k} |V_{kl}| + \sum_{k \in \widehat{W}^s} (|\widehat{B}_k| \log |\widehat{B}_k| + |\widehat{B}_k|)) = \mathcal{O}(n + \sum_{k \in \widehat{W}^s} |\widehat{B}_k| \log |\widehat{B}_k|)$. \square

En s'appuyant sur la Conséquence 4.6 et la représentation (4.16), le schéma général de calcul du $\widehat{\mathcal{F}}$ -rayon de stabilité est donné par l'Algorithme 4, où la variable binaire *flag* est utilisée pour contrôler si une nouvelle itération est nécessaire ou non.

Comme dans la Section 4.3, nous pouvons remarquer que pour le SALBP-1 l'égalité $\widehat{B}_k = \widetilde{V}_k$ est vérifiée pour tout $k \in \widehat{W}^s$. Ainsi, le Théorème 4.2 et la Conséquence 4.5 nous amènent à des conséquences déjà démontrées pour le cas particulier du SALBP-1.

Conséquence 4.7. [Sotskov et al., 2006] *Le $\widehat{\mathcal{F}}$ -rayon de stabilité de la solution $s \in \mathbf{S}_{\widehat{\mathcal{F}}}(t)$ pour SALBP-1 est calculé comme suit :*

$$\rho_{\widehat{\mathcal{F}}}(s, t) = \max_{k \in \widehat{W}^s} \max_{j \in \{0\} \cup \mathcal{I}_{|\widetilde{V}_k|}} \frac{t_w(V_k) - T_0 - \sum_{i \in \mathcal{I}_j} t_{l_i}}{|\widetilde{V}_k| - j},$$

où

$$0 = t_{l_0} \leq \dots \leq t_{l_{|\widetilde{V}_k|}}, \quad l_i \in \widetilde{V}_k, \quad i \in \mathcal{I}_{|\widetilde{V}_k|},$$

sont les temps d'exécution (triés dans l'ordre croissant) des opérations incertaines affectées à la station k .

Conséquence 4.8. [Sotskov et al., 2006] *Chaque solution quasi-faisable de SALBP-1 est $\widehat{\mathcal{F}}$ -stable.*

Algorithme 4 : Le calcul du $\widehat{\mathcal{F}}$ -rayon de stabilité

```

1  $flag \leftarrow \text{vrai}$  /* Tant que  $s$  peut devenir faisable */
2 pour chaque  $k \in \widehat{W}^s$  faire
3   si  $\widetilde{B}_k = \emptyset$  ou  $\widehat{B}_k = \emptyset$  alors
4      $\rho_{\widehat{\mathcal{F}}}(s, t) \leftarrow +\infty$ 
5      $flag \leftarrow \text{faux}$  /*  $s$  ne peut pas devenir faisable */
6     quitter la boucle
7   sinon
8     Trier  $\widehat{\Delta}_{kl}, l \in \widehat{B}_k : 0 = \widehat{\Delta}_{kl_0} < \widehat{\Delta}_{kl_1} \leq \dots \leq \widehat{\Delta}_{kl_{|\widehat{B}_k|}}$ 
9      $j \leftarrow 0, \widehat{\varphi}(k) \leftarrow 0, \widehat{\Theta} \leftarrow 0$ 
10    tant que  $j < |\widehat{B}_k|$  faire
11      si  $\widehat{\Theta} + (|\widehat{B}_k| - j)(\widehat{\Delta}_{kl_{j+1}} - \widehat{\Delta}_{kl_j}) \leq t_w(V_k) - T_0$  alors
12         $\widehat{\Theta} = \widehat{\Theta} + (|\widehat{B}_k| - j)(\widehat{\Delta}_{kl_{j+1}} - \widehat{\Delta}_{kl_j}), \widehat{\varphi}(k) \leftarrow \widehat{\Delta}_{kl_{j+1}}, j \leftarrow j + 1$ 
13      sinon
14        quitter la boucle
15      si  $j < |\widehat{B}_k|$  alors
16         $\widehat{\varphi}(k) \leftarrow \widehat{\varphi}(k) + \frac{t_w(V_k) - T_0 - \widehat{\Theta}}{|\widehat{B}_k| - j}$ 
17      sinon
18         $\rho_{\widehat{\mathcal{F}}}(s, t) \leftarrow +\infty$ 
19         $flag \leftarrow \text{faux}$  /*  $s$  ne peut pas devenir faisable */
20        quitter la boucle
21 si  $flag = \text{vrai}$  alors
22    $\rho_{\widehat{\mathcal{F}}}(s, t) \leftarrow \max\{\widehat{\varphi}(k) : k \in \widehat{W}^s\}$ 

```

4.5 Solutions optimales

Cette section s'intéresse à la stabilité des solutions optimales sous VTEO. En fait, il existe deux cas où une solution optimale s peut perdre son optimalité pour un nouveau vecteur perturbé t^* :

- s perd sa faisabilité (ce cas est étudié dans la Section 4.3) ;
- s reste faisable ($s \in \mathbf{S}_{\mathcal{F}}(t^*)$), mais il existe une solution quasi-faisable s^0 telle que $\mathcal{Z}(s^0, t) < \mathcal{Z}(s, t)$ (ce cas est étudié dans la Section 4.4).

Alors, les Théorèmes 4.1 et 4.2 nous amènent au théorème suivant.

Théorème 4.3. *Le \mathcal{O} -rayon de stabilité $\rho_{\mathcal{O}}(s, t)$ de la solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ est calculé comme suit :*

$$\rho_{\mathcal{O}}(s, t) = \min \left\{ \rho_{\mathcal{F}}(s, t), \min_{s' \in \mathbf{S}_{\tilde{\mathcal{F}}}(t)} \{ \rho_{\tilde{\mathcal{F}}}(s', t) : \mathcal{Z}(s', t) < \mathcal{Z}(s, t) \} \right\}.$$

D'après ce Théorème, la recherche du \mathcal{O} -rayon de stabilité est un problème difficile, car il est impératif de connaître l'ensemble des solutions quasi-faisables avec une valeur de la fonction objectif plus petite que celle de la solution optimale étudiée. Cependant, un algorithme de dichotomie numérique ϵ -approché de calcul du \mathcal{O} -rayon de stabilité d'une solution optimale s peut être développé en se basant sur les résultats obtenus. Cet algorithme est présenté dans Algorithme 5.

Algorithme 5 : Calcul numérique approché du \mathcal{O} -rayon de stabilité

```

1   $a \leftarrow 0, b \leftarrow \max\{t_j : j \in \tilde{V}\}$ 
2  tant que  $b - a > \epsilon$  faire
3      pour chaque  $j \in V$  faire
4          si  $j \in \tilde{V}$  alors
5               $t'_j \leftarrow \max\{0, t_j - \frac{a+b}{2}\}$ 
6          sinon
7               $t'_j \leftarrow t_j$ 
8      Trouver la solution optimale  $s^*$  pour le vecteur  $t'$ 
9      si  $\mathcal{Z}(s^*, t) < \mathcal{Z}(s, t)$  alors
10          $b \leftarrow \frac{a+b}{2}$  /* Une solution quasi-faisable devient optimale */
11     sinon
12          $a \leftarrow \frac{a+b}{2}$  /* La solution  $s$  reste optimale */
13  $\rho_{\mathcal{O}}(s, t) \leftarrow \min\{\rho_{\mathcal{F}}(s, t), \frac{a+b}{2}\}$ 

```

En utilisant les Conséquences 4.1 et 4.5, nous obtenons ce qui suit.

Conséquence 4.9. *Une solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ n'est pas \mathcal{O} -stable ssi le suivant est vérifié :*

$$\exists k \in \tilde{W}^s \quad (t_w(V_k) = T_0 \quad \& \quad \tilde{B}_{k0} \neq \emptyset). \quad (4.21)$$

Dans le cas du SALBP-1, le Théorème 4.3 et la Conséquence 4.9 nous amènent aux résultats connus.

Conséquence 4.10. [Sotskov *et al.*, 2006] *Le \mathcal{O} -rayon de stabilité $\rho_{\mathcal{O}}(s, t)$ de la solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ pour SALBP-1 est calculé comme suit :*

$$\rho_{\mathcal{O}}(s, t) = \min \left\{ \rho_{\mathcal{F}}(s, t), \min_{s' \in \mathbf{S}_{\tilde{\mathcal{F}}}(t)} \{ \rho_{\tilde{\mathcal{F}}}(s', t) : |W^{s'}| < |W^s| \} \right\}.$$

Conséquence 4.11. [Sotskov *et al.*, 2006] Une solution $s \in \mathbf{S}_{\mathcal{O}}(t)$ de SALBP-1 n'est pas \mathcal{O} -stable ssi il existe $k \in \widetilde{W}^s$ tel que $T_0 = t_w(V_k)$.

La section suivante est dédiée à la recherche du compromis entre la mesure de stabilité proposée et la valeur de la fonction objectif.

4.6 Recherche du compromis

Pour rechercher un compromis entre le coût de la ligne et la valeur du \mathcal{F} -rayon de stabilité, une approche basée sur le concept de l'optimalité au sens de Pareto [Ehrgott, 2005] est proposée. Comme dans le Chapitre 3, nous nous intéressons à la construction d'approximations de l'ensemble des solutions non-dominées ($\mathcal{ASN}\mathcal{D}$).

Pour construire cette approximation, un algorithme itératif est utilisé. À chaque étape, une procédure heuristique multi-start $\mathcal{H}(\rho)$ est appliquée. Elle construit des solutions faisables dont le \mathcal{F} -rayon de stabilité est supérieur à la valeur courante de ρ . Le temps pendant lequel l'heuristique est employée à chaque étape est limité par T_{\max} . Cette heuristique peut être considérée comme une modification de l'approche développée dans [Dolgui *et al.*, 2005]. Parmi les solutions construites à l'étape courante, seules celles avec un coût minimal sont considérées et la solution avec la plus grande valeur du \mathcal{F} -rayon de stabilité est choisie parmi elles. La valeur du \mathcal{F} -rayon de stabilité de la solution choisie sert comme une valeur courante de ρ , pour l'étape suivante. Cela continue jusqu'à ce que $\rho \geq T_0 - \tilde{t}_{\max}$ soit vérifié, où $\tilde{t}_{\max} = \max\{t_j : j \in \widetilde{V}\}$. Au départ de l'algorithme, $\rho = -1$.

Soit \mathcal{S} l'ensemble des solutions faisables choisies par l'algorithme. Alors, l'approximation utilisée $\mathcal{ASN}\mathcal{D}$ peut être exprimée comme suit :

$$\mathcal{ASN}\mathcal{D} = \{s \in \mathcal{S} : \nexists s' \in \mathcal{S} \ (s' \succ s)\},$$

où \succ est la relation binaire entre deux solutions faisables quelconques, s et s' , exprimant la règle de dominance au sens de Pareto (voir le Chapitre 3).

Pour une valeur donnée de ρ , l'heuristique construit une solution faisable en affectant autant d'opérations que possible au bloc courant de la station courante. Au départ, la solution faisable courante contient seulement une station vide avec un bloc vide. L'heuristique affecte les opérations à ce bloc jusqu'à ce qu'aucune opération ne puisse être affectée à cause des contraintes existantes. Ensuite, un nouveau bloc est ouvert (ou une nouvelle station avec un bloc vide), qui devient le bloc courant, et l'heuristique continue à affecter des opérations à ce bloc. Cela continue tant que toutes les opérations ne sont pas affectées, jusqu'à obtenir une solution faisable.

Soient k et l respectivement les numéros de la station courante et du bloc courant de la solution s en cours de construction. Pour choisir une opération à affecter à V_{kl} ,

la liste $\mathcal{CL}(k, l)$ d'opérations candidates est générée. La liste $\mathcal{CL}(k, l)$ contient toutes les opérations qui peuvent être affectées au bloc l de la station k . Cette liste est construite comme suit : les opérations qui n'ont pas été encore affectées sont examinées et l'opération j est placée dans $\mathcal{CL}(k, l)$ si toutes les conditions suivantes sont vérifiées :

- tous les prédécesseurs de j ont déjà été affectés,
- j n'est pas liée par des contraintes d'exclusion avec une opération déjà affectée au bloc k de la station l ,
- l'affectation de j ne viole pas la contrainte de capacité :

$$t_w(V_k \setminus V_{kl}) + t_b(V_{kl} \cup \{j\}) \leq T_0,$$

- lors de l'affectation de j , l'inégalité suivante doit être satisfaite : $\varphi(k) > \rho$, si $(V_k \cup \{j\}) \cap \tilde{V} \neq \emptyset$.

La dernière condition assure la construction d'une solution faisable dont le \mathcal{F} -rayon de stabilité est supérieur à ρ .

Si $\mathcal{CL}(k, l) = \emptyset$, il n'y a plus d'opérations à affecter au bloc courant. Un nouveau bloc est ouvert (si $l < b_0$) et $\mathcal{CL}(k, l+1)$ est construit, sinon une nouvelle station avec un bloc vide est ouvert et $\mathcal{CL}(k+1, 1)$ est construit. Si $\mathcal{CL}(k, l) \neq \emptyset$, une opération j est choisie au hasard dans $\mathcal{CL}(k, l)$ et est affectée à V_{kl} , $\mathcal{CL}(k, l)$ est ensuite reconstruite.

Une évaluation de l'algorithme proposé sur certaines instances proches de cas industriels est présentée dans la section suivante.

4.7 Résultats numériques

Les expérimentations ont été effectuées sur une machine équipée d'un processeur Intel Celeron 550 (2 GHz, 1 GB RAM). L'algorithme suggéré a été réalisé en utilisant le langage C++ et évalué sur 2 séries de 20 instances proches de cas industriels. Ces instances ont été générées en tenant compte des caractéristiques réelles de pièces à usiner sur les lignes de transfert à boîtiers multibroches. Les données initiales de ces instances peuvent être trouvées sur <http://www.emse.fr/~gurevsky/GALBP.zip> et sont présentées dans le Tableau 4.1, où :

- $\#$ est la cardinalité de l'ensemble correspondant,
- $OS = \frac{2|\mathcal{A}|}{|V|(|V|-1)} \cdot 100\%$ est la densité des contraintes de précedence représentées par le graphe $G = (V, \mathcal{A})$.

La Série 2, en comparaison avec la Série 1, est caractérisée par un plus grand nombre d'opérations et de contraintes d'exclusion. Pour ces deux séries, les données d'entrée suivantes ont été utilisées : $C_1 = 1$, $C_2 = 0.5$, $b_0 = 4$, $T_{\max} = 25$ sec.

TABLEAU 4.1 – Instances

	Série 1					Série 2				
	#V	T_0	OS	#E ^b	#E ^w	#V	T_0	OS	#E ^b	#E ^w
1	68	5.15	43.6	940	734	99	7.05	41.3	2210	1576
2	71	5.5875	51	1029	749	111	4.0125	45.7	2656	2230
3	78	4.6125	38.7	1399	1026	94	6.1375	44.3	1754	1662
4	71	6.575	53	1127	694	122	5.0625	43.8	2879	2549
5	72	7.8975	42.1	926	1018	105	4.9	47.8	1918	2318
6	74	5.5375	49.8	1266	705	97	5.4375	45.9	2128	1610
7	71	4.5875	49.7	998	793	101	4.3	36.9	2117	1566
8	75	6.0125	47.1	1071	911	87	5.7625	32.9	1427	1364
9	75	7.8975	49.5	1098	1009	110	5.475	40.3	2379	2073
10	76	5.7	41.1	1259	923	113	4.375	45.9	2784	2281
11	81	4.275	42.8	1567	918	113	5.85	39.5	2492	2370
12	92	5.9	46.5	1787	1203	96	4.7125	49.5	1815	1431
13	71	4.8875	52.7	1131	786	116	4.375	47.3	2332	2707
14	65	2.9125	42.8	1043	574	121	5.075	44.7	2699	2339
15	46	3.925	40	465	302	111	5.1375	39.7	2402	2008
16	74	5.075	42	1089	912	98	5.075	35.8	2218	1602
17	74	5.2	45.8	1039	1142	114	4.9875	48.2	2408	2421
18	70	3.9375	38.6	990	909	99	5.975	44	1856	1653
19	69	3.85	44.5	939	767	119	4.625	44.5	2615	2706
20	64	6.1125	38.9	770	783	127	7.8975	40.7	4034	2245

Pour chaque instance, neuf tests ont été exécutés avec un pourcentage d'opérations incertaines choisi dans l'ensemble suivant $\{10\%, 20\%, \dots, 90\%\}$. Chaque nouveau test est obtenu à partir du test précédent en conservant les mêmes opérations incertaines et en choisissant 10% de nouvelles opérations incertaines parmi les opérations déterministes.

Les Tableaux 4.2-4.4 et 4.5-4.7 montrent la performance de l'algorithme proposé pour les Séries 1 et 2, respectivement, où les notations suivantes sont utilisées : \mathcal{N} est le nombre de solutions non-dominées trouvées (c'est-à-dire $\mathcal{N} = \#\mathcal{ASN}\mathcal{D}$), \mathcal{Z}_{moy} est la valeur moyenne de la fonction objectif parmi les solutions de $\mathcal{ASN}\mathcal{D}$; ρ_{moy} est la valeur moyenne du \mathcal{F} -rayon de stabilité parmi les solutions de $\mathcal{ASN}\mathcal{D}$, \mathcal{Q} est une mesure spécifique du front $\mathcal{ASN}\mathcal{D}$ qui analyse la situation où l'augmentation relative du \mathcal{F} -rayon de stabilité est plus importante que l'augmentation relative de la fonction objectif entre chaque paire

de solutions voisines de $\mathcal{ASN}\mathcal{D}$. Autrement dit,

$$Q = \max_{i \in \mathcal{I}_{N-1}} \left\{ \frac{\rho_{\mathcal{F}}(s_{i+1}, t) - \rho_{\mathcal{F}}(s_i, t)}{\rho_{\mathcal{F}}(s_i, t)} \bigg/ \frac{\mathcal{Z}(s_{i+1}, t) - \mathcal{Z}(s_i, t)}{\mathcal{Z}(s_i, t)} \right\},$$

à condition que les solutions non-dominées $\mathcal{ASN}\mathcal{D} = \{s_1, s_2, \dots, s_N\}$ soient triées dans l'ordre croissant selon leurs valeurs de la fonction objectif (ou du \mathcal{F} -rayon de stabilité). D'une certaine manière, la valeur de Q caractérise la qualité du compromis fourni par les solutions de $\mathcal{ASN}\mathcal{D}$.

En analysant les résultats obtenus, nous pouvons voir que l'algorithme proposé a fourni en moyenne plus de solutions non-dominées pour la Série 2 que pour la Série 1 (voir la Figure 4.5). La conclusion opposée peut être faite pour la valeur moyenne ρ_{moy} (voir la Figure 4.6). Cela peut être expliqué par le fait que l'augmentation du nombre d'opérations et de contraintes provoque une augmentation du nombre de solutions faisables construites dans chaque étape de l'algorithme proposé. Par conséquent, cela implique une augmentation de densité de la distribution des solutions dans $\mathcal{ASN}\mathcal{D}$. Un exemple illustratif est donné dans la Figure 4.7, où deux fronts avec le même pourcentage d'opérations incertaines et le même nombre de solutions non-dominées pour les Séries 1 et 2 sont montrés.

Notons que, pratiquement pour tous les tests, Q est déterminé par les solutions situées près des solutions avec la valeur minimale de la fonction objectif. Cela montre l'adéquation de la mesure introduite. De plus, la valeur moyenne de Q , à travers tous les tests effectués, est égale à 60.91 pour la Série 1 et 100.27 pour la Série 2. Les valeurs assez grandes de Q montrent l'intérêt de notre approche de recherche d'un compromis.

4.8 Conclusion

Dans ce chapitre, nous avons étudié le cas de lignes d'usinage avec des stations à boîtiers multibroches permettant une exécution simultanée des opérations. Le problème d'équilibrage de ce type de lignes d'usinage a été examiné sous VTEO. Les conditions de stabilité et la formule du rayon de stabilité pour les solutions faisables, quasi-faisables, et optimales ont été développées. La complexité algorithmique de ces problèmes a été discutée. Des algorithmes polynomiaux pour le calcul du rayon de stabilité ont été proposés pour les solutions faisables et quasi-faisables, alors que pour les solutions optimales seul un algorithme approché de calcul du rayon de stabilité a pu être développé.

Notons que les résultats déjà connus pour le SALBP-1 peuvent être retrouvés à partir des résultats présentés dans ce chapitre.

Le concept de l'optimalité au sens de Pareto a été utilisé afin de trouver des compromis entre la fonction objectif de la solution étudiée à minimiser et son rayon de stabilité (au

sens de faisabilité) à maximiser. Un algorithme approché a également été proposé pour la recherche des solutions non-dominées au sens de Pareto pour le problème étudié. Cet algorithme peut être employé lors de la conception pour la recherche de compromis entre le coût de la ligne et le rayon de stabilité de solution.

Une communication à la conférence MOSIM'10 [Gurevsky *et al.*, 2010a] et un article soumis à la revue « Discrete Applied Mathematics » [Gurevsky *et al.*, xxxxb] sont les issues (en termes de productions scientifiques) des résultats présentés dans ce chapitre.

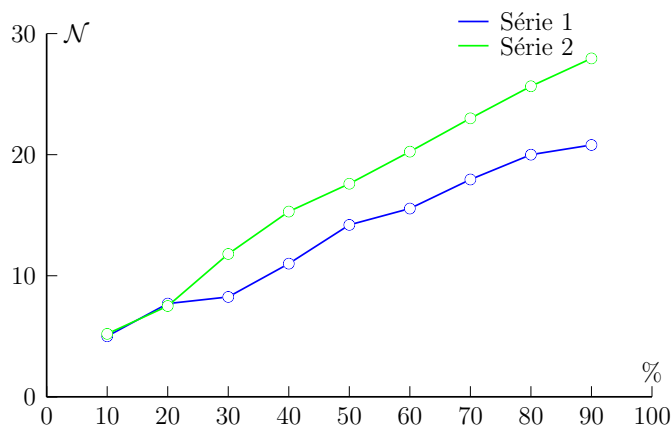


FIGURE 4.5 – Valeur moyenne de \mathcal{N} en fonction du pourcentage d'opérations incertaines.

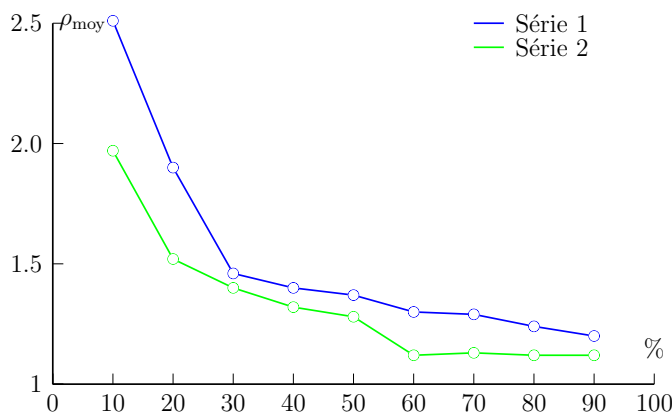


FIGURE 4.6 – Valeur moyenne ρ_{moy} en fonction du pourcentage d'opérations incertaines.

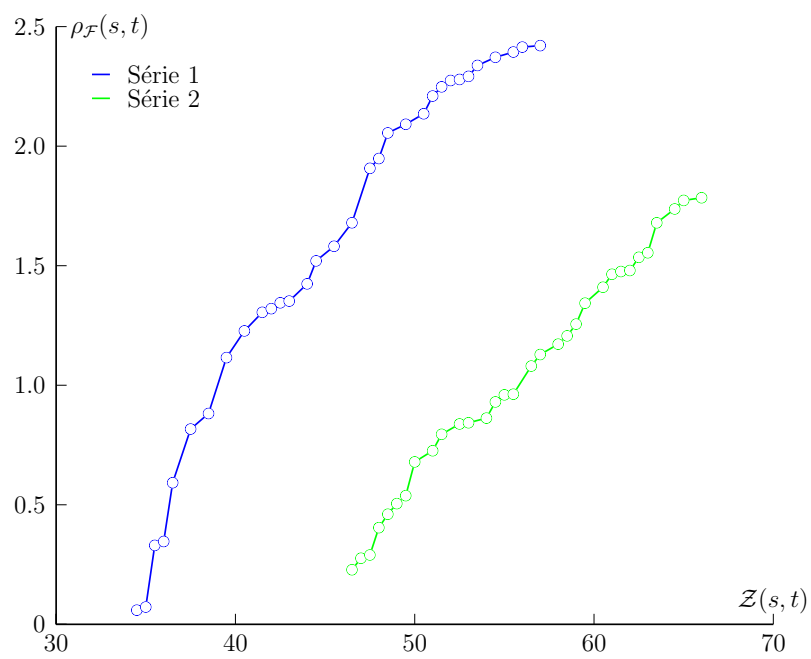


FIGURE 4.7 – $\mathcal{ASN}\mathcal{D}$ pour le Test 6 de la Série 1 et pour le Test 5 de la Série 2 avec 90% d'opérations incertaines.

TABLEAU 4.2 – Résultats pour la Série 1 (10%-30%)

	10%				20%				30%			
	\mathcal{N}	Z_{moy}	ρ_{moy}	\mathcal{Q}	\mathcal{N}	Z_{moy}	ρ_{moy}	\mathcal{Q}	#S	Z_{moy}	ρ_{moy}	\mathcal{Q}
1	8	27.25	3.17	20.34	12	29.21	2.51	26.08	3	26.67	1.35	19.07
2	7	31.21	2.74	14.86	13	33.58	2.30	18.17	5	31.10	1.38	14.49
3	2	47.00	2.25	5.81	5	47.70	2.03	15.80	13	49.50	1.95	19.82
4	7	27.00	4.25	6.03	9	29.00	3.45	25.75	3	26.50	1.52	25.73
5	7	35.00	4.12	17.58	5	34.70	2.50	22.20	4	34.13	1.29	16.40
6	3	35.50	1.51	140.97	7	36.43	1.63	82.42	10	38.10	1.63	95.61
7	7	27.43	2.10	18.40	4	26.50	1.30	11.88	8	27.00	1.17	19.84
8	4	25.75	3.31	13.68	8	27.44	2.48	18.65	12	28.71	2.12	12.71
9	8	42.25	3.96	48.16	3	41.00	1.54	48.42	4	41.25	1.52	71.79
10	7	37.64	3.33	25.72	13	39.50	2.95	46.94	7	36.50	1.32	38.45
11	5	44.70	2.01	18.42	11	45.77	1.48	16.19	11	45.64	1.22	39.21
12	7	44.36	3.89	53.32	16	46.81	2.81	20.42	26	49.62	2.34	42.24
13	2	36.50	1.16	85.17	4	37.50	1.27	92.99	6	38.50	1.25	53.89
14	2	36.00	1.52	12.50	6	38.00	1.27	19.58	13	40.35	1.26	16.04
15	5	28.70	2.46	22.50	5	29.00	1.56	20.36	7	30.00	1.47	51.42
16	3	42.50	2.01	18.00	6	43.50	1.47	27.93	5	43.80	1.45	19.77
17	4	41.13	1.15	96.90	7	42.50	1.19	43.72	9	42.89	1.12	444.01
18	3	41.50	1.42	11.04	8	42.81	1.11	144.58	9	43.50	1.05	57.82
19	4	30.00	1.12	44.15	6	30.58	1.05	127.30	8	31.56	1.02	62.15
20	5	21.80	2.68	30.94	6	22.25	2.11	21.08	12	23.25	1.78	94.90

TABLEAU 4.3 – Résultats pour la Série 1 (40%-60%)

	40%			50%			60%					
	\mathcal{N}	Z_{moy}	ρ_{moy}	Q	\mathcal{N}	Z_{moy}	ρ_{moy}	Q	\mathcal{N}	Z_{moy}	ρ_{moy}	Q
1	7	27.79	1.36	21.96	11	29.27	1.35	30.61	13	29.31	1.23	12.36
2	8	31.50	1.24	57.60	14	33.25	1.33	25.86	14	34.00	1.38	22.34
3	9	48.94	1.44	16.10	12	49.79	1.28	14.59	16	50.91	1.24	23.48
4	6	27.50	1.53	31.06	6	27.58	1.47	30.43	9	28.39	1.54	29.28
5	5	34.70	1.27	30.61	7	35.29	1.25	22.77	10	36.00	1.16	25.04
6	15	39.67	1.63	82.43	21	40.60	1.65	167.47	22	41.50	1.58	75.77
7	7	26.50	1.04	37.60	10	27.65	1.14	32.31	11	28.18	1.09	32.31
8	6	27.00	1.62	8.89	9	27.22	1.57	9.00	12	27.92	1.40	19.14
9	4	41.88	1.70	33.03	9	42.83	1.57	89.83	9	43.33	1.64	24.47
10	10	37.70	1.48	18.18	12	38.58	1.40	23.31	14	39.25	1.40	21.06
11	13	46.96	1.20	27.09	11	47.86	1.19	19.95	14	47.82	1.02	33.77
12	31	51.68	2.25	31.38	40	53.85	2.23	78.31	19	47.76	1.31	30.17
13	8	39.50	1.23	47.38	12	40.79	1.19	72.91	16	42.38	1.24	61.42
14	18	42.11	1.20	23.65	21	42.88	1.02	28.91	22	45.20	1.01	23.45
15	11	32.00	1.52	45.64	13	33.00	1.41	49.67	21	34.90	1.40	41.95
16	10	44.40	1.28	36.25	12	45.04	1.25	35.40	14	46.29	1.28	32.32
17	13	44.00	1.15	51.53	15	45.07	1.16	32.46	18	45.81	1.12	28.29
18	15	44.53	0.96	60.58	18	45.42	0.92	42.47	22	47.11	0.95	53.50
19	10	32.20	1.05	134.33	18	34.53	1.07	371.72	19	35.29	1.03	171.61
20	14	23.75	1.86	60.13	13	25.19	2.02	101.44	16	25.41	1.93	94.77

TABLEAU 4.4 – Résultats pour la Série 1 (70%-90%)

	70%				80%				90%			
	\mathcal{N}	Z_{moy}	ρ_{moy}	\mathcal{Q}	\mathcal{N}	Z_{moy}	ρ_{moy}	\mathcal{Q}	\mathcal{N}	Z_{moy}	ρ_{moy}	\mathcal{Q}
1	11	29.23	1.20	16.02	15	29.57	1.08	30.71	15	29.83	1.09	29.06
2	18	34.61	1.30	17.68	21	34.81	1.27	84.76	22	36.07	1.29	28.30
3	18	51.50	1.19	18.65	22	53.52	1.25	10.74	25	54.28	1.20	9.88
4	12	28.83	1.50	19.53	11	28.82	1.41	27.52	16	30.22	1.47	29.32
5	11	36.36	1.21	18.65	11	36.50	1.13	25.44	8	36.50	1.06	13.28
6	25	43.16	1.69	31.21	30	44.22	1.54	121.62	32	45.73	1.57	249.80
7	13	28.73	1.14	17.96	13	29.46	1.18	16.45	11	28.32	1.04	16.63
8	14	29.25	1.51	11.56	14	29.21	1.48	8.70	13	28.92	1.38	12.95
9	11	43.45	1.46	33.81	13	43.73	1.26	97.48	12	44.79	1.43	46.64
10	14	39.68	1.37	15.82	19	39.58	1.26	22.78	20	40.93	1.31	21.35
11	22	50.25	1.07	27.68	28	52.16	1.03	1856.39	33	53.81	1.04	19.23
12	22	48.80	1.31	43.07	26	50.12	1.35	20.39	25	49.88	1.26	32.84
13	22	42.89	1.14	54.37	21	44.00	1.16	45.26	18	44.50	1.16	48.91
14	26	46.10	0.99	22.65	31	47.58	0.95	22.14	34	48.75	0.89	19.83
15	19	36.00	1.38	52.67	17	35.18	1.26	49.86	21	35.14	1.13	36.89
16	18	46.83	1.23	19.00	24	48.25	1.18	16.80	22	48.70	1.19	14.89
17	21	47.67	1.22	21.66	22	47.70	1.12	50.87	31	48.58	1.10	624.87
18	26	48.42	0.97	38.43	29	49.26	0.91	57.88	28	50.71	0.93	42.47
19	17	35.47	1.03	192.54	14	35.36	0.99	192.54	15	35.50	0.98	192.54
20	19	25.53	1.94	94.41	19	27.05	2.02	94.77	15	25.13	1.45	92.04

TABLEAU 4.5 – Résultats pour la Série 2 (10%-30%)

	10%			20%			30%					
	\mathcal{N}	Z_{moy}	ρ_{moy}	Q	\mathcal{N}	Z_{moy}	ρ_{moy}	Q	$\#S$	Z_{moy}	ρ_{moy}	Q
1	4	42.00	5.08	13.70	13	44.65	3.82	27.22	21	46.07	3.33	19.06
2	4	62.88	1.29	63.12	9	64.50	1.08	25.04	17	66.79	0.99	61.00
3	5	49.00	2.27	50.38	9	50.17	2.22	35.23	12	50.75	2.03	26.75
4	6	57.50	1.59	129.55	8	57.25	0.90	826.07	14	58.82	1.03	74.65
5	5	48.30	1.68	50.07	11	48.77	1.43	291.70	11	49.32	1.15	29.72
6	7	48.71	1.28	74.35	13	50.23	1.36	41.88	15	51.53	1.38	71.91
7	8	42.69	1.71	54.35	10	43.30	1.06	61.64	13	44.58	0.95	71.73
8	3	45.50	2.76	7.86	3	46.00	2.53	17.50	13	48.69	2.42	16.29
9	6	36.75	3.52	29.24	4	35.63	1.44	50.65	5	36.40	1.51	39.70
10	6	63.75	1.50	46.04	9	64.94	1.27	43.39	13	66.08	1.12	52.16
11	4	35.38	1.18	145.01	8	36.81	1.38	92.11	11	37.50	1.30	44.29
12	3	33.50	0.96	121.28	5	33.90	1.13	374.57	9	35.00	1.07	111.54
13	11	48.64	2.43	529.05	9	47.11	1.14	32.72	7	48.07	1.22	30.91
14	5	43.80	1.35	195.40	8	45.13	1.24	36.27	13	45.85	1.05	49.56
15	3	36.00	0.76	246.17	6	37.42	1.21	79.48	13	39.04	1.20	383.44
16	5	45.00	2.93	15.41	5	45.00	1.48	51.21	11	46.50	1.35	38.29
17	9	50.72	3.02	263.52	1	48.50	2.10	0.00	8	50.44	1.56	74.23
18	3	38.00	2.12	42.27	8	39.50	1.83	47.31	13	40.81	1.68	56.87
19	5	41.70	1.35	41.18	9	42.94	1.21	45.96	15	44.27	1.04	24.48
20	2	62.50	0.60	86.36	2	62.50	0.64	58.47	2	62.50	0.55	145.92

TABLEAU 4.6 – Résultats pour la Série 2 (40%-60%)

	40%				50%				60%			
	\mathcal{N}	Z_{moy}	ρ_{moy}	\mathcal{Q}	\mathcal{N}	Z_{moy}	ρ_{moy}	\mathcal{Q}	\mathcal{N}	Z_{moy}	ρ_{moy}	\mathcal{Q}
1	16	44.94	2.73	10.55	23	46.67	2.52	9.52	12	43.71	1.60	38.40
2	27	68.78	0.96	75.74	29	70.07	0.92	53.75	32	72.70	0.90	55.31
3	16	51.88	2.11	22.22	25	55.36	2.10	27.10	15	52.37	1.46	23.46
4	16	61.25	1.16	66.29	23	61.67	1.10	263.00	25	63.78	1.14	48.33
5	15	50.50	1.14	568.28	17	51.26	1.12	322.58	24	52.48	1.03	34.36
6	19	52.42	1.15	58.32	22	52.43	1.03	736.17	24	54.42	1.14	48.88
7	19	46.16	0.91	157.69	22	47.59	0.93	68.32	25	47.76	0.82	42.14
8	18	50.75	2.28	26.76	26	54.17	2.23	24.76	14	49.00	1.36	83.44
9	12	37.92	1.40	23.77	14	38.96	1.38	54.90	17	39.53	1.35	54.90
10	18	66.97	1.02	79.45	25	69.94	1.05	108.55	32	71.56	1.00	57.13
11	10	38.25	1.34	102.55	13	39.35	1.35	28.21	15	39.13	1.18	197.48
12	13	36.23	0.99	309.72	13	36.77	0.99	148.97	17	37.65	0.96	93.43
13	15	49.13	1.01	83.65	16	49.50	0.98	52.29	20	51.15	0.93	167.56
14	13	46.65	1.05	95.99	17	47.41	0.96	125.24	23	48.67	0.91	161.70
15	14	39.75	1.22	35.38	15	42.07	1.31	29.38	20	41.78	1.12	177.42
16	16	47.84	1.22	32.24	18	48.44	1.17	36.27	17	48.41	1.08	34.41
17	16	53.31	1.62	19.67	15	54.70	1.61	34.84	30	56.72	1.47	606.51
18	12	40.38	1.30	24.43	14	40.79	1.30	30.97	15	41.40	1.26	31.11
19	17	45.94	1.07	30.46	16	45.84	1.01	31.01	25	47.28	1.02	30.07
20	4	62.75	0.63	154.98	4	62.88	0.61	234.76	3	63.00	0.61	195.25

TABLEAU 4.7 – Résultats pour la Série 2 (70%-90%)

	70%			80%			90%					
	\mathcal{N}	Z_{moy}	ρ_{moy}	Q	\mathcal{N}	Z_{moy}	ρ_{moy}	Q	\mathcal{N}	Z_{moy}	ρ_{moy}	Q
1	17	44.53	1.65	24.22	17	45.18	1.62	22.44	20	45.85	1.60	26.35
2	37	74.53	0.92	78.25	47	77.41	0.91	44.24	46	79.33	0.92	40.36
3	19	52.89	1.39	32.02	25	54.84	1.38	60.01	25	55.42	1.33	38.20
4	28	63.18	0.98	165.45	32	65.08	1.03	1094.33	41	67.26	1.04	294.15
5	23	53.87	1.07	40.21	25	54.14	1.01	49.18	32	55.97	1.04	37.45
6	29	55.41	1.13	52.42	30	56.50	1.14	47.00	30	56.35	1.15	71.53
7	30	49.50	0.94	183.32	31	49.29	0.87	147.55	34	51.28	0.93	147.71
8	16	49.31	1.28	67.31	19	50.42	1.31	45.17	21	52.02	1.38	44.74
9	19	40.50	1.35	54.90	19	39.95	1.33	54.90	17	41.18	1.39	54.90
10	38	74.07	1.04	68.07	42	73.92	0.98	69.89	42	75.99	1.01	75.73
11	23	40.78	1.20	58.82	19	40.87	1.18	93.23	22	41.75	1.18	41.62
12	19	39.03	1.03	100.58	26	39.90	1.03	39.90	22	39.18	0.95	176.77
13	21	52.40	0.92	48.32	21	53.40	1.00	24.50	29	52.95	0.86	156.31
14	24	50.69	1.07	46.24	32	51.95	1.03	128.34	33	51.89	1.03	165.25
15	22	42.77	1.14	70.56	22	43.52	1.18	111.08	27	43.50	1.11	213.10
16	22	49.80	1.09	41.78	27	51.44	1.17	28.49	28	52.14	1.13	61.46
17	28	57.91	1.47	44.05	31	59.55	1.45	91.69	41	61.01	1.41	101.16
18	19	42.03	1.31	33.51	17	42.38	1.25	37.84	20	43.48	1.30	29.99
19	21	47.24	0.99	38.67	25	48.18	0.99	34.55	23	48.04	1.02	24.07
20	5	63.30	0.65	97.78	6	63.67	0.58	132.86	6	63.67	0.61	97.68

Chapitre 5

Équilibrage de lignes d'assemblage de type SALBP-1 : approche robuste

Sommaire

5.1	Introduction	93
5.2	Définition du problème	94
5.2.1	Modèle déterministe	94
5.2.2	Modèle robuste	95
5.3	Algorithme par séparation et évaluation	96
5.4	Résultats numériques	98
5.5	Conclusion	101

5.1 Introduction

Ce chapitre s'intéresse au problème SALBP-1 avec des temps d'exécution des opérations connus sous la forme d'intervalles des valeurs possibles. Rappelons que le SALBP-1 consiste à affecter un ensemble d'opérations d'assemblage à des stations tout en respectant des contraintes de précédence et le temps de cycle (contrainte de capacité), avec pour objectif de minimiser le nombre de stations.

En se basant sur l'idée de l'article [Bertsimas and Sim, 2003], nous proposons un modèle robuste du SALBP-1 en utilisant un paramètre θ qui exprime le pourcentage d'opérations par station dont le temps d'exécution est susceptible de varier. Afin de trouver une solution optimale pour ce problème, nous proposons une procédure par séparation et évaluation.

Le reste du chapitre est organisé de la manière suivante. Dans la Section 5.2, les modèles déterministe et robuste du SALBP-1 sont présentés. La procédure par séparation

et évaluation pour le modèle robuste est décrite dans la Section 5.3. Les résultats des tests numériques sur des instances connues constituent la Section 5.4. Une conclusion est présentée dans la Section 5.5.

5.2 Définition du problème

5.2.1 Modèle déterministe

Toutes les opérations nécessaires pour finir l'assemblage du produit constituent l'ensemble $V = \{1, 2, \dots, n\}$ associé au vecteur $t = (t_1, t_2, \dots, t_n) \in \mathbb{R}_+^n$ de temps d'exécution, où t_j est le temps d'exécution de l'opération $j \in V$ et \mathbb{R}_+ est l'ensemble des nombres réels positifs. L'ensemble des opérations doivent être affectées aux stations en tenant compte des restrictions suivantes :

- Les *contraintes de capacité* qui sont caractérisées par le temps de cycle T_0 .
- Les *contraintes d'exclusion* qui définissent les paires d'opérations qui ne peuvent pas être affectées ensemble à la même station à cause d'incompatibilités technologiques. Ces contraintes sont représentées par la famille E de paires d'éléments de V telle que les opérations du sous-ensemble $e \in E$ ne peuvent pas être affectées à la même station.
- Les *contraintes de précedence* qui introduisent des relations partielles non-strictes entre les opérations. Elles sont représentées par un graphe orienté acyclique $G = (V, \mathcal{A})$. Un arc (i, j) appartient à \mathcal{A} si et seulement si l'opération j doit être affectée à une station qui ne précède pas la station où l'opération i est affectée. Néanmoins, cette relation partielle autorise l'affectation des opérations i et j à la même station.

En utilisant les notations introduites ci-dessus, le modèle déterministe de SALBP-1 est exprimé comme suit :

$$\text{Minimiser } \sum_{k=1}^{m_0} y_k \quad (5.1)$$

sous les contraintes

$$\sum_{k=1}^{m_0} x_{jk} = 1, \quad \forall j \in V, \quad (5.2)$$

$$\sum_{k=1}^{m_0} kx_{ik} \leq \sum_{k=1}^{m_0} kx_{jk}, \quad \forall (i, j) \in \mathcal{A}, \quad (5.3)$$

$$x_{ik} + x_{jk} \leq 1, \quad \forall \{i, j\} \in E, \quad \forall k \in \{1, 2, \dots, m_0\}, \quad (5.4)$$

$$\sum_{j \in V} t_j x_{jk} \leq T_0 y_k, \quad \forall k \in \{1, 2, \dots, m_0\}, \quad (5.5)$$

$$y_k \in \{0, 1\}, \quad \forall k \in \{1, 2, \dots, m_0\},$$

$$x_{jk} \in \{0, 1\}, \quad \forall j \in V, \quad \forall k \in \{1, 2, \dots, m_0\},$$

où m_0 est le nombre maximal admissible de stations. La variable y_k est égale à 1 si la station k est ouverte, 0 sinon. La variable binaire x_{jk} est égale à 1, si l'opération j est affectée à la station k et 0, sinon. L'expression (5.1) représente la fonction objectif. L'égalité (5.2) impose que chaque opération soit affectée à une seule station. Les contraintes de précédence sont modélisées par les inégalités (5.3). Les inégalités (5.4) introduisent les contraintes d'exclusion. Les inégalités (5.5) vérifient les contraintes de capacité.

5.2.2 Modèle robuste

Nous supposons que les temps d'exécution t_j , $j \in V$ sont connus sous la forme d'intervalles $[a_j, b_j]$ des valeurs possibles, où $0 < a_j \leq b_j$. C'est la seule information dont nous disposons. Nous ne connaissons pas de loi de distribution de probabilités à l'intérieur de ces intervalles. Il est évident que le pire des cas pour la réalisation des temps d'exécution des opérations (PCRTEO) correspond à $t_j = b_j$ pour tout $j \in V$, mais ce cas est assez improbable en pratique. Deux types d'opérations peuvent être distingués dans V :

- Opérations incertaines : leur temps d'exécution peut varier et le cas $t_j = b_j$ est possible pour elles.
- Opérations déterministes : leur temps d'exécution reste fixe et est égal à a_j .

Pour contrôler le degré de conservatisme, nous utilisons un paramètre $\theta \in [0, 1]$ qui reflète un risque du décideur. Ainsi, par exemple, le nombre d'opérations incertaines est limité par $\theta \cdot |V_k|$ pour chaque station k , où V_k est l'ensemble des opérations affectées à la station k .

De ce fait, le paramètre θ introduit une flexibilité permettant de se positionner entre le modèle déterministe et l'optimisation du PCRTEO. En effet, $\theta = 0$ (le cas optimiste) implique $t_j = a_j$, $j \in V$ (il n'existe pas d'opérations incertaines), alors que $\theta = 1$ (le cas pessimiste) signifie que le cas $t_j = b_j$ peut arriver pour tout $j \in V$ (toutes les opérations sont incertaines).

Cette approche est intégrée dans le modèle précédent en utilisant les contraintes (5.5') en remplacement de (5.5) :

$$\sum_{j \in V} a_j x_{jk} + \gamma_k(x) \leq T_0 y_k, \quad \forall k \in \{1, 2, \dots, m_0\}, \quad (5.5')$$

où

$$\gamma_k(x) = \max \left\{ \sum_{j \in V} (b_j - a_j) z_{jk} x_{jk} \mid \sum_{j \in V} z_{jk} \leq \theta \sum_{j \in V} x_{jk}; z_{jk} \in \{0, 1\} \right\}$$

est un problème de sac-à-dos qui détermine le pire des cas de la réalisation (en fonction du paramètre θ) des temps d'exécution des opérations affectées à la station k . Ici, la variable binaire z_{jk} est égale à 1 si l'opération j est considérée comme incertaine dans la station k , 0 sinon.

Pratiquement tous les algorithmes par séparation et évaluation développés pour le modèle déterministe de SALBP-1 [Scholl, 1999, chapitre 4] sont basés sur des techniques de calcul d'une borne inférieure locale qui ne sont pas applicables pour notre modèle robuste de SALBP-1. Par conséquent, un nouvel algorithme par séparation et évaluation est suggéré dans la section suivante.

5.3 Algorithme par séparation et évaluation

Notre algorithme par séparation et évaluation se base sur l'exploration systématique des solutions candidates à travers le développement d'un arbre énumératif multi-niveaux. Dans notre contexte, la structure de l'arbre a les propriétés suivantes. Chaque *niveau* est composé de *nœuds* correspondant aux solutions partiellement construites. Ainsi, par exemple, le niveau k contient toutes les solutions partielles dont la k -ème station est déjà chargée. Le niveau 0 se compose uniquement de la racine de l'arbre représentant une solution vide. Les *feuilles* de l'arbre expriment soit les solutions partielles dominées soit les solutions optimales.

Nous utilisons une recherche en largeur d'abord (RLA) afin de trouver toutes les solutions optimales. Cela signifie que toutes les solutions partielles possibles du niveau 1 sont générées et évaluées à partir de la racine. Ensuite, toutes les solutions partielles du niveau 2 sont générées à partir des solutions non-dominées du niveau 1 et ainsi de suite. Cela continue jusqu'à ce que tous les nœuds générés pour le niveau courant soient des *feuilles*.

Avant de donner les détails de la RLA, nous introduisons d'abord quelques notations :

- L_k est l'ensemble des solutions partielles correspondant au niveau k ,
- V_k^s est l'ensemble des opérations affectées à la station k dans la solution s .

La règle de dominance suivante est utilisée pour réduire l'espace de recherche.

Proposition 5.1 (Règle de dominance). *Si s_1 et s_2 appartiennent à L_k , $k \geq 1$ tels que $\cup_{i=1}^k V_i^{s_1} \supseteq \cup_{i=1}^k V_i^{s_2}$, alors s_2 est dominée.*

Démonstration. Introduisons une notation auxiliaire $Q(s, k)$ qui définit l'ensemble des solutions complètes dans lesquelles les premières k stations sont identiques à celles de la solution s . Alors, la Proposition 5.1 est équivalente à la proposition suivante : $\min\{w^s : s \in Q(s_1, k)\} \leq \min\{w^s : s \in Q(s_2, k)\}$, où w^s est le nombre de stations de la solution s . Pour prouver ceci, supposons au contraire qu'il existe $s' \in Q(s_2, k)$ telle que $w^{s'} < \min\{w^s : s \in Q(s_1, k)\}$. Alors, en construisant la solution s'' comme suit :

- $V_i^{s''} = V_i^{s_1}$ pour tout $i \in \{1, \dots, k\}$,
- $V_i^{s''} = V_i^{s'} \setminus U$ pour tout $i \in \{k+1, \dots, w^{s'}\}$, où $U = \cup_{i=1}^k V_i^{s_1} \setminus \cup_{i=1}^k V_i^{s_2}$,

nous obtenons que $w^{s''} = w^{s'}$ et $s'' \in Q(s_1, k)$. Mais ceci contredit ce que nous avons supposé au départ. □

Pour construire L_{k+1} à partir de L_k , l'ensemble W_{k+1} est utilisé. Au début, cet ensemble se compose de toutes les solutions partielles non-dominées de L_k . Ensuite, chaque solution de W_{k+1} est considérée et toutes les affectations possibles à la station $(k+1)$ sont générées à partir de cette solution. Toutes les solutions partielles générées de cette façon constituent L_{k+1} . Puis, le niveau suivant est considéré. Cela continue jusqu'à ce que toutes les opérations soient affectées, c'est-à-dire qu'une solution optimale soit obtenue.

La charge de la station $(k+1)$ de la solution $s \in W_{k+1}$ est formée pas à pas en affectant les opérations autant que possible (règle de dominance de Jackson, voir [Jackson, 1956]). Pour choisir une opération à affecter à V_{k+1}^s , une liste $\mathcal{CL}(s, k+1)$ est créée. La liste $\mathcal{CL}(s, k+1)$ contient toutes les opérations qui peuvent être affectées à la station $k+1$ de la solution s . Cette liste est générée comme suit : les opérations qui n'ont pas été encore affectées sont examinées et l'opération j est placée dans $\mathcal{CL}(s, k+1)$ si toutes les conditions suivantes sont vérifiées¹ :

- tous les prédécesseurs de j ont été déjà affectés,
- le numéro d'ordre de j est supérieur à celui des opérations déjà affectées à la station $k+1$ (afin d'éviter les permutations d'opérations),
- j n'est liée par des contraintes d'exclusion avec aucune autre opération affectée à la station $k+1$,
- l'affectation de j à la station $k+1$ ne viole pas les contraintes de capacité conformément à PCRTEO :

$$\sum_{i \in V_k^s \cup \{j\}} a_i + \max \left\{ \sum_{i \in U} (b_i - a_i) \mid U \subseteq V_k^s \cup \{j\}; |U| \leq \theta \cdot |V_k^s \cup \{j\}| \right\} \leq T_0.$$

1. Sans perte de généralité, il est supposé que les opérations dans le graphe de précedence sont numérotées de manière topologique.

Si $\mathcal{CL}(s, k+1) = \emptyset$, aucune opération ne peut être affectée à la station $k+1$. La solution s se déplace de W_{k+1} à L_{k+1} . Si $\mathcal{CL}(s, k+1) \neq \emptyset$, $|\mathcal{CL}(s, k+1)|$ nouvelles solutions sont générées à partir de la solution s en ajoutant à chaque fois seulement une opération $j \in \mathcal{CL}(s, k+1)$ à V_k^s . Toutes ces solutions se joignent à W_{k+1} et s est éliminée de W_{k+1} . Cela continue jusqu'à ce que W_{k+1} soit vide. Le schéma général de cette approche est donné par l'Algorithme 6.

Algorithme 6 : Recherche en largeur d'abord

```

1  $i \leftarrow 1, V_1^s \leftarrow \emptyset, L_1 \leftarrow \emptyset, W_1 \leftarrow \{s\}$ 
2 répéter
3   répéter
4     Considérer  $s \in W_i$ 
5     Construire la liste  $\mathcal{CL}(s, i)$ 
6     si  $\mathcal{CL}(s, i) \neq \emptyset$  alors
7       pour chaque  $j \in \mathcal{CL}(s, i)$  faire
8          $s' \leftarrow s, V_i^{s'} \leftarrow V_i^s \cup \{j\}$ 
9         si  $\bigcup_{k=1}^i V_k^{s'} = V$  alors
10          retourner  $s'$  /* Solution optimale est trouvée */
11        sinon
12           $W_i \leftarrow W_i \cup \{s'\}$ 
13      sinon
14         $L_i \leftarrow L_i \cup \{s\}$ 
15       $W_i \leftarrow W_i \setminus \{s\}$ 
16  jusqu'à  $W_i = \emptyset$ 
17   $W_{i+1}$  est composé des solutions partielles non-dominées de  $L_i$ 
18   $i \leftarrow i + 1$ 
19 jusqu'à false

```

L'évaluation de la performance de la méthode proposée sur quelques instances connues est faite dans la section suivante.

5.4 Résultats numériques

Les expérimentations ont été effectuées sur une machine équipée d'un processeur Intel Celeron 550 (2 GHz, 1 GB RAM). L'algorithme a été réalisé en utilisant le langage C++ et évalué sur 3 séries d'instance connues (voir <http://www.assembly-line-balancing.de>)

présentées dans le Tableau 5.1, où $OS = \frac{2|A|}{|V|(|V|-1)} \cdot 100\%$ est la densité des contraintes de précédence dans le graphe $G = (V, \mathcal{A})$.

Pour chaque instance, onze tests ont été exécutés avec le paramètre θ variant dans $\{0.0, 0.1, \dots, 1.0\}$. Pour toute opération $j \in V$, la borne inférieure de son temps d'exécution a_j a été initialisée par la valeur nominale du temps d'exécution t_j de l'instance correspondante, alors que la borne supérieure b_j a été fixée à $1.5t_j$.

TABLEAU 5.1 – Instances

Nom	n	T_0	$ E $	$OS, \%$
Petite Taille				
Bowman	8	26	4	75.00
Jackson	11	11	10	58.18
Jaeschke	9	10	4	83.33
Mansoor	11	67	7	60.00
Mertens	7	10	2	52.38
Taille Moyenne				
Buxey	29	38	54	50.74
Gunther	35	61	81	59.50
Heskia	28	163	342	22.49
Mitchell	21	20	39	70.95
Lutz1	32	2100	83	83.47
Roszieg	25	20	32	71.67
Sawyer	30	38	75	44.83
Grande Taille				
Hahn	53	2663	228	83.82
Lutz2	89	16	21	77.55
Lutz3	89	113	150	77.55
Tonge	70	235	572	59.42

Ci-dessous, un exemple illustratif est donné pour l'instance de Jackson (voir le Tableau 5.1). Le graphe représentant les contraintes de précédence avec les temps d'exécution connus sous la forme d'intervalles pour onze opérations est montré dans la Figure 5.1. Les contraintes d'exclusion $E = \{\{1, 4\}, \{1, 8\}, \{3, 5\}, \{3, 11\}, \{6, 7\}, \{7, 8\}, \{7, 9\}, \{7, 11\}, \{8, 10\}, \{10, 11\}\}$ sont utilisées et $\theta = 0.5$. La Figure 5.2 montre le processus de séparation de RLA, où les nœuds représentent seulement la charge des stations pour

stations. Ainsi, le processus de séparation d'un niveau à un autre se réalise rapidement.

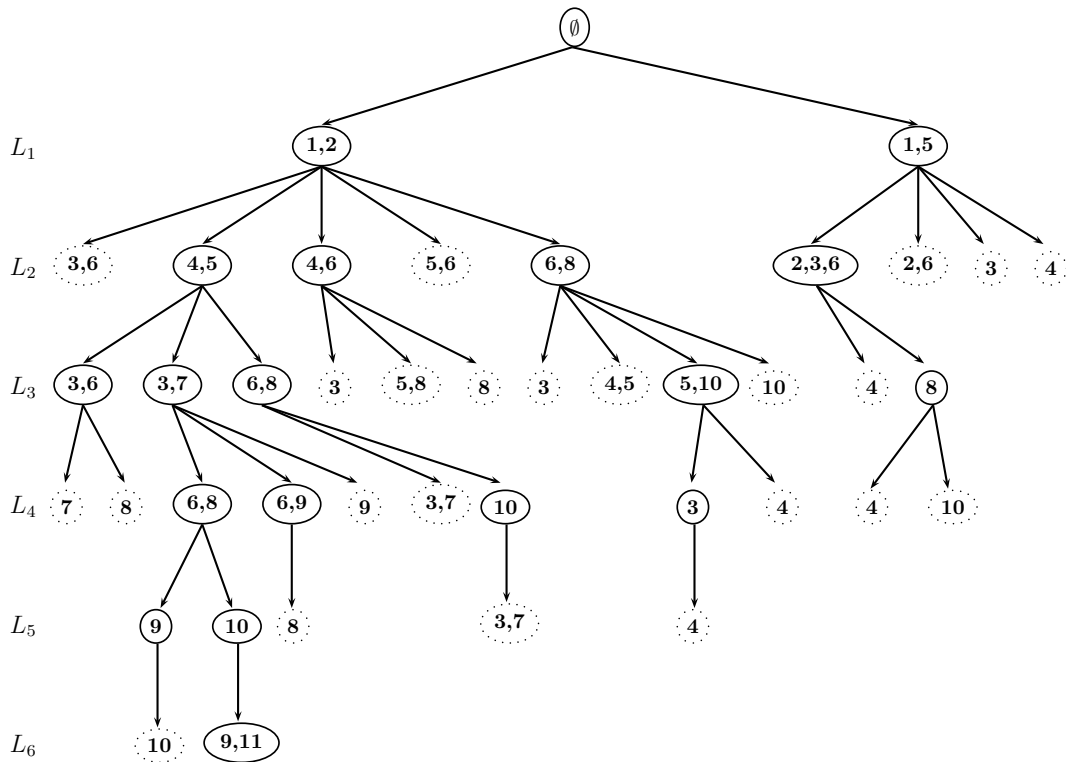


FIGURE 5.2 – Arbre énumératif de RLA pour l’instance de Jackson.

La Figure 5.5 montre le nombre moyen de stations pour les instances de petite, moyenne, et grande taille, en fonction de θ . Pour étudier l’effet de θ , l’indicateur

$$Q = \text{moy}_{i \in \mathcal{I}_{10}} \frac{w_i - w_{i-1}}{w_{i-1}}$$

est introduit. Ici, $\mathcal{I}_n = \{1, 2, \dots, n\}$, w_i est le nombre de stations dans la solution optimale trouvée pour $\theta = \frac{i}{10}$. L’indicateur Q montre l’augmentation relative moyenne du nombre de stations en fonction de θ . En regardant Q , nous concluons qu’il n’existe pas de différence significative (Q est respectivement égal à 0.04, 0.04, et 0.03), malgré le fait que les graphes correspondants aux instances de petite et moyenne tailles sont plus plats que pour les instances de grande taille. Cela vient du fait que le modèle proposé est robuste par rapport au paramètre θ et à la taille des instances du problème.

5.5 Conclusion

Dans ce chapitre, un modèle robuste a été proposé pour le SALBP-1. Pour trouver les solutions optimales, un algorithme par séparation et évaluation a été développé. Cet

algorithme a été évalué sur des instances connues dans la littérature.

Une particularité du modèle proposé réside dans l'utilisation d'un paramètre $\theta \in [0, 1]$ qui introduit une flexibilité permettant de se positionner entre le cas optimiste ($\theta = 0$) et le cas pessimiste ($\theta = 1$) de la réalisation des temps d'exécution des opérations. Ceci dit, décider quelle valeur de θ il faut utiliser pour chaque cas pratique reste un problème important et non résolu. Cela peut être fait pendant le stade préliminaire de conception de lignes d'assemblage avec l'aide de la simulation ou en se basant sur l'expérience du concepteur. Une autre question pour des recherches futures est l'adaptation du modèle robuste proposé pour des lignes d'assemblage avec une structure plus complexe et pour les lignes d'usinage.

Notons aussi que le modèle robuste introduit pour le SALBP-1 peut être facilement généralisé pour le cas où il y a un paramètre propre θ pour chaque station. Cette généralisation peut être appliquée au cas où le concepteur serait capable de distinguer précisément les stations avec des niveaux différents de variabilité des temps d'exécution des opérations.

Les résultats de ce chapitre ont donné lieu à une communication à la conférence ROADEF 2011 [Gurevsky *et al.*, 2011] et à un article soumis à la revue « Journal of the Operational Research Society » [Gurevsky *et al.*, xxxxc].

TABLEAU 5.2 – Le temps de résolution (sec) pour les instances de taille moyenne

Nom	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Buxey	0.06	0.08	0.14	0.14	0.08	0.11	0.11	0.09	0.08	0.08	0.08
Gunther	0.02	0.02	0.06	0.03	0.03	0.02	0.03	0.02	0.02	0.02	0.02
Heskia	27.01	26.89	1437.7	682.14	209.53	44.58	64.5	5.8	4.63	4.81	2.47
Mitchell	0	0	0	0	0	0	0	0.02	0	0	0
Lutz1	0	0	0	0	0.02	0	0	0	0	0	0
Roszieg	0	0	0	0	0	0	0	0	0	0	0
Sawyer	0.25	0.25	0.38	0.39	0.23	0.25	0.27	0.27	0.31	0.33	0.2

TABLEAU 5.3 – Le temps de résolution (sec) pour les instances de grande taille

Nom	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Hahn	0.02	0	0	0	0.02	0.02	0	0.02	0.02	0	0.02
Lutz2	21.36	21.53	14.86	37.45	116.25	3.8	3.91	7.44	7.73	7.69	2.78
Lutz3	0.52	0.53	2.34	17.86	3.16	0.75	0.74	1.58	1.88	2.64	1.30
Tonge	0.27	0.25	0.97	1.06	1.80	3.22	4.42	3.52	3.17	3.38	4.02

TABLEAU 5.4 – Évolution du nombre de stations pour les instances de petite taille

Nom	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Bowman	4	4	4	4	4	5	5	5	5	5	5
Jackson	5	5	5	5	5	6	6	6	6	6	8
Jaeschke	4	4	4	4	5	6	6	6	6	6	7
Mansoor	3	3	3	4	4	5	5	5	5	5	5
Martens	3	3	3	3	3	5	5	5	5	5	5

TABLEAU 5.5 – Évolution du nombre de stations pour les instances de taille moyenne

Nom	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Buxey	9	9	9	10	11	11	11	11	11	11	14
Gunther	9	9	9	10	10	12	12	12	12	12	14
Heskia	7	7	7	7	8	8	8	8	8	8	10
Mitchell	6	6	6	6	7	8	8	8	8	8	8
Lutz1	8	8	9	10	10	10	10	10	10	10	12
Roszieg	7	7	7	8	8	8	8	9	9	9	10
Sawyer	9	9	9	10	10	11	11	11	11	11	14

TABLEAU 5.6 – Évolution du nombre de stations pour les instances de grande taille

Nom	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Hahn	8	8	8	8	8	9	9	9	9	9	11
Lutz2	32	32	32	33	35	42	42	43	43	43	49
Lutz3	17	17	18	18	19	21	21	21	21	21	24
Tonge	22	22	22	22	23	24	24	24	24	24	26

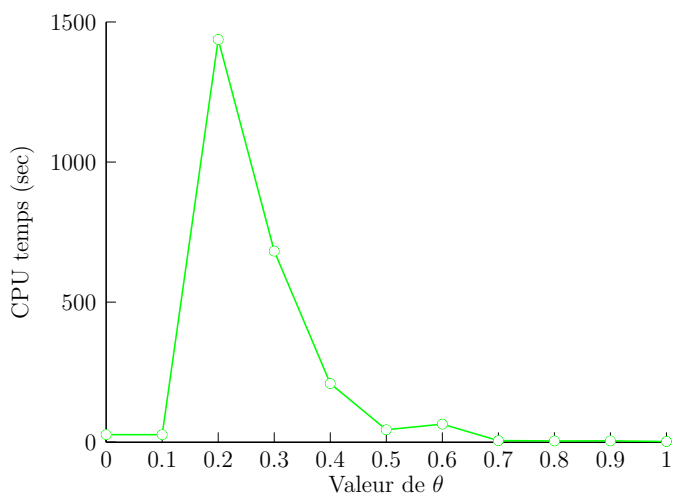


FIGURE 5.3 – Le temps de résolution pour l'instance de Heskia.

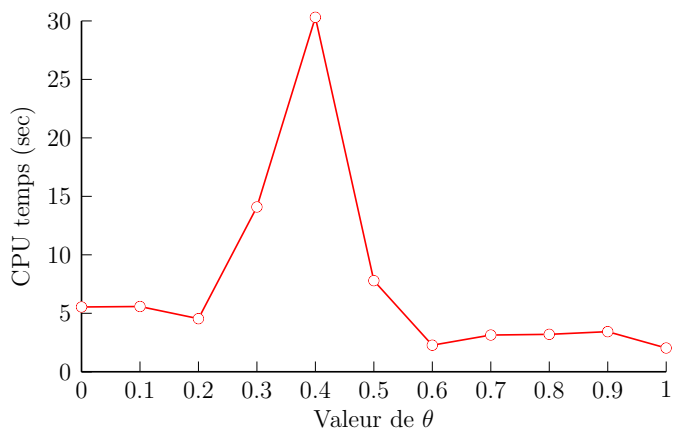


FIGURE 5.4 – Le temps de résolution moyen pour les instances de grande taille.

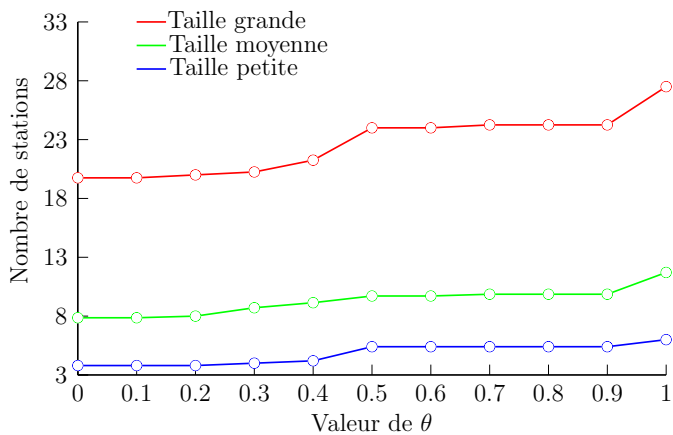


FIGURE 5.5 – Évolution du nombre moyen de stations pour les trois séries.

Conclusion

Dans ce mémoire, nous avons présenté une étude sur la prise en compte de l'incertitude à l'étape de la conception des systèmes de fabrication. Pour les problèmes d'optimisation qui apparaissent à cette étape, souvent les concepteurs ne disposent pas de données précises mais seulement d'estimations. Malheureusement, pendant le fonctionnement du système, certaines données peuvent avoir des valeurs qui sont différentes des estimations qui ont été faites à l'étape de conception au point de remettre en cause la solution fournie. Le but de notre étude était de pallier à cette situation en développant des modèles et des approches permettant de choisir parmi les solutions obtenues celle qui sera la plus stable face aux variations des données. Plus particulièrement, nous avons travaillé sur les problèmes d'équilibrage de charge des lignes d'assemblage et d'usinage sous incertitudes sur les temps opératoires.

Nous avons commencé par analyser les approches existantes dans la littérature pour modéliser l'incertitude des données dans les problèmes d'optimisation combinatoire. Notre objectif était de déterminer les approches qui correspondent le mieux au contexte de la conception des systèmes de fabrication. L'étude effectuée dans le Chapitre 1 nous a orienté vers deux approches : l'approche robuste et l'analyse de sensibilité. Dans le Chapitre 2, lors de la présentation de la problématique de la conception des lignes de fabrication, cette hypothèse a été confirmée. En effet, ces approches sont les plus adaptées au contexte étudié, car elles ne demandent pas aux concepteurs de fournir des lois de probabilité ou des degrés d'appartenance à des ensembles flous pour les données incertaines. La pratique montre que ces informations sont souvent difficiles à obtenir dans le cadre de la conception d'une nouvelle ligne de fabrication. Également, dans le Chapitre 2, nous avons présenté une revue de la littérature concernant la résolution des problèmes d'optimisation qui apparaissent à l'étape de conception des systèmes de fabrication. Cette analyse a montré qu'il y a un besoin de nouveaux modèles pour l'équilibrage des lignes d'assemblage et d'usinage tenant compte de l'incertitude liée aux temps opératoires. Dans ce mémoire, nous avons étudié trois problèmes de ce type. Les résultats obtenus ont été présentés dans les Chapitres 3, 4 et 5.

Dans le Chapitre 3, nous avons étudié le problème de la conception de lignes d'as-

semblage connu dans la littérature sous le nom de SALBP-E. Nous avons trouvé les conditions de stabilité et la formule du rayon de stabilité pour les solutions faisables, quasi-faisables, et optimales de ce problème dans le cas de variations des temps opératoires. Leur complexité de calcul a également été examinée. Nous avons pu proposer des algorithmes polynomiaux pour le calcul du rayon de stabilité des solutions faisables. Notons que les résultats pour les SALBP-1 et SALBP-2 peuvent être directement obtenus à partir des résultats que nous avons présentés pour le SALBP-E.

Dans sa version connue dans la littérature, ce problème vise à maximiser l'efficacité de la ligne en minimisant le nombre de stations de la ligne multiplié par le temps de travail sur la station la plus chargée. Nous avons reformulé ce problème comme un problème bicritère ayant pour le premier critère la minimisation de l'ancienne fonction objectif et pour second la maximisation du rayon de stabilité de la solution. Pour trouver les solutions non-dominées au sens de Pareto, nous avons proposé deux approches complémentaires que nous avons testées sur une série d'exemples numériques. Nous pouvons conclure qu'elles sont suffisamment efficaces pour être appliquées lors de la conception des lignes d'assemblage. Elles aideront les concepteurs à choisir parmi des solutions de bonne qualité celle qui est la plus stable vis-à-vis de variations des temps opératoires.

Dans le Chapitre 4, nous avons étendu cette approche au problème de conception des lignes d'usinage, connu dans la littérature sous le nom de TLBP. Dans un premier temps, nous avons mené l'analyse de sensibilité afin d'évaluer l'impact de changements des temps opératoires sur la performance des solutions faisables, quasi-faisables et optimales. Des algorithmes polynomiaux pour le calcul du rayon de stabilité ont été proposés pour les solutions faisables et quasi-faisables, alors que pour les solutions optimales un algorithme approché de calcul du rayon de stabilité a été développé. Notons que les résultats pour le SALBP-1 peuvent être obtenus comme des conséquences des résultats présentés pour le TLBP.

Comme dans le Chapitre 3, ici aussi, un problème bicritère a été introduit qui vise d'une part à optimiser la fonction objectif et, d'autre part à maximiser le rayon de stabilité. Pour résoudre ce problème, nous avons développé une approche qui se base sur une méthode heuristique et qui cherche des solutions non-dominées au sens de Pareto pour ces deux critères. L'approche proposée a été testée sur des instances connues.

Dans le Chapitre 5, nous avons considéré le problème de type SALBP-1 dans le cas où les temps opératoires sont connus sous la forme d'intervalles des valeurs admissibles. Dans cette situation, la préoccupation n'était plus de trouver une solution optimale, mais une (ou des) meilleure solution(s) parmi celles qui « se comportent correctement » dans toutes les situations possibles de réalisation des temps opératoires. Nous avons présenté un modèle de programmation robuste et développé une méthode par séparation

et évaluation pour sa résolution. L'algorithme proposé a montré de bonnes performances sur des instances recensées dans la littérature. Cependant, il sera certainement moins efficace pour la résolution d'instances de plus grande taille, le problème traité étant \mathcal{NP} -difficile. Par conséquent, le développement d'une méthode heuristique plus adaptée à la résolution de telles instances est nécessaire.

Dans ce mémoire, nous avons traité le problème de conception de lignes de fabrication en présence de temps opératoires soumis à des incertitudes. Cependant, la problématique de la prise en compte de l'incertitude des données lors de la conception de ces lignes ne se limite pas à la considération des variations de ces données numériques. Ainsi, les changements éventuels peuvent intervenir aussi au niveau de différentes contraintes comme les contraintes de précédence ou les contraintes d'exclusion. Jusqu'à présent, cet aspect n'est pas encore bien traité dans la littérature. C'est une piste de nos recherches futures dans ce domaine.

Bibliographie

- [Aissi *et al.*, 2005] H. Aissi, C. Bazgan, and D. Vanderpooten. Complexity of the min-max and min-max regret assignment problems. *Operations Research Letters*, 33(6) :634–640, 2005.
- [Aissi *et al.*, 2011] H. Aissi, M.A. Aloulou, and M.Y. Kovalyov. Minimizing the number of late jobs on a single machine under due date uncertainty. *Journal of Scheduling*, 14(4) :351–360, 2011.
- [Alem and Morabito, 2012] D.J. Alem and R. Morabito. Production planning in furniture settings via robust optimization. *Computers & Operations Research*, 39(2) :139–150, 2012.
- [Aloulou and Della Croce, 2008] M.A. Aloulou and F. Della Croce. Complexity of single machine scheduling problems under scenario-based uncertainty. *Operations Research Letters*, 36(3) :338–342, 2008.
- [Amen, 2000a] M. Amen. An exact method for cost-oriented assembly line balancing. *International Journal of Production Economics*, 64(1-3) :187–195, 2000.
- [Amen, 2000b] M. Amen. Heuristic methods for cost-oriented assembly line balancing : A survey. *International Journal of Production Economics*, 68(1) :1–14, 2000.
- [Amen, 2001] M. Amen. Heuristic methods for cost-oriented assembly line balancing : A comparison on solution quality and computing time. *International Journal of Production Economics*, 69(3) :255–264, 2001.
- [Amen, 2006] M. Amen. Cost-oriented assembly line balancing : Model formulations, solution difficulty, upper and lower bounds. *European Journal of Operational Research*, 168(3) :747–770, 2006.
- [Andrés *et al.*, 2008] C. Andrés, C. Miralles, and R. Pastor. Balancing and scheduling tasks in assembly lines with sequence-dependent setup times. *European Journal of Operational Research*, 187(3) :1212–1223, 2008.
- [Arcus, 1965] A.L. Arcus. A computer method of sequencing operations for assembly lines. *International Journal of Production Research*, 4(4) :259–277, 1965.

- [Askin and Standridge, 1993] G.R. Askin and C.R. Standridge. *Modeling and Analysis of Manufacturing Systems*. John Wiley & Sons, 1993.
- [Ağpak and Gökçen, 2007] K. Ağpak and H. Gökçen. A chance-constrained approach to stochastic line balancing problem. *European Journal of Operational Research*, 180(3) :1098–1115, 2007.
- [Averbakh and Lebedev, 2004] I. Averbakh and V. Lebedev. Interval data minmax regret network optimization problems. *Discrete Applied Mathematics*, 138(3) :289–301, 2004.
- [Averbakh, 2004] I. Averbakh. Minmax regret linear resource allocation problems. *Operations Research Letters*, 32(2) :174–180, 2004.
- [Balinski, 1985] M.L. Balinski. Signature methods for the assignment problem. *Operations Research*, 33(3) :527–536, 1985.
- [Bard, 1989] J.F. Bard. Assembly line balancing with parallel workstations and dead time. *International Journal of Production Research*, 27(6) :1005–1018, 1989.
- [Bautista and Pereira, 2002] J. Bautista and J. Pereira. Ant algorithms for assembly line balancing. In *Ant Algorithms*, volume 2463 of *Lecture Notes in Computer Science*, pages 49–61. Springer Berlin/Heidelberg, 2002.
- [Bautista and Pereira, 2007] J. Bautista and J. Pereira. Ant algorithms for a time and space constrained assembly line balancing problem. *European Journal of Operational Research*, 177(3) :2016–2032, 2007.
- [Baybars, 1986] I. Baybars. A survey of exact algorithms for the simple assembly line balancing problem. *Management Science*, 32(8) :909–932, 1986.
- [Baykasoglu and Dereli, 2008] A. Baykasoglu and T. Dereli. Two-sided assembly line balancing using an ant-colony-based heuristic. *The International Journal of Advanced Manufacturing Technology*, 36(5-6) :582–588, 2008.
- [Baykasoglu, 2006] A. Baykasoglu. Multi-rule multi-objective simulated annealing algorithm for straight and U type assembly line balancing problems. *Journal of Intelligent Manufacturing*, 17(2) :217–232, 2006.
- [Baykasoglu and Özbakır, 2007] A. Baykasoglu and L. Özbakır. Stochastic U-line balancing using genetic algorithms. *The International Journal of Advanced Manufacturing Technology*, 32(1-2) :139–147, 2007.
- [Becker and Scholl, 2006] C. Becker and A. Scholl. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research*, 168(3) :694–715, 2006.
- [Belgacem and Hifi, 2008a] T. Belgacem and M. Hifi. Sensitivity analysis of the knapsack problem : Tighter lower and upper bound limits. *Journal of Systems Science and Systems Engineering*, 17(2) :156–170, 2008.

-
- [Belgacem and Hifi, 2008b] T. Belgacem and M. Hifi. Sensitivity analysis of the knapsack sharing problem : perturbation of the profit of an item. *International Transactions in Operational Research*, 15(1) :35–49, 2008.
- [Belgacem and Hifi, 2008c] T. Belgacem and M. Hifi. Sensitivity analysis of the knapsack sharing problem : Perturbation of the weight of an item. *Computers & Industrial Engineering*, 35(1) :295–308, 2008.
- [Belgacem and Hifi, 2008d] T. Belgacem and M. Hifi. Sensitivity analysis of the optimum to perturbation of the profit of a subset of items in the binary knapsack problem. *Discrete Optimization*, 5(4) :755–761, 2008.
- [Belmokhtar *et al.*, 2006] S. Belmokhtar, A. Dolgui, N. Guschinsky, and G. Levin. Integer programming models for logical layout design of modular machining lines. *Computers & Industrial Engineering*, 51(3) :502–518, 2006.
- [Bertsimas and Sim, 2003] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming*, 98(1-3) :49–71, 2003.
- [Bertsimas and Thiele, 2006] D. Bertsimas and A. Thiele. A robust optimization approach to inventory theory. *Operations Research*, 54(1) :150–168, 2006.
- [Billaut *et al.*, 2005] J.-C. Billaut, A. Moukrim, and E. Sanlaville, editors. *Flexibilité et robustesse en ordonnancement*. Lavoisier, 2005.
- [Boctor, 1995] F.F. Boctor. A multiple-rule heuristic for assembly line balancing. *Journal of the Operational Research Society*, 46 :62–69, 1995.
- [Bräsel *et al.*, 1996] H. Bräsel, Yu.N. Sotskov, and F. Werner. Stability of a schedule minimizing mean flow time. *Mathematical and Computer Modelling*, 24(10) :39–53, 1996.
- [Buxey, 1974] G.M. Buxey. Assembly line balancing with multiple stations. *Management Science*, 20(6) :1010–1021, 1974.
- [Chakravarti and Wagelmans, 1998] N. Chakravarti and A.P.M. Wagelmans. Calculation of stability radii for combinatorial optimization problems. *Operations Research Letters*, 23(1-2) :1–7, 1998.
- [Chiang and Urban, 2006] W.-C. Chiang and T.L. Urban. The stochastic U-line balancing problem : A heuristic procedure. *European Journal of Operational Research*, 175(3) :1767–1781, 2006.
- [Chiang, 1998] W.-C. Chiang. The application of a tabu search metaheuristic to the assembly line balancing problem. *Annals of Operations Research*, 77(0) :209–227, 1998.
- [Dantzig, 1957] G.B. Dantzig. Discrete variable extremum problems. *Operations Research*, 5(2) :266–288, 1957.

- [Davenport and Beck, 2000] A.J. Davenport and J.C. Beck. A survey of techniques for scheduling with uncertainty, 2000.
- [Deckro, 1989] R.F. Deckro. Balancing cycle time and workstations. *IIE Transactions*, 21(2) :106–111, 1989.
- [Delorme *et al.*, 2012] X. Delorme, A. Dolgui, and M.Y. Kovalyov. Combinatorial design of a minimum cost transfer line. *Omega*, 40(1) :31–41, 2012.
- [Dijkstra, 1959] E.W. Dijkstra. A note on two problems in connection with graphs. *Numerische Mathematik*, 1 :269–271, 1959.
- [Dolgui and Ihnatsenka, 2009] A. Dolgui and I. Ihnatsenka. Branch and bound algorithm for a transfer line design problem : Stations with sequentially activated multi-spindle heads. *European Journal of Operational Research*, 197(3) :1119–1132, 2009.
- [Dolgui and Proth, 2006] A. Dolgui and J.-M. Proth. *Les systèmes de production modernes*. Lavoisier, 2006.
- [Dolgui *et al.*, 2005] A. Dolgui, B. Finel, F. Vernadat, N. Guschinsky, and G. Levin. A heuristic approach for transfer lines balancing. *Journal of Intelligent Manufacturing*, 16(2) :159–172, 2005.
- [Dolgui *et al.*, 2006a] A. Dolgui, B. Finel, O. Guschinskaya, N. Guschinsky, G. Levin, and F. Vernadat. Balancing large-scale machining lines with multi-spindle heads using decomposition. *International Journal of Production Research*, 44(18) :4105–4120, 2006.
- [Dolgui *et al.*, 2006b] A. Dolgui, B. Finel, N.N. Guschinsky, G.M. Levin, and F.B. Vernadat. MIP approach to balancing transfer lines with blocks of parallel operations. *IIE Transactions*, 38(10) :869–882, 2006.
- [Dolgui *et al.*, 2006c] A. Dolgui, N. Guschinsky, and G. Levin. A decomposition method for transfer line life cycle cost optimisation. *Journal of Mathematical Modelling and Algorithms*, 5(2) :215–238, 2006.
- [Dolgui *et al.*, 2008] A. Dolgui, N. Guschinsky, G. Levin, and J.-M. Proth. Optimisation of multi-position machines and transfer lines. *European Journal of Operational Research*, 185(3) :1375–1389, 2008.
- [Easton, 1990] F.F. Easton. A dynamic program with fathoming and dynamic upper bounds for the assembly line balancing problem. *Computers & Operations Research*, 17(2) :163–175, 1990.
- [Ehrgott, 2005] M. Ehrgott. *Multicriteria Optimization*. Springer Berlin/Heidelberg, 2 edition, 2005.
- [Emelichev and Gurevskii, 2009] V.A. Emelichev and E.E. Gurevskii. On the regularization of vector integer quadratic programming problems. *Cybernetics and Systems Analysis*, 45(2) :274–280, 2009.

-
- [Emelichev *et al.*, 2002] V.A. Emelichev, E. Girlich, Yu.V. Nikulin, and D.P. Podkopaev. Stability and regularization of vector problems of integer linear programming. *Optimization*, 51(4) :645–676, 2002.
- [Erel and Sarin, 1998] E. Erel and S.C. Sarin. A survey of the assembly line balancing procedures. *Production Planning & Control*, 9(5) :414–434, 1998.
- [Essafi *et al.*, 2010a] M. Essafi, X. Delorme, and A. Dolgui. Balancing lines with CNC machines : A multi-start ant based heuristic. *CIRP Journal of Manufacturing Science and Technology*, 2(3) :176–182, 2010.
- [Essafi *et al.*, 2010b] M. Essafi, X. Delorme, A. Dolgui, and O. Guschinskaya. A MIP approach for balancing transfer line with complex industrial constraints. *Computers & Industrial Engineering*, 3(3) :393–400, 2010.
- [Garey and Johnson, 1979] M.R. Garey and D.S. Johnson. *Computers and Intractability : A Guide to the NP-Completeness*. W.H. Freeman and Company, 1979.
- [Gen *et al.*, 1996] M. Gen, Y. Tsujimura, and Y. Li. Fuzzy assembly line balancing using genetic algorithms. *Computers & Industrial Engineering*, 31(3-4) :631–634, 1996.
- [Geoffrion and Nauss, 1977] A.M. Geoffrion and R. Nauss. Parametric and postoptimality analysis in integer linear programming. *Management Science*, 23(5) :453–466, 1977.
- [Ghosh and Gagnon, 1989] S. Ghosh and R.J. Gagnon. A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems. *International Journal of Production Research*, 27(4) :637–670, 1989.
- [Gordeev and Leont’ev, 1980] E.N. Gordeev and V.K. Leont’ev. Stability in bottleneck problems. *USSR Computational Mathematics and Mathematical Physics*, 20(4) :275–280, 1980.
- [Gordeev *et al.*, 1983] E.N. Gordeev, V.K. Leont’ev, and I.Kh. Sigal. Computational algorithms for finding the radius of stability in problems of choice. *USSR Computational Mathematics and Mathematical Physics*, 23(4) :128–132, 1983.
- [Gordeev, 1989] E.N. Gordeev. Solution stability in a shortest path problem. *Discrete Mathematics (in Russian)*, 1(3) :45–56, 1989.
- [Gourgand *et al.*, 2007] M. Gourgand, N. Grangeon, and S. Norre. Metaheuristics based on bin packing for the line balancing problem. *RAIRO Operations Research*, 41(2) :193–211, 2007.
- [Graham *et al.*, 1979] R.L. Graham, E.L. Lawler, E.L. Lenstra, and A.H.G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling : A survey. *Annals of Discrete Mathematics*, 4 :287–326, 1979.

- [Greenberg, 1998] H.J. Greenberg. An annotated bibliography for post-solution analysis in mixed integer and combinatorial optimization. In D.L. Woodruff, editor, *Advances in Computational and Stochastic Optimization, Logic Programming, and Heuristic Search : Interfaces in Computer Science and Operations Research*, Operations Research/Computer Science Interfaces Series, pages 97–148. Kluwer Academic Publishers, 1998.
- [Gupta and Rosenhead, 1968] S.K. Gupta and J. Rosenhead. Robustness in sequential investment decisions. *Management Science*, 15(2) :B18–B29, 1968.
- [Gurevsky *et al.*, 2009] E. Gurevsky, O. Guschinskaya, and A. Dolgui. Qualitative stability analysis of an optimal balance for an assembly line with fixed stations number. In *Proceedings of the 14th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'2009), Palma de Mallorca (Spain), September 22–26. 6 pages (CD-ROM)*, 2009.
- [Gurevsky *et al.*, 2010a] E. Gurevsky, O. Guschinskaya, and A. Dolgui. Analyse de sensibilité du problème de l'équilibrage des lignes d'usinage : perturbation des temps opératoires. In *Actes de la 8ème ENIM IFAC Conférence Internationale de Modélisation et Simulation (MOSIM'10). Hammamet (Tunisie), 10–12 Mai, 2010. 8 pages (USB flash)*, 2010.
- [Gurevsky *et al.*, 2010b] E. Gurevsky, O. Guschinskaya, and A. Dolgui. Analyse de sensibilité pour le problème d'équilibrage des lignes d'assemblage de type SALBP-2. In *Résumés du 11e 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 (USB flash)*, 2010.
- [Gurevsky *et al.*, 2010c] E. Gurevsky, O. Guschinskaya, and A. Dolgui. Assembly line balancing problems under uncertainties. In *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 (CD-ROM)*, 2010.
- [Gurevsky *et al.*, 2011] E. Gurevsky, Ö. Hazır, and A. Dolgui. Équilibrage robuste flexible de SALBP-1 avec les données incertaines. In *Résumés du 12e Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2011). Saint-Étienne (France), 2–4 Mars, 2011. 2 pages (USB flash)*, 2011.
- [Gurevsky *et al.*, xxxxa] E. Gurevsky, O. Battaïa, and A. Dolgui. Balancing of simple assembly lines under variations of task processing times. *Annals of Operations Research (soumis)*, xxxx.
- [Gurevsky *et al.*, xxxxb] E. Gurevsky, O. Battaïa, and A. Dolgui. Measure of stability for

-
- a generalized assembly line balancing problem. *Discrete Applied Mathematics (soumis)*, xxxx.
- [Gurevsky *et al.*, xxxxc] E. Gurevsky, Ö. Hazır, O. Battaïa, and A. Dolgui. Robust balancing of simple assembly lines with interval task processing times. *Journal of the Operational Research Society (soumis)*, xxxx.
- [Guschinskaya and Dolgui, 2009] O. Guschinskaya and A. Dolgui. Comparison of exact and heuristic methods for a transfer line balancing problem. *International Journal of Production Economics*, 120(2) :276–286, 2009.
- [Guschinskaya and Dolgui, 2010] O. Guschinskaya and A. Dolgui. Équilibrage de lignes de production : état de l’art. *Journal Européen des Systèmes Automatisés*, 44(9-10) :1081–1119, 2010.
- [Guschinskaya *et al.*, 2008] O. Guschinskaya, A. Dolgui, N. Guschinsky, and G. Levin. A heuristic multi-start decomposition approach for optimal design of serial machining lines. *European Journal of Operational Research*, 189(3) :902–913, 2008.
- [Guschinskaya *et al.*, 2011] O. Guschinskaya, E. Gurevsky, A. Dolgui, and A. Ereemeev. Metaheuristic approaches for the design of machining lines. *The International Journal of Advanced Manufacturing Technology*, 55(1-4) :11–22, 2011.
- [Hall and Posner, 2004] N.G. Hall and M.E. Posner. Sensitivity analysis for scheduling problems. *Journal of Scheduling*, 7(1) :49–83, 2004.
- [Hazır *et al.*, 2011] Ö. Hazır, E. Erel, and Y. Günalay. Robust optimization models for the discrete timecost trade-off problem. *International Journal of Production Economics*, 130(1) :87–95, 2011.
- [Held *et al.*, 1963] M. Held, R.M. Karp, and R. Shareshian. Assembly-line balancing – dynamic programming with precedence constraints. *Operations Research*, 11(3) :442–459, 1963.
- [Helgeson and Birnie, 1961] W.P. Helgeson and D.P. Birnie. Assembly line balancing using the ranked positional weight technique. *Journal of Industrial Engineering*, 12(6) :394–398, 1961.
- [Hifi *et al.*, 2005] M. Hifi, H. Mhalla, and S. Sadfi. Sensitivity of the optimum to perturbations of the profit or weight of an item in the binary knapsack problem. *Journal of Combinatorial Optimization*, 10(3) :239–260, 2005.
- [Hifi *et al.*, 2008] M. Hifi, H. Mhalla, and S. Sadfi. An adaptive algorithm for the knapsack problem : perturbation of the profit or weight of an arbitrary item. *European Journal of Industrial Engineering*, 2(2) :134–152, 2008.

- [Hoffmann, 1992] T.R. Hoffmann. Eureka : A hybrid system for assembly line balancing. *Management Science*, 38(1) :39–47, 1992.
- [Hop, 2006] N.V. Hop. A heuristic solution for fuzzy mixed-model line balancing problem. *European Journal of Operational Research*, 168(3) :798–810, 2006.
- [Ignall, 1965] E.J. Ignall. A review of assembly line balancing. *The Journal of Industrial Engineering*, 16 :244–254, 1965.
- [Jackson, 1956] J.R. Jackson. A computing procedure for a line balancing problem. *Management Science*, 2(3) :261–271, 1956.
- [Johnson, 1988] R.V. Johnson. Optimally balancing large assembly lines with "Fable". *Management Science*, 34(2) :240–253, 1988.
- [Karaşan *et al.*, 2001] O.E. Karaşan, M.Ç. Pınar, and H. Yaman. The robust shortest path problem with interval data. Technical report, Industrial Engineering Department, Bilkent University, 2001.
- [Kasperski, 2005] A. Kasperski. Minimizing maximal regret in the single machine sequencing problem with maximum lateness criterion. *Operations Research Letters*, 33(4) :431–436, 2005.
- [Kim *et al.*, 2009] Y.K. Kim, W.S. Song, and J.H. Kim. A mathematical model and a genetic algorithm for two-sided assembly line balancing. *Computers & Operations Research*, 36(3) :853–865, 2009.
- [Kouvelis and Yu, 1997] P. Kouvelis and G. Yu. *Robust discrete optimization and its applications*. Kluwer, 1997.
- [Kravchenko *et al.*, 1995] S.A. Kravchenko, Yu.N. Sotskov, and F. Werner. Optimal schedules with infinitely large stability radius. *Optimization*, 33(3) :271–280, 1995.
- [Kruskal, 1976] J.B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, 7(1) :48–50, 1976.
- [Lapierre *et al.*, 2006] S.D. Lapierre, A. Ruiz, and P. Soriano. Balancing assembly lines with tabu search. *European Journal of Operational Research*, 168(3) :826–837, 2006.
- [Lawler, 1973] E.L. Lawler. Optimal sequencing of a single machine subject to precedence constraints. *Management Science*, 19(5) :544–546, 1973.
- [Lebedev and Averbakh, 2006] V. Lebedev and I. Averbakh. Complexity of minimizing the total flow time with interval data and minmax regret criterion. *Discrete Applied Mathematics*, 154(15) :2167–2177, 2006.
- [Lebedeva *et al.*, 2005] T.T. Lebedeva, N.V. Semenova, and T.I. Sergienko. Stability of vector problems of integer optimization : Relationship with the stability of sets of

-
- optimal and nonoptimal solutions. *Cybernetics and Systems Analysis*, 41(4) :551–558, 2005.
- [Leont'ev, 1975] V.K. Leont'ev. Stability of the travelling salesman problem. *USSR Computational Mathematics and Mathematical Physics*, 15(5) :199–213, 1975.
- [Leontev, 1979] V.K. Leontev. Stability in linear discrete optimization problems. *Cybernetics Problems (in Russian)*, 35 :169–184, 1979.
- [Levitin *et al.*, 2006] G. Levitin, J. Rubinovitz, and B. Shnits. A genetic algorithm for robotic assembly line balancing. *European Journal of Operational Research*, 168(3) :811–825, 2006.
- [Libura *et al.*, 1998] M. Libura, E.S. van der Poort, G. Sierksma, and J.A.A. van der Veen. Stability aspects of the traveling salesman problem based on k -best solutions. *Discrete Applied Mathematics*, 87(1-3) :159–185, 1998.
- [Liu *et al.*, 2008] S.B. Liu, K.M. Ng, and H.L. Ong. Branch-and-bound algorithms for simple assembly line balancing problem. *The International Journal of Advanced Manufacturing Technology*, 36(1-2) :169–177, 2008.
- [Mahjoub, 2004] A. Mahjoub. *Étude de la robustesse des algorithmes pour des problèmes d'ordonnancement et de localisation*. PhD thesis, Institut National Polytechnique de Grenoble, 2004.
- [McMullen and Frazier, 1998] P.R. McMullen and G.V. Frazier. Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel workstations. *International Journal of Production Research*, 36(10) :2717–2741, 1998.
- [McMullen and Tarasewich, 2003] P.R. McMullen and P. Tarasewich. Using ant techniques to solve the assembly line balancing problem. *IIE Transactions*, 35(7) :605–617, 2003.
- [McMullen and Tarasewich, 2006] P.R. McMullen and P. Tarasewich. Multi-objective assembly line balancing via a modified ant colony optimization technique. *International Journal of Production Research*, 44(1) :27–42, 2006.
- [Miralles *et al.*, 2008] C. Miralles, J.P. García-Sabater, C. Andrés, and M. Cardós. Branch and bound procedures for solving the assembly line worker assignment and balancing problem : Application to sheltered work centres for disabled. *Discrete Applied Mathematics*, 156(3) :352–367, 2008.
- [Montemanni and Gambardella, 2005] R. Montemanni and L.M. Gambardella. A branch and bound algorithm for the robust spanning tree problem with interval data. *European Journal of Operational Research*, 161(3) :771–779, 2005.

- [Montemanni *et al.*, 2004] R. Montemanni, L.M. Gambardella, and A.V. Donati. A branch and bound algorithm for the robust shortest path problem with interval data. *Operations Research Letters*, 32(3) :225–232, 2004.
- [Moon and Yao, 2011] Y. Moon and T. Yao. A robust mean absolute deviation model for portfolio optimization. *Computers & Operations Research*, 38(9) :1251–1258, 2011.
- [Moore, 1968] J.M. Moore. An n job, one machine sequencing algorithm for minimizing the number of late jobs. *Management Science*, 15(1) :102–109, 1968.
- [Özcan and Toklu, 2009] U. Özcan and B. Toklu. A tabu search algorithm for two-sided assembly line balancing. *The International Journal of Advanced Manufacturing Technology*, 43(7-8) :822–829, 2009.
- [Özcan, 2010] U. Özcan. 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, 2010.
- [Pastor *et al.*, 2002] R. Pastor, C. Andrés, A. Duran, and M. Pérez. Tabu search algorithms for an industrial multi-product and multi-objective assembly line balancing problem, with reduction of the task dispersion. *Journal of the Operational Research Society*, 53 :1317–1323, 2002.
- [Pettie, 2005] S. Pettie. Sensitivity analysis of minimum spanning trees in sub-inverse-Ackermann time. In *Algorithms and Computation*, volume 3827 of *Lecture Notes in Computer Science*, pages 964–973. Springer Berlin / Heidelberg, 2005.
- [Piper and Zoltners, 1975] C.J. Piper and A.A. Zoltners. Implicit enumeration based algorithms for postoptimizing zero-one programs. *Naval Research Logistics Quarterly*, 22(4) :791–809, 1975.
- [Piper and Zoltners, 1976] C.J. Piper and A.A. Zoltners. Some easy postoptimality analysis for zero-one programming. *Management Science*, 22(7) :759–765, 1976.
- [Prim, 1957] R.C. Prim. Shortest connection networks and some generalizations. *Bell System Technical Journal*, 36 :1389–1401, 1957.
- [Ramaswamy *et al.*, 2000] R. Ramaswamy, N. Chakravarti, and D. Ghosh. Complexity of determining exact tolerances for min-max combinatorial optimization problems. Technical report, University of Groningen, Groningen, The Netherlands, 2000.
- [Ramaswamy *et al.*, 2005] R. Ramaswamy, J.B. Orlin, and N. Chakravarti. Sensitivity analysis for shortest path problems and maximum capacity path problems in undirected graphs. *Mathematical Programming*, 102(2) :355–369, 2005.
- [Raouf and Tsui, 1982] A. Raouf and C.L. Tsui. A new method for assembly line balancing having stochastic work elements. *Computers & Industrial Engineering*, 6(2) :131–148, 1982.

-
- [Rekiek and Delchambre, 2006] B. Rekiek and A. Delchambre. *Assembly Line Design : The Balancing of Mixed-Model Hybrid Assembly Lines with Genetic Algorithms*. Springer, 2006.
- [Rekiek *et al.*, 2001] B. Rekiek, P. De Lit, F. Pellichero, T. L'Eglise, P. Fouda, E. Falkenauer, and A. Delchambre. A multiple objective grouping genetic algorithm for assembly line design. *Journal of Intelligent Manufacturing*, 12(5-6) :467–485, 2001.
- [Rekiek *et al.*, 2002] B. Rekiek, A. Dolgui, A. Delchambre, and A. Bratcu. State of art of optimization methods for assembly line design. *Annual Reviews in Control*, 26(2) :163–174, 2002.
- [Roodman, 1972] G.M. Roodman. Postoptimality analysis in zero-one programming by implicit enumeration. *Naval Research Logistics Quarterly*, 19(3) :435–447, 1972.
- [Roodman, 1974] G.M. Roodman. Postoptimality analysis in integer programming by implicit enumeration : The mixed integer case. *Naval Research Logistics Quarterly*, 21(4) :595–607, 1974.
- [Rosenblatt and Carlson, 1985] M.J. Rosenblatt and R.C. Carlson. Designing a production line to maximize profit. *IIE Transactions*, 17(2) :117–122, 1985.
- [Rossi, 2003] A. Rossi. *Ordonnancement en milieu incertain, mise en œuvre d'une démarche robuste*. PhD thesis, Institut National Polytechnique de Grenoble, 2003.
- [Rubinovitz and Levitin, 1995] J. Rubinovitz and G. Levitin. Genetic algorithm for assembly line balancing. *International Journal of Production Economics*, 41(1-3) :343–354, 1995.
- [Sabuncuoglu *et al.*, 2000] I. Sabuncuoglu, E. Erel, and M. Tanyer. Assembly line balancing using genetic algorithms. *Journal of Intelligent Manufacturing*, 11(3) :295–310, 2000.
- [Sabuncuoglu *et al.*, 2009] I. Sabuncuoglu, E. Erel, and A. Alp. Ant colony optimization for the single model U-type assembly line balancing problem. *International Journal of Production Economics*, 120(2) :287–300, 2009.
- [Salveson, 1955] M.E. Salveson. The assembly line balancing problem. *Journal of Industrial Engineering*, 6(3) :18–25, 1955.
- [Sanlaville, 2005] E. Sanlaville. *Ordonnancement sous conditions changeantes : Comment prendre en compte les variations, aléas, incertitudes sur les données ?* HDR, Université Blaise Pascal, 2005.
- [Scholl and Becker, 2006] A. Scholl and C. Becker. State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research*, 168(3) :666–693, 2006.

- [Scholl and Klein, 1997] A. Scholl and R. Klein. SALOME : A bidirectional branch-and-bound procedure for assembly line balancing. *INFORMS Journal on Computing*, 9(4) :319–334, 1997.
- [Scholl *et al.*, 2008] A. Scholl, N. Boysen, and M. Fliedner. The sequence-dependent assembly line balancing problem. *OR Spectrum*, 30(3) :579–609, 2008.
- [Scholl, 1999] A. Scholl. *Balancing and Sequencing of Assembly Lines*. Physica-Verlag Heidelberg, 2 edition, 1999.
- [Shier and Witzgall, 1980] D.R. Shier and C. Witzgall. Arc tolerance in shortest path and network flow problems. *Networks*, 10(4) :277–291, 1980.
- [Simaria and Vilarinho, 2004] A.S. Simaria and P.M. Vilarinho. A genetic algorithm based approach to the mixed-model assembly line balancing problem of type II. *Computers & Industrial Engineering*, 47(4) :391–407, 2004.
- [Simaria and Vilarinho, 2009] A.S. Simaria and P.M. Vilarinho. 2-ANTBAL : An ant colony optimisation algorithm for balancing two-sided assembly lines. *Computers & Industrial Engineering*, 56(2) :489–506, 2009.
- [Smith, 1956] W.E. Smith. Various optimizers for single-stage production. *Naval Research Logistic Quarterly*, 3(1-2) :59–66, 1956.
- [Sotskov *et al.*, 1997] Y. Sotskov, N.Y. Sotskova, and F. Werner. Stability of an optimal schedule in a job shop. *Omega*, 25(4) :397–414, 1997.
- [Sotskov *et al.*, 1998a] Y.N. Sotskov, V.S. Tanaev, and F. Werner. Stability radius of an optimal schedule : a survey and recent developments. In G. Yu, editor, *Industrial Applications of Combinatorial Optimization*, volume 16 of *Applied Optimization*, pages 72–108. Kluwer Academic Publishers, 1998.
- [Sotskov *et al.*, 1998b] Y.N. Sotskov, A.P.M. Wagelmans, and F. Werner. On the calculation of the stability radius of an optimal or an approximate schedule. *Annals of Operations Research*, 83(0) :213–252, 1998.
- [Sotskov *et al.*, 2006] Y.N. Sotskov, A. Dolgui, and M.-C. Portmann. Stability analysis of an optimal balance for an assembly line with fixed cycle time. *European Journal of Operational Research*, 168(3) :783–797, 2006.
- [Sotskov *et al.*, 2010] Yu.N. Sotskov, N.Yu. Sotskova, T.-C. Lai, and F. Werner. *Scheduling under Uncertainty : Theory and Algorithms*. Belorusskaya nauka, Minsk, 2010.
- [Sotskov, 1991] Y.N. Sotskov. Stability of an optimal schedule. *European Journal of Operational Research*, 55(1) :91–102, 1991.
- [Suresh *et al.*, 1996] G. Suresh, V.V. Vinod, and S. Sahu. A genetic algorithm for assembly line balancing. *Production Planning & Control*, 7(1) :38–46, 1996.

-
- [Talbot *et al.*, 1986] F.B. Talbot, J.H. Patterson, and W.V. Gehrlein. A comparative evaluation of heuristic line balancing techniques. *Management Science*, 32(4) :430–454, 1986.
- [Tarjan, 1975] R.E. Tarjan. Efficiency of a good but not linear set union algorithm. *Journal of the Association for Computing Machinery*, 22(2) :215–225, 1975.
- [Tarjan, 1982] R.E. Tarjan. Sensitivity analysis of minimum spanning trees and shortest path trees. *Information Processing Letters*, 14 :30–33, 1982.
- [Tasan and Tunali, 2008] S.O. Tasan and S. Tunali. A review of the current applications of genetic algorithms in assembly line balancing. *Journal of Intelligent Manufacturing*, 19(1) :49–69, 2008.
- [Tsujimura *et al.*, 1995] Y. Tsujimura, M. Gen, and E. Kubota. Solving fuzzy assembly-line balancing problem with genetic algorithms. *Computers & Industrial Engineering*, 29(1-4) :543–547, 1995.
- [Urban and Chiang, 2006] T.L. Urban and W.-C. Chiang. An optimal piecewise-linear program for the U-line balancing problem with stochastic task times. *European Journal of Operational Research*, 168(3) :771–782, 2006.
- [van der Poort, 1997] E.S. van der Poort. *Aspects of sensitivity analysis for the traveling salesman problem*. PhD thesis, University of Groningen, Groningen, The Netherlands, 1997.
- [Van Hoesel and Wagelmans, 1999] S. Van Hoesel and A. Wagelmans. On the complexity of postoptimality analysis of 0/1 programs. *Discrete Applied Mathematics*, 91(1-3) :251–263, 1999.
- [Vilarinho and Simaria, 2002] P.M. Vilarinho and A.S. Simaria. A two-stage heuristic method for balancing mixed-model assembly lines with parallel workstations. *International Journal of Production Research*, 40(6) :1405–1420, 2002.
- [Vilarinho and Simaria, 2006] P.M. Vilarinho and A.S. Simaria. ANTBAL : an ant colony optimization algorithm for balancing mixed-model assembly lines with parallel workstations. *International Journal of Production Research*, 44(2) :291–303, 2006.
- [Wu *et al.*, 2008] E.-F. Wu, Y. Jin, J.-S. Bao, and X.-F. Hu. A branch-and-bound algorithm for two-sided assembly line balancing. *The International Journal of Advanced Manufacturing Technology*, 39(9-10) :1009–1015, 2008.
- [Yaman *et al.*, 2001] H. Yaman, O.E. Karaşan, and M.Ç. Pınar. The robust spanning tree problem with interval data. *Operations Research Letters*, 29(1) :31–40, 2001.
- [Yang and Yu, 2002] J. Yang and G. Yu. On the robust single machine scheduling problem. *Journal of Combinatorial Optimization*, 6(1) :17–33, 2002.

- [Yu and Yang, 1998] G. Yu and J. Yang. On the robust shortest path problem. *Computers & Operations Research*, 25(6) :457–468, 1998.
- [Yu, 1996] G. Yu. On the max-min 0-1 knapsack problem with robust optimization applications. *Operations Research*, 44(2) :407–415, 1996.

NNT : 2011 EMSE 0634

Evgeny GUREVSKY

Production line design under uncertainty: sensitivity analysis and robust
approach

Speciality: Industrial Engineering

Keywords: production lines, balancing, uncertainty, sensitivity analysis, robust optimization, Pareto optima, stability radius, heuristics

Abstract:

The presented work deals with the design of production systems in uncertain context. The design of such systems can be interpreted as an optimization problem that consists to find a configuration optimizing certain objectives and respecting technological and economical constraints. The production systems studied in this thesis are the assembly and transfer lines. The first one is the line that can be represented as a flow-oriented chain of workstations where, at each workstation, the tasks are executed in a sequential manner. The second is a particular line that is composed of transfer machines including several multi-spindle heads where the tasks are executed simultaneously.

At first, we describe different approaches that permit to model the uncertainty of data in optimization. A particular attention is attracted to two following approaches: robust approach and sensitivity analysis. Then, we present three applications: the design of assembly and transfer lines under variations of task processing times and the design of an assembly line with interval task processing times. For each application, we identify the expected performances as well as the complexity of taking into account the uncertainty. Thereafter, we propose some new optimization criteria in adequacy with the introduced problematic. Finally, resolution methods are developed to solve different problems engendered by these criteria.

NNT : 2011 EMSE 0634

Evgeny GUREVSKY

Conception de lignes de fabrication sous incertitudes : analyse de
sensibilité et approche robuste

Spécialité : Génie Industriel

Mots clefs : lignes de fabrication, équilibrage, incertitude, analyse de sensibilité, optimisation robuste, optimum de Pareto, rayon de stabilité, heuristiques.

Résumé :

Les travaux présentés dans cette thèse portent sur la conception de systèmes de fabrication en contexte incertain. La conception d'un tel système peut être vue comme un problème d'optimisation qui consiste à trouver une configuration qui permet d'optimiser certains objectifs tout en respectant des contraintes technologiques et économiques connues. Les systèmes de fabrication étudiés dans ce mémoire sont des lignes d'assemblage et d'usinage. La première est une ligne qui se présente comme une chaîne de postes de travail où, dans chaque poste, les opérations d'assemblage s'exécutent de manière séquentielle. La deuxième, quant à elle, est une ligne particulière qui se compose de machines de transfert comportant plusieurs boîtiers multibroches où les opérations s'exécutent simultanément.

Dans un premier temps, nous décrivons de différentes approches permettant de modéliser l'incertitude des données en optimisation. Une attention particulière est portée sur les deux approches suivantes : l'approche robuste et l'analyse de sensibilité. Puis, nous présentons trois applications : la conception d'une ligne d'assemblage et d'une ligne d'usinage soumises aux variations de temps opératoires et la conception d'une ligne d'assemblage avec les temps opératoires connus sous la forme d'intervalles des valeurs possibles. Pour chaque application, nous identifions les performances attendues ainsi que la complexité de la prise en compte de l'incertitude. Ensuite, nous proposons de nouveaux critères d'optimisation en adéquation avec la problématique introduite. Enfin des méthodes de résolution sont développées pour appréhender les différents problèmes mis en évidence par ces critères.