

Table des matières

Avant Propos	i
Remerciement	ii
Résumé.....	v
Abstract	vi
Liste des figures.....	x
Liste des tableaux.....	xii
Introduction générale.....	13
Chapitre I Les problèmes de chargement/déchargement dans le cas du transport conteneurisé : généralités et contexte	17
1 Introduction.....	17
2 Transport conteneurisé.....	18
2.1 La conteneurisation.....	18
2.2 Modes de transport.....	19
2.2.1 Transport routier.....	19
2.2.2 Transport ferroviaire.....	19
2.2.3 Transport maritime.....	19
2.2.4 Transport fluvial.....	20
2.2.5 Situation en Europe.....	22
2.2.6 Performance environnementale des différents modes de transport.....	23
2.2.7 Transport intermodal	25
3 Opérations dans un terminal à conteneurs	25
4 Problème de chargement et d'arrimage des conteneurs dans les navires.....	26
4.1 Définition du problème.....	26
4.2 Etat de l'art.....	27
4.2.1 Approche en plusieurs phases	27
4.2.2 Approche en une seule phase.....	29
5 Le problème de <i>bin-packing</i>	32
5.1 Le bin-packing en une dimension (BPP-1D).....	32
5.2 Le bin-packing en deux dimensions (BPP-2D).....	34
5.3 Le bin-packing en trois dimensions (BPP-3D)	35
5.3.1 Méthodes de résolution	36
6 Conclusion.....	36
Chapitre II Le <i>shifting</i> dans le problème de voyageur de commerce : de la modélisation à la résolution	38
1 Introduction et travaux antérieurs	38
2 Description du problème.....	41
3 Formulation mathématique du ShTSP.....	46

4	Résolution du ShTSP	50
4.1	Principe de l'algorithme de colonie de fourmis (ACO)	50
4.2	L'adaptation de l'algorithme de colonie de fourmis (ACO)	51
4.3	Algorithme direct	52
4.4	Adaptation de l'algorithme de colonie de fourmis parallèles	53
4.5	Réglage des paramètres	54
4.6	Résultats expérimentaux	54
4.6.1	Génération des instances	55
4.6.2	Résultats numériques et discussion	57
5	Conclusion	61

Chapitre III Optimisation du problème de chargement et d'arrimage des conteneurs avec coût de *shifting* non-uniforme : le cas d'une seule barge 62

1	Introduction	62
2	Transport vert : transport par barge comme alternative écologique aux autres modes du transport intermodal	63
3	Description du problème d'optimisation des plans de chargement et d'arrimage des conteneurs dans la barge	64
3.1	Structure physique de la barge	64
3.2	La politique du « dernier entré, premier sorti » (LIFO)	66
3.3	Les frais non-uniformes des opérations de manipulation des conteneurs	67
3.4	Notion de stabilité de la barge	69
3.4.1	Stabilité transversale : la gîte	70
3.4.2	Stabilité longitudinale : l'assiette	72
3.5	Coordonnées des positions de la barge	73
3.6	Notion de fragilité des conteneurs et son influence sur le poids limite de chaque pile	77
3.7	L'influence des tailles de conteneurs sur le plan de chargement	81
4	Formulation mathématique du 3D-BCSPP	82
5	Résolution et discussion des résultats	91
5.1	L'étude de faisabilité des instances	91
5.2	Génération des instances du 3D-BCSPP	92
5.3	Résultats numériques et discussion	96
5.4	Comparaison numérique	99
5.4.1	La performance des mesures de stabilité	99
5.4.2	Comparaison des frais de <i>shiftings</i> dans le cas uniforme et non-uniforme	102
6	Conclusion	103

Chapitre IV Optimisation multi-objectif du problème de chargement et d'arrimage des conteneurs en 3D dans un système de convoi de barges: modélisation et résolution par le NSGA-II 105

1	Introduction	105
2	Transport fluvial par un système de convoi de barges	106
3	Description du problème	107
3.1	Relation du 3D-CCSPP avec le problème de <i>bin-packing</i> en 3D	109
3.2	La stabilité de la barge selon la distance métacentrique GM	109
3.3	Aspect multi-objectif du 3D-CCSPP	110
4	Formulation mathématique du 3D-CCSPP	112
5	Méthodes de résolution	119
5.1	Aspect multi-objectif de l'approche de résolution	120
5.2	Principe général de la méthode NSGA-II	120
5.3	Adaptation du NSGA-II pour le 3D-CCSPP	121
5.3.1	Codage d'un individu	121
5.3.2	Génération de la population initiale	123
5.3.2.1	Règles de chargement	124
5.3.2.2	Heuristique de chargement des conteneurs dans une seule barge H_UniBarge	125
5.3.2.3	Randomiser l'heuristique constructive H_UniBarge	126
5.3.2.4	Heuristique de chargement des conteneurs dans un convoi de barges H_MultiBarge ...	127

5.3.2.5	Diversification de la population initiale.....	130
5.3.3	Classification des individus	130
5.3.3.1	La notion de non-dominance	130
5.3.3.2	Le mécanisme de sélection Pareto « <i>ranking</i> ».....	130
5.3.3.3	Préservation de la diversité	131
5.3.4	Opérateur de sélection.....	132
5.3.5	Etape de reproduction	133
5.3.5.1	Opérateur de croisement	133
5.3.5.2	Heuristique de la vérification de faisabilité des individus (HVF)	135
5.3.5.3	Opérateur de mutation	139
5.3.6	Etape de remplacement	140
5.3.7	Critère d'arrêt.	140
5.4	Résultats numériques et expérimentaux	140
5.4.1	Environnement expérimental et réglage des paramètres	141
5.4.2	Génération des instances du 3D-CCSPP	141
5.4.3	Résultats numériques et discussion	143
5.4.3.1	Résolution du modèle mathématique par CPLEX	144
5.4.3.2	Résultats numériques obtenus par le NSGA-II.....	146
5.4.3.3	Représentation graphique d'un ensemble de fronts de Pareto	150
5.4.3.4	Analyse des résultats : comparaison et discussions	152
5.4.3.5	Analyse de performance des plans de chargement obtenus par NSGA-II	156
6	Conclusion.....	163
Conclusion générale et travaux futurs		165
Liste de références.....		169

Liste des figures

Figure I-1. Le trafic mondial des ports de conteneurs entre 2000 et 2016 (Worldbank, 2016)	20
Figure I-2. Formes du transport fluvial	21
Figure I-3. (a) Automoteur, (b) barge, (c) caboteur fluvio-maritime (Source : www.meretdesign.fr)	21
Figure I-4. Passage d'une barge sous un pont.....	22
Figure I-5. Part de marché du mode fluvial dans les ports maritimes de la rangée nord-européenne (Oudani, 2016)	22
Figure I-6. Premier système portuaire français HAROPA. Source : http://www.haropaports.com	23
Figure I-7. Illustration comparative des trois modes de transport intérieur	24
Figure I-8. Livres d'émission par tonne-mille du transport routier, ferroviaire et fluvial. Source: (Haulk, 1998)	24
Figure I-9. Efficacité énergétique des trois modes de transport intérieur. *Source : http://www.caria.org	24
Figure I-10. Grues de quai	25
Figure I-11. Vue schématisée d'un terminal à conteneurs.....	27
Figure I-12. Exemple d'un rangement non orthogonal	35
Figure I-13. La contrainte de l'orientation dans le BPP-2D.....	35
Figure I-14. Exemple d'un packing en 3D.....	35
Figure II-1. Pile de cinq produits.....	42
Figure II-2. Point d'accès vertical et horizontal	43
Figure II-3. Illustration d'une tournée de véhicule.....	43
Figure II-4. Piles produits, niveaux, clients.....	44
Figure II-5. Exemple d'une solution réalisable du ShTSP	44
Figure II-6. Illustration de chaque état de la pile pendant toute la tournée.....	45
Figure II-7. Les deux objectifs (minimiser le coût distance et minimiser le coût <i>shifting</i>) ne varient pas dans le même sens	46
Figure II-8. L'algorithme d'ACO-parallèles	53
Figure II-9. Comparaison du temps CPU de l'ACO et de l'ACO-Parallèles, les deux appliqués aux instances du Tableau II-1 avec 100 fourmis utilisant quatre unités de calcul	60
Figure II-10. Comparaison du temps CPU de l'ACO et de l'ACO-Parallèles, les deux appliqués aux instances du Tableau II-1 avec 100 fourmis utilisant huit unités de calcul	60
Figure II-11. Comparaison du temps CPU (ms) de l'ACO-Parallèles adapté aux instances du Tableau II-2 avec 100 fourmis en utilisant quatre puis huit unités de calcul	61
Figure III-1. Vues d'en-haut, transversale et longitudinale.....	65
Figure III-2. Le système BAROTI.....	65
Figure III-3. Structure cellulaire générale d'une barge à conteneurs	66
Figure III-4. Illustration des slots de la barge.....	66
Figure III-5. Plan d'ensemble des conteneurs sur une barge de 208 EVP	66
Figure III-6. La politique LIFO appliquée en haut de la barge.....	67
Figure III-7. Illustration du plan d'arrimage du conteneur sur la barge	67
Figure III-8. Un exemple de transport le long d'une rivière avec P ports à visiter. (Source : adaptée à partir de (Alfandari et al., 2019)).....	68
Figure III-9. Les facteurs de stabilité	70
Figure III-10. Illustration de la stabilité d'une balance. Source : https://www.slideshare.net/pcorsano/active-management-of-vessel-stability	70
Figure III-12. La gîte et la distance métacentrique de la barge.....	71
Figure III-11. Stabilité transversale d'une barge. Source : https://www.slideshare.net/pcorsano/active-management-of-vessel-stability	71
Figure III-13. L'assiette longitudinale de la barge.....	72
Figure III-14. Dimensions des conteneurs les plus courants (adaptée par Delgado (2013) à partir de (Pacino, 2012))	73
Figure III-15. Les positions dans la barge selon la largeur, la longueur et la hauteur	74
Figure III-16. Le centre de gravité et le centre géométrique des positions	74
Figure III-17. La position du repère R dans la barge	75
Figure III-18. Les coordonnées du centre de gravité G de la barge.....	75

Figure III-19. Exemple d'une barge de capacité 45 EVP	75
Figure III-20. Les coordonnées des positions dans la barge en 3D	76
Figure III-21. La notion de fragilité d'un conteneur.....	78
Figure III-22. Une instance d'un ensemble de conteneurs.....	79
Figure III-23. Une solution réalisable de l'instance de la Figure III-22 avec calcul explicite de la fragilité de chaque pile	80
Figure III-24. Une solution non réalisable au sens de fragilité.....	81
Figure III-25. La stabilité transversale et longitudinale de la barge.....	85
Figure III-26. Contraintes de tailles mixtes de conteneurs (a).....	86
Figure III-27. Les pièces de coins des conteneurs 20' et 40'	87
Figure III-28. La contrainte de pièces de coins	88
Figure III-29. Contraintes de tailles mixtes de conteneurs (b).....	88
Figure III-30. Le <i>shifting</i> pour un conteneur 40'	89
Figure III-31. Instance simplifiée du 3D-BCSPP	91
Figure III-32. Problème de faisabilité des instances du 3D-BCSPP.....	92
Figure III-33. L'influence de la nature de tailles de conteneurs sur la stabilité transversale de la barge	100
Figure III-34. L'influence de la nature de tailles de conteneurs sur la stabilité longitudinale de la barge	100
Figure III-35. L'influence de la distribution des intervalles de poids sur la stabilité transversale.....	101
Figure III-36. L'influence de la distribution des intervalles de poids sur la stabilité longitudinale.....	102
Figure IV-1. Deux types de système de barges poussées dans un convoi. Source : (Van Hassel, 2011).....	107
Figure IV-2. Exemple d'un convoi poussé de deux barges homogènes de capacité 27 EVP totalement remplies.....	108
Figure IV-3. La distance métacentrique GM de la barge	110
Figure IV-4. Le plan de chargement obtenu lorsque seule la stabilité est maximisée.....	111
Figure IV-5. Le plan de chargement obtenu lorsque seul le nombre de <i>shiftings</i> est minimisé.....	111
Figure IV-6. Le plan de chargement obtenu lorsque seul le nombre de barges réellement utilisées est minimisé.....	112
Figure IV-7. La position du métacentre M dans la barge.....	116
Figure IV-8. Schéma explicatif des étapes du NSGA-II.....	121
Figure IV-9. Illustration du codage proposé pour le 3D-CCSPP.....	122
Figure IV-10. Illustration d'une solution réalisable du 3D-CCSPP (non optimale).....	123
Figure IV-11. Illustration de la règle de chargement RP1	125
Figure IV-12. Illustration de la règle de chargement RP2	125
Figure IV-13. Illustration des fronts de Pareto dans un problème de minimisation bi-objectifs.....	130
Figure IV-14. Calcul de distance d'encombrement (<i>crowding</i>)	132
Figure IV-15. Fonctionnement de l'opérateur croisement proposé pour le 3D-CCSPP	134
Figure IV-16. Fonctionnement de la stratégie de correction pour l'individu enfant E1	137
Figure IV-17. Fonctionnement de la stratégie de correction pour l'individu enfant E2	138
Figure IV-18. Fonctionnement de l'opérateur de mutation M1	139
Figure IV-19. La procédure schématique du NSGA-II en une itération.....	140
Figure IV-20. Front optimal de Pareto en 3D obtenu par le NSGA-II pour l'instance CCSP35.....	150
Figure IV-21. Front optimal de Pareto en 3D obtenu par le NSGA-II pour l'instance CCSP39.....	150
Figure IV-22. Front optimal de Pareto en 3D obtenu par le NSGA-II pour l'instance CCSP50.....	151
Figure IV-23. Front optimal de Pareto en 3D obtenu par le NSGA-II pour l'instance CCSP64.....	151
Figure IV-24. Front optimal de Pareto en 3D, de l'instance CCSP19, obtenu par le NSGA-II en utilisant l'heuristique HNF_MultiBarge	152
Figure IV-25. Front optimal de Pareto en 3D, de l'instance CCSP19, obtenu par le NSGA-II en utilisant l'heuristique HFF_MultiBarge.....	152
Figure IV-26. Comparaison entre NF-NSGA-II et FF-NSGA-II selon les valeurs de chaque fonction objectif.....	153
Figure IV-27. L'influence de P sur le nombre de <i>shiftings</i>	154
Figure IV-28. L'influence de $ N $ sur le nombre de <i>shiftings</i>	155
Figure IV-29. L'influence de N sur le nombre de barges réellement utilisées.....	156
Figure IV-30. L'influence de $ N $ sur la stabilité.....	156
Figure IV-31. Pire et meilleur cas de <i>shifting</i> total pour une pile.....	157
Figure IV-32. Comparaison entre le nombre de barges réellement utilisées dans le convoi obtenu par NSGA-II et celui dans le meilleur cas	163

Liste des tableaux

Tableau II-1. Description des instances avec petit $ V $	55
Tableau II-2. Description des instances avec grand $ V $	56
Tableau II-3. Résultats numériques des classes c1, c2 et c3 trouvés par l'ACO et comparés par la Méthode Directe	57
Tableau II-4. Résultats numériques de l'adaptation de l'ACO et de l'ACO-parallèles sur les instances du Tableau II-1	58
Tableau II-5. Résultats numériques de l'adaptation des ACO et des ACO-parallèles sur les instances du Tableau II-2	59
Tableau III-1. Paramètres des ports	93
Tableau III-2. Caractéristiques des barges	93
Tableau III-3. Les intervalles de poids des conteneurs	94
Tableau III-4. Caractéristiques des conteneurs pour chaque instance	94
Tableau III-5. Génération des instances du 3D-BCSPP	95
Tableau III-6. Résultats numériques d'un ensemble d'instances du Tableau III-5 , en minimisant les frais de <i>shiftings</i> où $\alpha = 1$ et $\beta = \gamma = 0$ et leur statut correspondant	96
Tableau III-7. Résultats numériques d'un ensemble d'instances du Tableau III-5 , en optimisant la gîte et l'assiette, où « $\beta = 1$ et $\alpha = \gamma = 0$ » et « $\gamma = 1$ et $\alpha = \beta = 0$ » respectivement, et leur statut correspondant	98
Tableau III-8. Comparaison des frais de <i>shiftings</i> dans le cas uniforme et non-uniforme	103
Tableau IV-1. Réglage des paramètres de l'algorithme proposé	141
Tableau IV-2. Caractéristiques relatives aux convois de barges	142
Tableau IV-3. Les intervalles de poids des conteneurs	142
Tableau IV-4. Informations relatives aux conteneurs	142
Tableau IV-5. Génération des instances du 3D-CCSPP	143
Tableau IV-6. Résultats numériques des instances de petite taille du Tableau IV-5 et leur statut correspondant, en optimisant le nombre de <i>shiftings</i> , la stabilité en GM et le nombre de barges du convoi, où « $\alpha=1, \beta=0$ et $\gamma=0$ », « $\alpha=0, \beta=1$ et $\gamma=0$ » et « $\alpha=0, \beta=0$ et $\gamma=1$ » respectivement	145
Tableau IV-7. Caractéristiques des instances résolues par CPLEX	146
Tableau IV-8. Résultats numériques obtenus par NSGA-II et CPLEX pour les instances de petite taille	148
Tableau IV-9. Information des individus du front optimal de Pareto trouvé par le NSGA-II pour le cas des instances de grande taille	149
Tableau IV-10. Capacités des barges utilisées dans l'analyse de la Figure IV-29	155
Tableau IV-11. Analyse de performance des résultats obtenus par NSGA-II en termes de nombres de <i>shiftings</i> par rapport au pire des cas	158
Tableau IV-12. Analyses de performance du facteur de stabilité GM obtenu par NSGA-II par rapport à 1m et 1.5m par barge	160
Tableau IV-13. Analyse de performance des résultats obtenus par le NSGA-II en termes de nombre de barges réellement utilisées par rapport au meilleur cas	162

Introduction générale

Depuis son introduction pendant le 17^{ème} siècle, la logistique actuelle est devenue l'une des indicateurs les plus significatifs dans le développement des pays. De ce fait, de nombreuses définitions dans la littérature existent. À notre avis, la définition la plus célèbre et la plus simple est celle donnée par Swamidass (2000) connue sous le nom des « Sept Règles », à savoir, « avoir le bon produit, en bonne quantité, en bon état, au bon endroit, au bon moment, pour le bon client et au bon prix ». Chaque axe de la chaîne logistique pose divers problèmes qui doivent être résolus afin de garantir une utilisation optimale des ressources nécessaires au cycle de vie d'un produit donné, depuis le fournisseur jusqu'au consommateur final. Parmi ces axes, nous trouvons les problèmes de transport qui ont fait l'objet d'une attention soutenue en raison de leur importance et leur impact sur l'ensemble de toute la chaîne logistique, pour les responsables de la logistique et des chercheurs. Par conséquent, les études faites dans ce domaine sont très nombreuses et diversifiées. Elles touchent les niveaux stratégique, tactique et opérationnel et elles se sont intéressées à l'optimisation des moyens (flotte, personnels, ..., etc.) et aussi à l'environnement (émissions de gaz à effet de serre, ..., etc.). Souvent, on néglige un facteur essentiel qui peut influencer le coût total, à savoir le coût de chargement/déchargement de la marchandise.

Au niveau stratégique, l'infrastructure est l'un des problèmes les plus difficiles qui se posent à tous les pays, et aux pays en voie de développement en particulier. Certains pays sont dotés d'un ensemble de ports, d'aéroports et de liaisons routières, ferroviaires, et fluviales entre ces derniers et les différentes villes et entre les villes elles-mêmes. Pour d'autres, malheureusement, de très grands efforts doivent être exercés. Des pays ont, en plus des infrastructures réalisées par l'être humain, les infrastructures naturelles, telles que les fleuves à des profondeurs assez importantes qui permettent le transport entre certaines villes par des engins appelés barges. Parmi les pays dotés des voies navigables qui permettent le transport par barge, nous citons le Maroc qui a 3500 km de côtes, la France qui a, en plus des océans, un certain nombre de fleuves, notamment, la Seine qui relie le Grand Port Maritime du Havre (GPMH) à la région parisienne en passant par plusieurs ports intérieurs (Rouen, Gennevilliers, Limay-Porcheville, Evry, Nogent sur Seine, ..., etc.). La barge est l'un

des moyens de transport pour le fret en vrac, les produits liquides et les produits conteneurisés.

Pour l'utilisation des barges, un certain nombre de règles doivent être respectées aux points de chargement/déchargement et entre ces points. En effet, pendant le chargement, le problème de l'équilibre de la barge se pose, ce qui impose une modalité particulière de chargement. Autrement dit, les contraintes de stabilité de la barge se posent aux points de chargement et de déchargement. Par ailleurs, entre les ports d'origine et de destination, on peut trouver un certain nombre de ponts, ce qui impose la limitation de la hauteur des barges. Les barges se déplacent en convoi à l'aide d'un pousseur donc une certaine longueur s'impose et bien sur la largeur est imposée par la nature du fleuve.

Dans cette thèse, nous nous intéressons à la modélisation et à la résolution par des méthodes exactes et approchées des problèmes de chargement/déchargement et leur influence sur le coût dans les cas du transport routier et fluvial. Notre objectif principal est, d'une part, l'étude de l'influence des opérations de chargement/déchargement dans les problèmes de routage et d'autre part, l'étude de ce problème dans le cas d'une seule barge et dans le cas multi-barges. Ce dernier est connu sous le nom de problème de chargement et d'arrimage des conteneurs, en anglais, *Container Stowage Planning Problem* (CSPP). Les conteneurs sont empilés selon le principe « dernier entré, premier sorti », de telle sorte que pour décharger un certain conteneur, il est nécessaire de décharger puis recharger tous les conteneurs qui se trouvent au-dessous, ce qui rend un conteneur directement accessible uniquement s'il se trouve en haut de la pile, chaque déplacement temporaire étant appelé *shifting*. Nous pouvons inclure le problème de *shifting* dans le cadre du problème de *bin-packing* à trois dimensions. Il influence fortement l'optimisation de temps global du transport.

Un nombre très important d'études sur le CSPP ont été réalisées au cours des dernières années, principalement axées sur le transport maritime entre terminaux de conteneurs portuaires utilisant les navires porte-conteneurs. Les premières tentatives de résolution du CSPP remontent au travail de Van et Webster (1969). Depuis lors, différentes méthodes de résolution ont été appliquées, notamment la programmation mathématique, la programmation par contraintes, les heuristiques, les méta-heuristiques et la simulation, nous citons les travaux de Aslidis (1989), Botter et Brinati (1992), Ambrosino et Sciomachen (1998), Avriel et al. (1998), Davidor (1998), Dubrovsky et al. (2002), Imai et al. (2002), Sciomachen et Tanfani (2003), Ambrosino et al. (2004), Li et al. (2008), Aye et al. (2010), Delgado et al. (2012), Pacino et Jensen (2012), Ding et Chou (2015), Arajo et al. (2016) et Azevedo et al. (2018). Bien qu'une attention considérable ait été accordée dans la littérature au CSPP dans le cas du transport maritime, à notre connaissance, seuls Li et al. (2017) et Min et al. (2017) ont étudié le problème de chargement et d'arrimage des conteneurs dans le cas du transport fluvial avec une seule barge.

Le présent mémoire est rédigé en français et se compose en plus de cette introduction, de quatre chapitres et d'une conclusion.

Dans le chapitre I, « le problème de chargement/déchargement dans le cas du transport conteneurisé: généralités et contexte », après une introduction, nous rappelons le concept de

conteneurisation qui apparaît aujourd'hui comme un phénomène majeur qui a marqué l'économie générale mondiale, et qui continuera, sans doute, à le faire. Puis, nous mettons l'accent sur les principaux facteurs qui déterminent la compétitivité entre les différents modes de transport intermodal. Ensuite, nous résumons les principales problématiques inhérentes au transport par conteneurs, et nous nous focalisons sur celles liées aux techniques de chargement/déchargement qui sont aujourd'hui au cœur des préoccupations de tous les acteurs du transport maritime, fluvial, ferroviaire et routier. Par ailleurs, nous présentons le contexte général du problème de chargement et d'arrimage des conteneurs en définissant le concept clé des mouvements *shiftings*, en passant en revue les différents travaux qui existent dans la littérature. Avant de conclure, nous résumons les différentes formes du problème de *bin-packing* (*Bin-Packing Problem*, (BPP)) vue sa relation avec le CSPP.

Dans le chapitre II, « le *shifting* dans le problème de voyageur de commerce: de la modélisation à la résolution », nous introduisons la notion de *shifting* dans le problème de voyageur de commerce (*Shifting Traveling Salesman Problem*, (ShTSP)) en tant que nouvelle variante du problème de voyageur de commerce (TSP) associé au problème de chargement et d'arrimage des conteneurs (CSPP). Nous décrivons les différentes caractéristiques du ShTSP, qui relève le défi consistant à déterminer le parcours optimal d'un véhicule en tenant compte des mouvements supplémentaires des produits lors de leur déchargement, causés par la politique « dernier entré, premier sorti ». Nous proposons, ensuite, un modèle mathématique non linéaire mixte, puis nous adaptons les algorithmes de colonie de fourmis séquentielles et de colonies de fourmis parallèles pour résoudre le problème.

Dans le chapitre III, « optimisation du problème de chargement et d'arrimage des conteneurs en trois dimensions avec coût de *shifting* non-uniforme : le cas d'une seule barge », nous nous intéressons à un cas inspiré par des problèmes réels de chargement et d'arrimage des conteneurs dans les terminaux portuaires fluviaux (3D - *Barge Container Storage Planning Problem*, (3D-BCSPP)), où les coûts de chargement/déchargement sont non-uniformes, c.-à-d., ils diffèrent d'un port à un autre. Compte tenu d'une seule barge avec ses caractéristiques structurelles et son itinéraire, décrit par une séquence de ports à visiter où chaque port a des frais de manutention de conteneurs différents, le problème consiste à définir le plan d'arrimage pour un ensemble donné de conteneurs de différentes tailles, poids, fragilités et ports de destination. Tous ces conteneurs doivent être chargés à bord en optimisant le coût total de *shiftings* et la stabilité de la barge. Nous décrivons le problème formellement et identifions ses différentes caractéristiques. Puis nous proposons un modèle mathématique approprié qui permet d'attribuer la position exacte à chaque conteneur tout en respectant les contraintes de chevauchement, des emplacements vides, de tailles des conteneurs et de fragilité. Une série de tests expérimentaux est ensuite effectuée à l'aide du solveur CPLEX, les résultats vérifient la faisabilité du modèle proposé et démontrent son efficacité.

Dans le chapitre IV, « optimisation multi-objectif du problème de chargement et d'arrimage des conteneurs dans un système de convoi de barges en trois dimensions: modélisation et résolution par le NSGA-II », nous nous intéressons au concept de massification dans le

transport fluvial, où le plan de chargement et d'arrimage des conteneurs est appliqué pour un système de convoi constitué d'un bateau poussant plusieurs barges amarrées entre elles. Nous notons cette variante du CSPP par 3D-CCSPP (3D – *barge Convoy Container Stowage Planning Problem*). Nous traitons ce problème sur la base de sa relation avec le problème de *bin-packing* en trois dimensions (3D-BPP). Nous proposons, d'abord, un modèle mathématique dans lequel nous considérons l'aspect multi-objectif en optimisant le nombre de *shiftings*, la stabilité du convoi et le nombre de barges réellement utilisées dans le convoi. C'est un aspect nouveau et pratique compte tenu de l'état de l'art des problèmes de chargement des barges. Une série d'exemples pratiques résolus par le solveur CPLEX est ensuite proposé. En outre, à cause de son NP-complétude, nous avons proposé de développer la méta-heuristique multi-objective NSGA-II (*Non-dominated Sorting Genetic Algorithm-II*), en la combinant avec les heuristiques du problème de *bin-packing* pour résoudre le problème sur des instances de grandes tailles. L'ensemble des résultats numériques obtenus est enfin présenté et évalué en utilisant des mesures de performances adaptées au problème.

Chapitre I

Les problèmes de chargement/déchargement dans le cas du transport conteneurisé : généralités et contexte

1 Introduction

Le transport conteneurisé est un système impliquant le transport de marchandises en utilisant des conteneurs standardisés afin de réduire le coût du transport dans le cadre intermodal. Cependant, plusieurs problématiques inhérentes au transport par conteneurs ont été révélées durant ces dernières années. Parmi ces problèmes, nous citons le problème d'ordonnancement des grues de quai, le problème d'allocation des postes à quai, le problème de la rupture de charge, le problème de la logistique du dernier kilomètre et le problème de la connexion entre les différents modes de transport. Dans ce travail, nous nous intéressons à ceux liés aux techniques de chargement/déchargement, en particulier, le problème d'optimisation des plans de chargement et d'arrimage des conteneurs (*Container Stowage Planning Problem*, (CSPP)).

Dans sa version la plus simple, le CSPP consiste à organiser un ensemble de conteneurs sur un navire porte-conteneurs, les conteneurs n'étant accessibles qu'à partir du haut de la pile. De ce fait, les opérations de *shiftings*, qui consistent au déchargement temporaire puis au rechargement des conteneurs, sont inévitables si un conteneur devant être déchargé sur le port actuel est empilé sous des conteneurs à décharger dans des ports ultérieurs sur la route du navire.

Nous commençons par rappeler le concept de conteneurisation et quelques différentes caractéristiques des principaux modes liés au transport intermodal. Puis, nous donnons une étude comparative de ces modes en se basant spécifiquement sur l'impact environnemental. Quelques problèmes d'optimisation liés aux différentes opérations dans un terminal à conteneurs sont ensuite présentés. Parmi ces problèmes, nous mettons l'accent sur la planification de chargement et d'arrimage de conteneurs (CSPP), en passant en revue les différentes recherches trouvées dans la littérature. Vue la relation du problème de *bin-packing* (*Bin-Packing Problem*, (BPP)) avec le CSPP, nous présentons, à la fin, le contexte général et les différentes formes du BPP.

2 Transport conteneurisé

*“The **box** (container) is what makes the world go **round**”*, Jean-Paul Rodrigue et Brian Slack

2.1 La conteneurisation

L'introduction du concept de conteneurisation a contribué à réduire considérablement le coût du transport de marchandises (Brooks, 2000). Depuis lors, le conteneur est devenu un pilier du commerce dans le monde entier en se développant de façon régulière et continue. En effet, plus de 80% des marchandises diverses sont maintenant transportées en conteneurs. En raison de cette croissance extrêmement importante, les exploitants de terminaux portuaires ont commencé à réfléchir davantage à mener plusieurs études sur les problèmes liés aux mouvements des conteneurs afin d'améliorer l'efficacité et réduire les coûts totaux de transport.

La conteneurisation concerne le transport intermodal de marchandises conteneurisées, en utilisant plusieurs modes de transport tels que les navires, les camions, les trains et les barges, sans aucune manipulation du fret lui-même lors du changement de mode (Crainic et Kim, 2007). Ceci dit, son importance réside dans la possibilité de déplacer le fret avec fluidité, entre plusieurs modes de transport. La conteneurisation joue alors un rôle majeur dans la réduction de la manutention du fret, impliquant ainsi une amélioration de la sécurité, une réduction des dommages et des pertes, et une garantie d'un transport plus rapide en diminuant les temps de rupture de charge et de transbordement (Agerschou, 2004).

Les conteneurs, qui font partie de la catégorie des UTI « Unités de Transport Intermodal », sont de grandes boîtes métalliques de dimensions standards multiples. Leurs dimensions ont été normalisées au niveau international, certes, leur référence statistique reste le 20 pieds, l'EVP¹ ou, plus fréquemment, le TEU² avec les conversions suivantes :

- ▶ Un 20 pieds vaut 1 EVP (1 TEU).
- ▶ Un 40 pieds vaut 2 EVP (1 FEU³ = 2 TEU)

En revanche, ils partagent le même système de fixation, avec lequel ils sont munis à tous les angles de pièces de coin (*corner casting*) permettant de les transborder d'un véhicule à l'autre et de les arrimer efficacement en piles, que ce soit dans des zones de stockage fixes, telles que les dépôts, les entrepôts et les terminaux portuaires, ou dans les zones de stockage mobiles, telles que les baies, les piles des grands porte-conteneurs et des barges.

¹ Equivalent Vingt Pieds

² En anglais, *Twenty feet Equivalent Unit*

³ En anglais, *Forty feet Equivalent Unit*

2.2 Modes de transport

Les modes de transport sont les composants essentiels des systèmes de transport, ils peuvent être regroupés en trois grandes catégories sur la base du milieu exploité: terre, eau et air. Chaque mode a ses propres exigences et fonctionnalités et est adapté aux besoins spécifiques du trafic de marchandises et de passagers. Cela donne lieu à des différences marquées dans la manière dont les modes sont déployés et utilisés dans différentes parties du monde (Rodrigue et al., 2017). Nous résumons par la suite les différentes caractéristiques des principaux modes de base de transport.

2.2.1 Transport routier

Le transport routier est le mode de transport qui s'est le plus développé au cours des 50 dernières années, tant pour le transport de passagers que pour le transport de marchandises (Rodrigue et al., 2017). Il occupe toujours une place prépondérante. En effet, malgré la présence d'alternatives, le transport routier possède des avantages importants par rapport aux autres modes. L'un de ses attributs les plus importants est la flexibilité du choix d'itinéraire, une fois qu'un réseau de routes est fourni. Le transport routier offre l'opportunité unique de fournir un service porte-à-porte pour les passagers et la marchandise, un avantage notable du transport routier par rapport aux autres modes de transport. Les externalités négatives sur l'environnement et le potentiel limité pour réaliser des économies d'échelle restent les principaux inconvénients du transport routier.

2.2.2 Transport ferroviaire

Le transport ferroviaire repose sur une logique économique solide, ce qui en fait une option modale concurrentielle pour la mobilité des passagers et du fret. La capacité des trains de transporter de grandes quantités de marchandises et un nombre important de passagers sur les moyennes et longues distances dans un temps raisonnable est le principal atout de ce mode. De plus, l'utilisation de wagons à double empilement a révolutionné le transport ferroviaire avec une efficacité énergétique accrue et des réductions de coûts pour le transport de conteneurs. Les avantages sont évidents puisque deux conteneurs de 40' (ou 4 de 20') peuvent être empilés sur un wagon, doublant ainsi la capacité d'un train-unité. Dans l'ensemble, le transport ferroviaire est plus efficace que le transport routier, bien que son principal inconvénient soit la flexibilité puisque le trafic doit suivre des itinéraires fixes.

2.2.3 Transport maritime

En raison des propriétés physiques de l'eau conférant une flottabilité et un frottement limité, le transport maritime est le mode le plus efficace pour transporter de grandes quantités de fret sur de longues distances (Rodrigue et al., 2017). Les principales routes maritimes sont composées d'océans, de côtes, de mers, de lacs, de rivières et de canaux.

Le transport maritime des marchandises est considéré comme l'un des modes les plus massifié. En effet, il assure plus de 80% des échanges de marchandises au niveau mondial en utilisant des navires porte-conteneurs de taille croissante qui est passée de quelques centaines de conteneurs à une vingtaine de milliers de conteneurs, permettant ainsi le transport de grandes quantités de marchandises à un coût raisonnable. Le navire *OOCL INDONESIA*, qui peut transporter jusqu'à 21 413 EVP, construit en 2018 et naviguant actuellement sous le pavillon de Hong Kong, est un exemple.

Selon les données trouvées dans (Worldbank, 2016), le trafic de conteneurs dans les ports maritimes devient de plus en plus important. En effet, le trafic mondial des ports à conteneurs est passé de 225 millions d'EVP en 2000 à plus de 701 millions d'EVP en 2016 (**Figure I-1**).

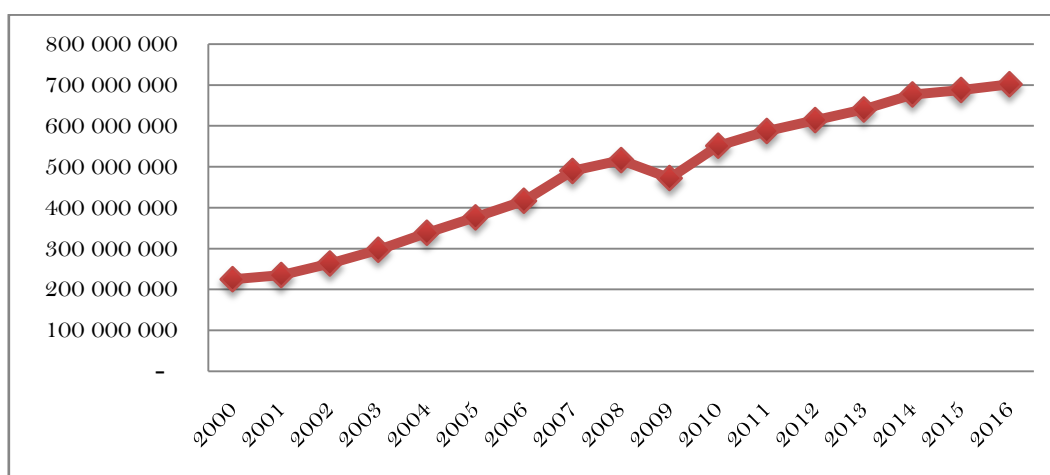


Figure I-1. Le trafic mondial des ports de conteneurs entre 2000 et 2016 (Worldbank, 2016)

Il est à noter que le transport maritime a des coûts de passage portuaire élevés, car les infrastructures sont parmi les plus chères à construire, entretenir et améliorer. Ceci rend la location des quais très coûteuse pour les transporteurs.

2.2.4 Transport fluvial

Par ses caractéristiques de transport massifié, le mode fluvial constitue un moyen de prolongation « à terre » du transport maritime en s'appuyant sur différents navires fluviaux, notamment, les péniches, les automoteurs, les caboteurs fluvio-maritimes ou les convois de barges porte-conteneurs qui offrent une capacité de transport utilitaire inégalable parmi les modes terrestres (**Figure I-2**). En effet, le transbordement s'effectue dans des ports en eau profonde qui agissent comme des *hubs* dans lesquels les conteneurs sont arrimés sur une barge et sont acheminés vers les ports intérieurs (Louis, 2016).

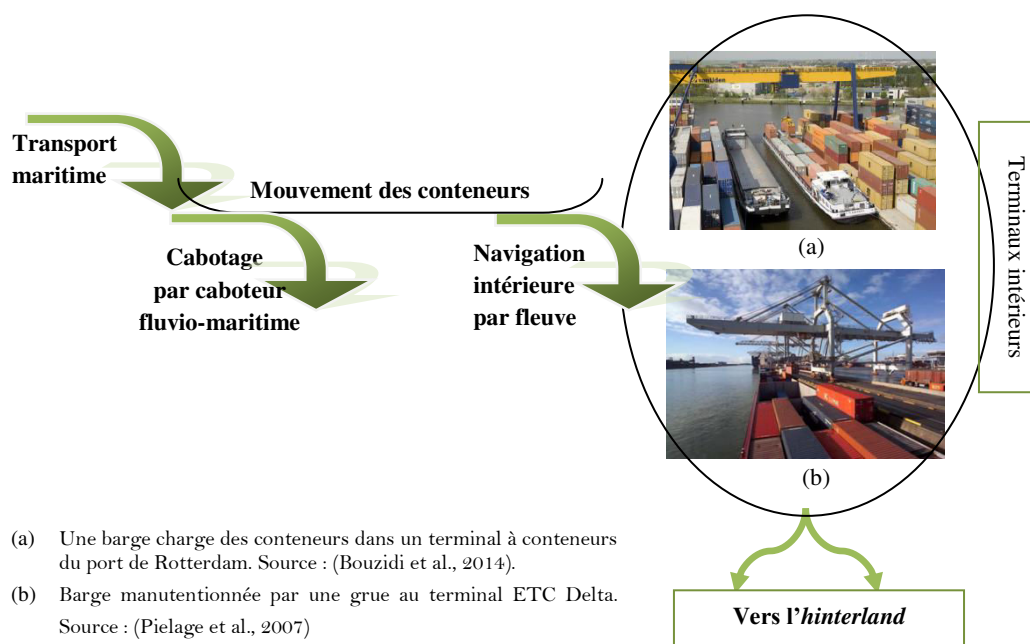


Figure I-2. Formes du transport fluvial

Les conteneurs sont empilés et remorqués vers leurs destinations correspondantes sur les voies d'eau en utilisant des barges. Ces dernières utilisent les réseaux de voies navigables existants qui ne nécessitent pas autant d'infrastructures, d'assistance et de soutien que les réseaux ferroviaires et routiers. Initialement, les barges étaient principalement utilisées dans les systèmes d'eau de petite et moyenne taille comme les rivières, les lacs ou les canaux, mais récemment, leur rôle s'est accru et devenu de plus en plus important dans la navigation fluvio-maritime (Government of Gujarat, 2017), notamment, pour décongestionner les zones d'accès aux sites portuaires de commerce international. Il est à noter que le caboteur fluvio-maritime est un type de navire considéré comme compromis entre le fluvial et le maritime, dont les caractéristiques de tirant d'eau et de tirant d'air lui permettent aussi bien d'affronter les océans que de remonter profondément à l'intérieur des terres par fleuves et canaux à grand gabarit avec un tirant d'eau relativement faible. Ils circulent en cabotage, entre des ports situés tout au long des côtes d'un pays ou d'un continent. La **Figure I-3** montre un exemple de navires fluviaux.

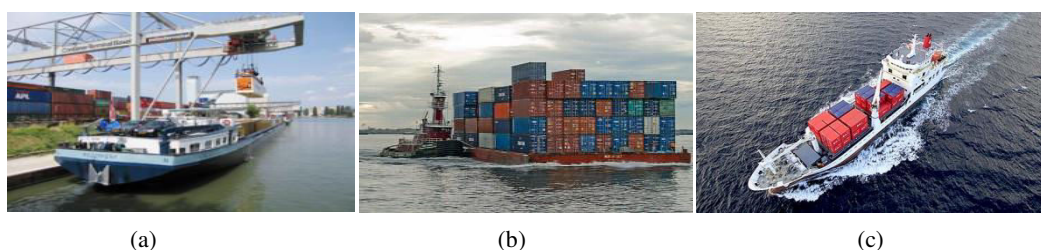


Figure I-3. (a) Automoteur, (b) barge, (c) caboteur fluvio-maritime (Source : www.meretdesign.fr)

La capacité de chargement des conteneurs dans les navires fluviaux peut varier de manière significative, environ une dizaine à une centaine d'EVP. De plus, dans le cas des systèmes de convois de barges (Pielage et al., 2007), cette capacité est fortement déterminée par les

caractéristiques de la voie navigable et même de la hauteur des ponts (lorsqu'ils existent dans le trajet) (**Figure I-4**).



Figure I-4. Passage d'une barge sous un pont

2.2.5 Situation en Europe

Dans le nord-ouest de l'Europe, le transport par barge joue un rôle de plus en plus important dans le traitement de trafic. Le transport de conteneurs par barge a ses origines dans les transports entre Anvers, Rotterdam et le bassin du Rhin. Au cours de la dernière décennie, il s'est également fortement développé le long de l'axe nord-sud entre le Benelux et le nord de la France (Rodrigue et al., 2017). Anvers et Rotterdam ont traité ensemble près de 5 millions d'EVP de trafic fluvial intérieur en 2010, soit environ 95% du transport de conteneurs européen par barges (Frémont et Franc, 2008). On trouve également des développements prometteurs au niveau du transport fluvial via la Seine, entre le Havre et la région parisienne, dans le bassin Rhône-Saône entre Marseille et la région lyonnaise et sur l'Elbe et la Weser en Allemagne du Nord.

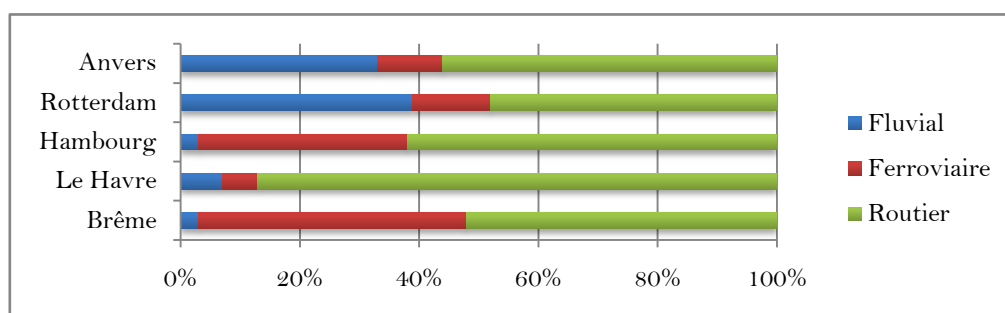


Figure I-5. Part de marché du mode fluvial dans les ports maritimes de la rangée nord-européenne (Oudani, 2016)

Grâce à ses excellents réseaux de voies navigables, l'Europe du Nord-Ouest a enregistré une utilisation importante de bateaux de navigation intérieure (**Figure I-5**). En effet, le volume total de conteneurs transportés par des barges est de 39% dans le port de Rotterdam, de 33% dans le port d'Anvers, de 7% dans le port du Havre et de 3% dans le port de Hambourg.

Comme outil de coopération, les trois ports de la vallée de la Seine en France (Le Havre, ROuen et PARIS) ont choisi de créer, en 2012, un groupement d'intérêt économique sous l'appellation d'« HAROPA » (**Figure I-6**), qui s'étend sur plus de 500 km de voies navigables et dessert un vaste *hinterland*, afin de faire face à la concurrence des ports européens.

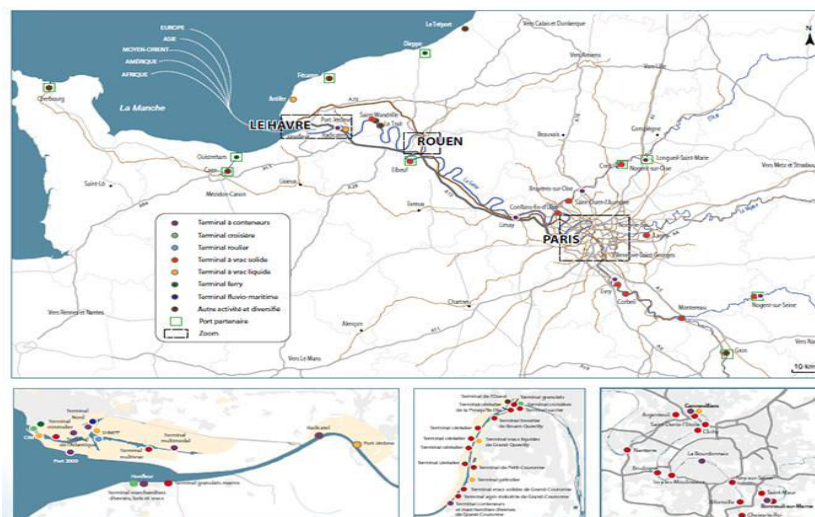


Figure I-6. Premier système portuaire français HAROPA. Source : <http://www.haropaports.com>

Le trafic *hinterland* affiche un record historique avec une croissance du trafic fluvial de conteneurs sur le bassin de la Seine qui a atteint 76 000 EVP, soit une progression de 12,6%⁴. Les chiffres clés du transport fluvial HAROPA sont résumés comme suit :

- ▶ 1^{er} port fluvial français pour le trafic conteneurisé et 2^{ème} d'Europe,
- ▶ 25 % des conteneurs à destination de la région parisienne passant par la Seine,
- ▶ 28 millions de tonnes transportées,
- ▶ 456 000 EVP d'activité fluviale en 2014 (+5 %).

2.2.6 Performance environnementale des différents modes de transport

Ces derniers temps, de nombreuses études ont été menées pour analyser spécifiquement l'impact environnemental des modalités de transport de conteneurs, y compris les barges, les trains et les camions. Bloemhof et al. (2011) ont montré la position concurrentielle de la navigation intérieure vis-à-vis des autres différents modes de transport. En se basant sur les travaux de Mihyeon et Amekudzi (2005), ils ont développé plusieurs mesures de durabilité et ont réalisé une étude de cas sur des barges électriques dont l'énergie provient de batteries rechargeables à terre et logées dans un conteneur standard chargé à bord. Pilot et al. (2016) ont déclaré que, grâce à leur capacité à transporter de grands volumes de marchandises, les bateaux de navigation intérieure réalisent de bonnes performances sur les aspects environnementaux, notamment en ce qui concerne les émissions de CO₂. Une seule barge d'une capacité de 200 EVP sur la voie fluviale peut transporter le même nombre de conteneurs que 50 wagons à double empilement sur le rail ou 200 semi-remorques sur la route (**Figure I-7**).

⁴ Chiffres pour le trafic fluvial de conteneurs maritimes à fin mai 2018 (mesuré au Havre). Source : www.haropaports.com

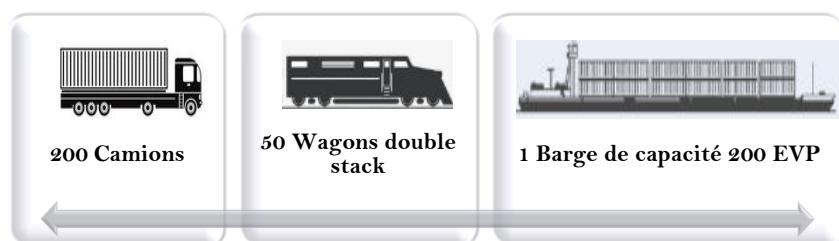


Figure I-7. Illustration comparative des trois modes de transport intérieur

En ce qui concerne la pollution causée par le rail ou la route, les émissions des barges produisent cumulativement moins d'émissions par tonne-mille de marchandises transportées (**Figure I-8**). Compte tenu de cela, des réductions d'émissions importantes peuvent être réalisées en utilisant des barges.

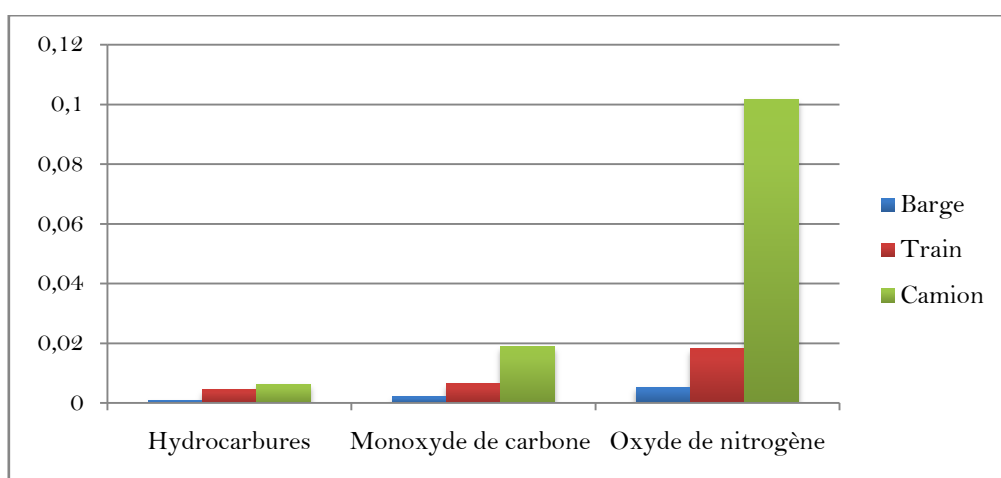


Figure I-8. Livres d'émission par tonne-mille du transport routier, ferroviaire et fluvial. Source: (Haulk, 1998)

En ce qui concerne la distance relative pour le transport d'une tonne de marchandises par chaque mode de transport intérieur pour chaque gallon de carburant brûlé, le transport de conteneurs par barges offre un rendement énergétique presque neuf fois supérieur à celui du transport routier et 3 fois supérieur à celui du transport par voies ferrées (**Figure I-9**).

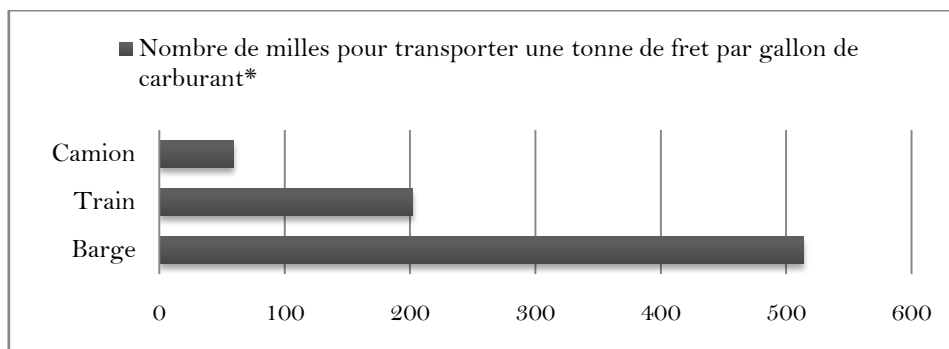


Figure I-9. Efficacité énergétique des trois modes de transport intérieur. *Source : <http://www.caria.org>

Par ailleurs, l'utilisation des barges comme moyen de transport intermodal est fortement favorisée dans plusieurs ports à conteneurs (notamment le port de Rotterdam). Cependant, le processus de transport des conteneurs par barges n'est toujours pas optimal, car ces dernières passent souvent plus de temps que nécessaire au port. Il faut donc impérativement optimiser leurs opérations de manutention. Une bonne planification de chargement/déchargement peut réduire considérablement les coûts.

2.2.7 Transport intermodal

Chaque mode de transport présente des avantages opérationnels et commerciaux essentiels. Les modes peuvent se concurrencer ou se compléter en termes de coût, de vitesse, d'accessibilité, de fréquence, de sécurité, de confort, etc. Récemment, une tendance à l'intégration des modes via l'intermodalité a apparue. Ainsi, le transport intermodal implique l'utilisation d'au moins deux modes différents de transport, de la manière la plus productive, lors d'un trajet allant d'une origine à une destination (Rodrigue et al., 2017). Parmi les principales formes d'intermodalité, citons :

- ▶ Le transport combiné ferroviaire-routier qui permet le transbordement des conteneurs des trains vers les camions et vice-versa,
- ▶ Le transport combiné fluvial-routier qui permet le transbordement des conteneurs des unités fluviales comme les barges vers les camions et vice-versa,
- ▶ Le transport combiné maritime-fluvial qui connecte directement le navire à quai et la barge.

3 Opérations dans un terminal à conteneurs

Les terminaux maritimes et fluviaux doivent répondre à la demande croissante des échanges liés au commerce national et international et rester compétitifs en termes de prix et de services. Dans ce contexte, les coûts de construction de postes à quai et d'exploitation d'équipements de terminal à conteneurs, y compris des grues de quai (**Figure I-10**), sont très élevés pour les exploitants de terminaux à conteneurs. De même, les coûts d'accostage et de manutention sont extrêmement élevés pour les exploitants de navires maritimes et fluviaux. Cela souligne l'importance d'achever tous les processus aussi efficacement que possible, dans le but d'obtenir une réduction significative des coûts et une augmentation des profits à la fois pour les opérateurs de ports et de navires (Al-Dhaheri et Diabat, 2017).



Figure I-10. Grues de quai

Selon Vacca et al. (2007), il existe trois niveaux de planification concernant l'optimisation des opérations dans un terminal à conteneurs : le niveau stratégique, le niveau tactique et le niveau opérationnel. Plus de détails sont donnés dans (Steenken et al., 2004) et (Guan et al., 2013). Le niveau opérationnel implique des décisions en temps réel concernant un ensemble de problèmes, parmi lesquels nous citons :

- ▶ Allocation de poste à quai (*Berth Allocation Problem*, (BAP)) : ce problème cherche un ordonnancement des navires dans chaque poste à quai en tenant compte de la distance de sécurité minimale entre deux navires, étant donné que deux navires ne peuvent pas partager le même poste à quai à la même période de temps (Guan et Cheung, 2004),
- ▶ Planification des grues de quai (*Quay Crane Scheduling Problem*, (QCSP)) : le temps nécessaire au déchargement/chargement des porte-conteneurs dépend de la planification des grues de quai pour chaque section de navire. Une contrainte importante est de ne pas permettre aux grues de quai de se croiser, leurs mouvements étant limités à un rail commun (Kim et Park, 2004),
- ▶ Planification de chargement et d'arrimage de conteneurs (*Container Stowage Planning*, (CSP)) : ce problème vise à développer un plan de chargement des conteneurs dans les navires porte-conteneurs qui permet de minimiser le nombre de mouvements improductifs mais nécessaires au déchargement/chargement du navire en tenant compte de plusieurs facteurs. C'est le problème qui nous intéresse dans ce travail et que nous expliquons dans la section suivante.

4 Problème de chargement et d'arrimage des conteneurs dans les navires

4.1 Définition du problème

La diminution du temps de déchargement des conteneurs dans leurs ports de destination, tout au long du trajet du navire réduit automatiquement le temps de transport global de l'ensemble de la chaîne de transport. Dans ce contexte, la position des conteneurs est le facteur qui influe le plus sur les coûts de transport vu que, parfois, à cause d'un certain ordre de conteneurs empilés dans un navire, le conteneur devant être retiré le premier est empilé sous d'autres conteneurs qui ont une destination plus lointaine. De ce fait, afin de récupérer le conteneur courant, tous les conteneurs au-dessus doivent être déchargés en premier, puis rechargés à nouveau après le déchargement du conteneur voulu. Ces mouvements supplémentaires sont appelés *shiftings* et doivent, donc, être réduits au maximum vu qu'ils génèrent des coûts importants de main-d'œuvre, de matériel et de temps et nuisent à la satisfaction des clients (**Figure I-11**).

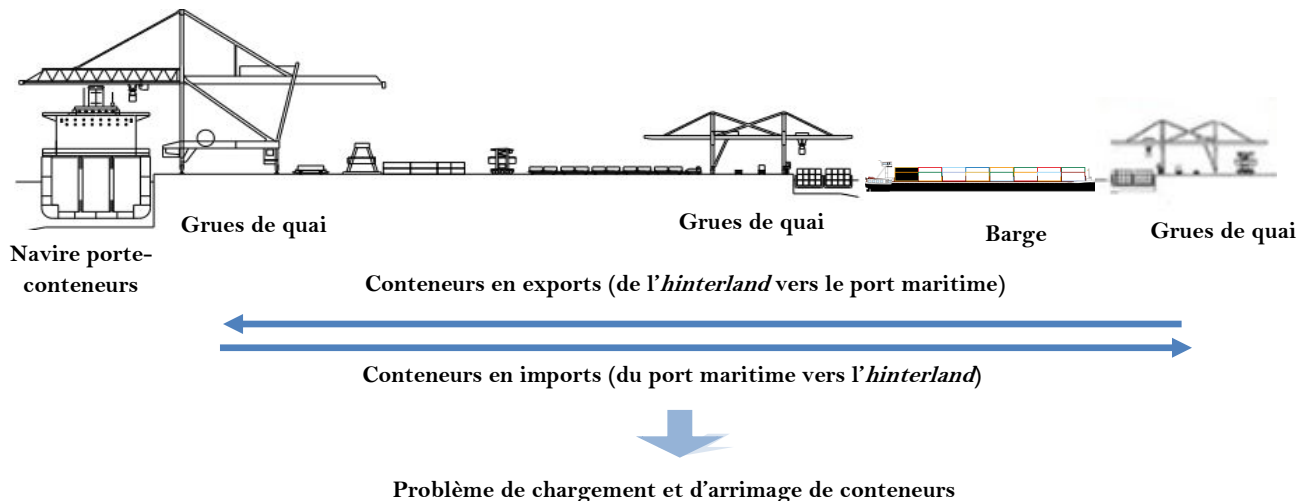


Figure I-11. Vue schématisée d'un terminal à conteneurs

De ce fait, l'efficacité opérationnelle à la fois du port et du navire dépend fortement d'une planification adéquate des mouvements des conteneurs, appelée planification de l'arrimage. Le problème d'optimisation des plans de chargement et d'arrimage des conteneurs (CSPP), également connu sous le nom anglais *Master Bay Plan Problem* (MBPP), se situe dans la classe des problèmes NP-difficiles (Avriel et al., 2000). Il peut être défini comme le problème d'affectation et d'arrimage de différents types de conteneurs, chacun ayant ses propres caractéristiques, dans un ensemble d'emplacements disponibles au sein du porte-conteneurs, en respectant un ensemble de contraintes structurelles, opérationnelles et physiques, liées à la fois, aux conteneurs et au navire, tout en assurant la stabilité du navire et en réduisant le nombre de mouvements supplémentaires « *shiftings* ». Dans la section suivante, nous passons en revue les principaux travaux de synthèse sur le CSPP.

4.2 Etat de l'art

Un nombre très important d'études sur la planification de chargement et d'arrimage des conteneurs (CSPP) ont été publiées au cours des dernières années et ont principalement porté sur le transport maritime entre terminaux de conteneurs portuaires en utilisant des navires porte-conteneurs. Comme indiqué dans (Pacino et al., 2011), les contributions du CSPP peuvent en gros être classées selon deux types d'approches, les approches en une seule phase et les approches en plusieurs phases.

4.2.1 Approche en plusieurs phases

Dans les approches multi-phases, le CSPP est traité sous le nom *Master Bay Plan Problem*, qui est hiérarchiquement divisé en sous-problèmes, souvent en un problème de planification principale (*Master Planning Problem*, (MPP)) et un problème de planification des emplacements (*Slot Planning Problem*, (SPP)). Tout d'abord, un plan général de chargement est créé en distribuant les conteneurs dans les baies des navires. Ensuite, une phase de planification des emplacements exacts se concentre sur chaque baie à la fois, afin de déterminer la position de chaque conteneur.

Wilson et Roach (1999, 2000) divisent le processus d'arrimage des conteneurs en deux sous-processus de planification stratégique et tactique. Ils utilisent notamment des algorithmes de *Branch & Bound* pour résoudre le problème d'affectation de conteneurs à un bloc de baies d'un navire; dans la deuxième étape, ils trouvent un plan détaillé qui attribue à chaque conteneur un emplacement ou une position spécifique d'un bloc de baie en utilisant un algorithme de recherche tabou. Leur approche est capable de trouver des solutions mais l'optimalité n'est pas nécessairement atteinte.

Une décomposition du CSPP en deux sous-problèmes est également proposée dans (Kang et Kim, 2002) ; dans la première phase, les auteurs attribuent des groupes de conteneurs de 20' et de 40' aux cales du navire, tandis que dans la seconde, ils déterminent la séquence de chargement des conteneurs dans chaque cale. Des méthodes heuristiques sont proposées dans le but de minimiser le temps estimé de manipulation de conteneur dans chaque port du trajet.

Sciomachen et Tanfani (2003) présentent, à leur tour, une méthode heuristique pour résoudre le MBPP en se basant sur sa connexion au problème de *bin-packing* à trois dimensions (3D-BPP). La procédure heuristique présentée ici est destinée à exploiter le potentiel de l'algorithme du 3D-BPP proposé par Martello et al. (2000). L'objectif principal est de minimiser le temps total de chargement qui est donné par la somme des temps requis pour le chargement de tous les conteneurs dans leurs emplacements correspondants. Les contraintes de stabilité et d'empilement sont considérées dans l'ordre de chargement des conteneurs et non par les mesures de stabilité du navire.

Dans leurs travaux ultérieurs, Sciomachen et Tanfani (2007) sont allés plus loin et ont inclus dans l'heuristique proposée (Sciomachen et Tanfani, 2003) les équipements du côté quai, à savoir, les grues de quai. Les objectifs étaient de minimiser le temps de chargement total et d'utiliser efficacement l'équipement de quai. La principale contribution de leur travail consiste à évaluer l'influence des plans d'arrimage sur les performances du quai. L'approche a été validée en utilisant des cas de test réels du port de Gênes (Italie), à savoir : un porte-conteneurs d'une capacité de 198 EVP, des conteneurs de poids différents, une ou deux grues et deux ou trois destinations. Cependant, ils n'ont pas tenu compte de l'influence de la stabilité du navire en fonction de la répartition du poids des conteneurs.

De même, Wei-Ying et al. (2005) ont décomposé le problème de planification de l'arrimage pour un porte-conteneurs desservant de nombreux ports en deux sous-problèmes afin de réduire la complexité de calcul. L'étude faite se concentre principalement sur le premier sous-problème qui est considéré comme un problème de *bin-packing* où les baies de navire sont considérées comme des *bins*, le nombre de positions dans chaque baie est considéré comme la capacité du *bin* correspondant et les conteneurs avec leurs différentes caractéristiques sont traités comme des objets à charger. Le but est de minimiser les mouvements *shiftings* et le nombre de baies occupées par les conteneurs dans chaque port.

Dans (Delgado et al., 2012), une approche de programmation par contraintes pour traiter le problème de planification des emplacements (SPP), est proposée. Les auteurs présentent un

modèle pour l'arrimage de conteneurs standards de 20' et de 40', de type normal et réfrigéré, dans une partie donnée du navire, dans le but de réduire au minimum le nombre de *shiftings*, le nombre de piles de conteneurs ayant différentes destinations et le nombre de piles utilisées.

Parallèlement, Pacino et al. (2012) ont présenté un modèle linéaire pour traiter le problème de planification des emplacements (SPP) en incluant des citernes à ballast. Les auteurs présentent une approche permettant de linéariser le calcul du centre de gravité et des tableaux de données hydrostatiques du navire afin de formuler des contraintes de stabilité.

Ambrosino et al. (2004) ont proposé un algorithme en trois phases pour résoudre le MBPP. La première phase consiste à diviser le navire en différentes parties et associe des conteneurs groupés à différents sous-ensembles de baies sans spécifier leur position réelle. Ensuite, ils attribuent la position exacte à chaque conteneur en résolvant un modèle de programmation linéaire en variables binaires. Lors de la dernière phase, certains échanges de recherche locaux sont effectués pour vérifier et éliminer les solutions non-réalisables causées par les contraintes de stabilité.

Il est également intéressant de noter le travail de Parreño et al. (2016), qui ont étudié le problème de planification des emplacements (SPP) en tenant compte des mouvements de conteneurs et des règles de séparation qui se posent lors de la manipulation de conteneurs de marchandises dangereuses.

Une description plus détaillée du MBPP, ainsi que de ses principales contraintes, est donnée dans (Ambrosino et al., 2004), tandis que les méthodes heuristiques proposées pour le résoudre sont comparées dans (Ambrosino et al., 2010).

4.2.2 Approche en une seule phase

Les approches en une seule phase représentent l'espace cargo du navire en tant qu'une structure de données à base cellulaire permettant ainsi de décrire tout le plan d'arrimage en une seule étape.

Avriel et al. (1998) présentent un modèle qui minimise le nombre des mouvements *shiftings* dans la planification de l'arrimage. Cependant, le modèle a une applicabilité limitée en raison de sa complexité. Les auteurs ont donc développé l'heuristique suspensive qui se concentre uniquement sur les contraintes d'empilement de conteneurs et n'inclut aucune considération de stabilité du navire.

Dubrovsky et al. (2002) utilisent un algorithme génétique efficace pour minimiser les mouvements de conteneurs, tout en incluant certains critères de stabilité du navire; les auteurs réduisent considérablement l'espace de recherche en utilisant un schéma de codage compact et efficace. Cependant, le problème a été résolu dans le cas d'une seule baie où tous les conteneurs ont un poids homogène et une même taille standard. En raison de cette simplification, il est facile d'obtenir une répartition équilibrée du poids. En outre, la question de l'équilibre longitudinal a été négligée dans leur étude.

Imai et al. (2002) ont étudié le problème de chargement et d'arrimage en considérant une seule cale de navire et une seule pile dans la cour du port à conteneurs. En utilisant un nombre estimé des mouvements *shiftings*, ils ont trouvé des solutions ayant une stabilité de navire acceptable représentée par un facteur appelé « distance métacentrique » que nous expliquons par la suite.

Ultérieurement, Imai et al. (2006) ont présenté une méthode d'optimisation multicritères du CSPP en considérant la stabilité du navire et le nombre de mouvement des conteneurs. Pour les mesures de stabilité du navire, la distance métacentrique, la gîte⁵ et l'assiette⁶ ont été utilisés, et pour le *shifting*, un nombre estimé a été utilisé.

Une autre recherche a été effectuée par Delgado et al. (2009). Dans leur travail, ils ont appliqué la programmation par contraintes (CP) au problème de la planification d'arrimage. Ils ont indiqué que l'approche CP avait surpassé une approche de programmation en nombres entiers et une approche de génération de colonnes dans une étude préliminaire. Cependant, ils n'ont testé leur approche que dans le cas d'une seule baie en cale du navire.

Tavares de Azevedo et al. (2014) ont étendu le problème de planification du chargement des porte-conteneurs en deux dimensions (*2D Container ship Loading Planning Problem*, (2D-CLPP)), proposée par Avriel et Penn (1993), en un problème de trois dimensions. Ils ont inclus deux critères d'optimisation : le nombre de mouvements des conteneurs et la stabilité du porte-conteneurs. Cependant, ils ont supposé que chaque conteneur avait la même taille et le même poids unitaire. Ils ont introduit une nouvelle approche de codage/décodage appelée « représentation par règles » qui décrit les règles de chargement/déchargement d'un conteneur. Cette nouvelle représentation est combinée avec un algorithme génétique, un recuit simulé et une recherche en faisceau.

Ding et Chou (2015) examinent le problème de planification de l'arrimage lorsqu'un navire porte-conteneurs visite successivement une série de ports sans tenir compte des mesures de stabilité. Ils développent un algorithme heuristique pour générer des plans d'arrimage avec un nombre raisonnable de mouvements *shiftings* et montrent que l'algorithme fonctionne mieux que l'heuristique suspensionnelle proposée dans (Avriel et al., 1998).

De plus, une étude récente portant sur un problème relatif, à savoir, le problème de minimisation des piles d'arrimage avec contraintes de zéro-*shiftings* a été étudiée par Wang et al. (2014). Ce problème vise à trouver un nombre minimal de piles nécessaires pour accueillir tous les conteneurs dans un voyage multiport sans aucun mouvement *shifting*.

Le principal inconvénient d'appliquer l'approche en une seule phase pour résoudre le CSPP est que ce problème est NP-difficile et qu'il n'y a aucune garantie d'obtenir une solution optimale pour les navires de grande taille dans un délai raisonnable de calcul (Wilson et Roach, 1999). Toutefois, dans l'approche en deux phases, étant donné que le plan d'arrimage principal, qui est généré au niveau de la première phase, ne prend pas en compte les

⁵ L'inclinaison transversale d'un navire

⁶ L'inclinaison longitudinale d'un navire

contraintes d'empilement, il est possible que le plan des emplacements qui en résultent, au niveau de la deuxième phase, ne soit pas réalisable (Parreño et al., 2016). Pacino et al. (2011) traitent ce problème avec une procédure de post-optimisation dans laquelle les conteneurs sont supprimés des solutions jusqu'à ce qu'un plan des emplacements réalisable soit atteint.

Tous les travaux rapportés ci-dessus ont été principalement concentrés pour résoudre le problème de chargement et d'arrimage des conteneurs dans des navires maritimes porte-conteneurs. À notre connaissance, seuls Li et al. (2017) et Hu et Cai (2017) ont étudié le problème de chargement et d'arrimage dans un navire fluvial porte-conteneurs. Ils l'ont appliqué sur le fleuve Yangzi Jiang en Chine. Li et al. (2017) ont abordé le problème par une approche en deux phases. Ils ont respectivement conçu une procédure de recherche adaptative aléatoire gloutonne (*Greedy Randomized Adaptive Search Procedure*, (GRASP)) et une heuristique de stratégie évolutionniste (*Heuristic Evolutionary Strategy Algorithm*, (HES)) pour résoudre le problème d'optimisation multi-objectif à deux niveaux. Au premier niveau, ils se sont concentrés sur la minimisation de la différence de poids longitudinale et du nombre de baies utilisées pendant le trajet. Au deuxième niveau, les conteneurs sont divisés en groupes en fonction de leurs caractéristiques, telles que la destination, la taille et le poids, puis, le problème de planification des emplacements (SPP) est traité en se concentrant sur la minimisation du nombre de *shiftings* et du moment d'inclinaison de chaque baie. Hu et Cai (2017) ont proposé un modèle mathématique multi-objectif qui porte uniquement sur les contraintes de stabilité sans tenir compte des contraintes d'empilement liées aux conteneurs. Ils ont utilisé un algorithme heuristique pour générer la solution initiale en minimisant le nombre de *shiftings*, puis cette solution est optimisée par un algorithme génétique en considérant la stabilité longitudinale comme objectif d'optimisation.

Une préoccupation commune aux différentes approches adaptées pour étudier le CSPP consiste à trouver le compromis entre la minimisation des mouvements improductifs *shiftings* et la satisfaction des mesures de stabilité, soit, par le biais de contraintes, soit, par le biais d'une fonction objectif. Certaines approches en deux phases imposent la stabilité en tant que contraintes, mais n'expliquent pas, par exemple, comment le respect des contraintes définies dans la planification principale (au niveau de la première phase) sera assuré après l'application de la planification des emplacements (au niveau de la deuxième phase) et inversement (Delgado et al., 2012). Parfois, pour les grandes instances, il est difficile de traiter la stabilité comme une contrainte (Ambrosino et al., 2010). Sinon, la prise en compte des mesures de stabilité à l'aide d'une fonction objectif conduit au problème multi-objectif de planification de chargement des porte-conteneurs (*3D Container ship Loading Plan Problem*, (CLPP)) (Araújo et al., 2016 ; Tavares de Azevedo et al., 2014). Comme l'ont montré Tavares de Azevedo et al. (2014), donner la priorité à la stabilité du navire dans l'objectif pourrait avoir un impact important sur l'arrangement des conteneurs dans le navire et sur le nombre de mouvements *shiftings* nécessaires lors du déchargement. Cependant, cette limitation pratique est primordiale. Pour une discussion plus détaillée sur les mesures de stabilité et les fonctions objectifs correspondantes, voir (Imai et al., 2006), (Barrass et Derrett, 1999) et (Azevedo et al., 2018).

Dans cette thèse, nous avons traité le problème de chargement et d'arrimage des conteneurs en une seule phase, générant ainsi un plan de chargement détaillé en fonction de la structure cellulaire des navires fluviaux tout en incluant la stabilité dans la fonction objectif. Ce choix est principalement justifié par le fait que, dans le cas de la navigation intérieure, les unités fluviales utilisées sont beaucoup plus petites que les méga-navires dans le transport maritime, mais aussi, elles sont plus touchées par le problème de stabilité à cause de la nature de leur structure physique (Li et al., 2017).

Comme nous venons de mentionner, un certain nombre de chercheurs ont traité le problème de chargement et d'arrimage de conteneurs dans des navires en se basant sur sa relation avec le problème de *bin-packing* (*Bin-Packing Problem*, (BPP)) (Sciomachen et Tanfani, 2003, 2007 ; Wei-Ying et al., 2005). Dans la section suivante, nous donnons un aperçu général sur les différentes caractéristiques et variantes du BPP.

5 Le problème de *bin-packing*

Le problème de *bin-packing* (aussi connu sous le nom du problème de placement ou d'emballage) est l'un des problèmes majeurs qui relève de la recherche opérationnelle et de l'optimisation combinatoire. Il présente des enjeux industriels et économiques très importants, que ce soit sur le plan de production, de transport, de gestion des plannings, de télécommunication, etc. (El Hayek, 2006). La version la plus simple du problème consiste à trouver le rangement le plus économique possible pour un ensemble d'objets dans un ensemble de *bins* en respectant certaines règles de placement. L'objectif principal est de maximiser l'utilisation du matériel et donc de minimiser la zone gaspillée (Baldi et al., 2014 ; Berkey, 1988 ; Berkey et Wang, 1994 ; Boutevin et al., 2003 ; Quiroz-Castellanos et al., 2015).

5.1 Le bin-packing en une dimension (BPP-1D)

5.1.1 Définition du problème

Dans un problème de *bin-packing* unidimensionnel les objets ont une seule dimension, elle peut représenter le poids, la taille, le coût, le temps, etc. Sans perte de généralité, nous supposons que la capacité des *bins* est égale à 1. Alors le problème peut s'énoncer de la manière suivante : étant donné un entier n et un ensemble $U = \{u_1, \dots, u_n\}$ de n objets, auxquels est associée une taille $s(u_i) \in [0,1]$, il s'agit de trouver une partition de U en k sous-ensembles X de U , k est choisi minimal et chaque ensemble X de la partition doit vérifier :

$$\sum_{i \in X} s(u_i) \leq 1$$

En d'autres termes, on place les objets u_i dans des boîtes de taille 1 et on cherche à minimiser le nombre de boîtes nécessaires.

5.1.2 Modélisation du problème

Dans le problème classique, les données sont :

- ▶ m : le nombre d'objets à placer dans les *bins*,
- ▶ n : le nombre de *bins*,
- ▶ a_i : la taille de l'objet i ,
- ▶ b_j : la capacité du *bin* j .

Nous cherchons le rangement optimal pour tous ces objets (qui minimise le nombre de *bins* utilisés). D'abord, pour qu'un rangement soit valide, la somme des tailles des objets affectés à un *bin* j doit être inférieure ou égale à b_j . Pour décrire une solution, nous pouvons utiliser des variables binaires pour indiquer dans quel *bin* j chaque objet i est rangé.

$$x_{ij} = \begin{cases} 1 & \text{Si l'objet } i \text{ est placé dans le bin } j \\ 0 & \text{Sinon} \end{cases}$$

$$y_j = \begin{cases} 1 & \text{Si le bin } j \text{ fait partie de la solution} \\ 0 & \text{Sinon} \end{cases}$$

Dans ce problème, il faut minimiser le nombre de *bins* utilisés :

$$\text{Min} \sum_{j=1}^n y_j$$

Sous les contraintes suivantes :

$$\sum_{i=1}^m a_i x_{ij} \leq b_j y_j, \quad j = 1, \dots, n$$

$$\sum_{j=1}^n x_{ij} = 1, \quad i = 1, \dots, m$$

La première inégalité, appelée contraintes de sac-à-dos, signifie qu'on ne peut pas dépasser la taille d'un *bin* pour un rangement donné. Il est à noter que la partie droite de l'inégalité oblige y_j à prendre la valeur 1 dès qu'un objet est rangé dans le *bin* j . La deuxième égalité contrôle le fait que chaque objet est rangé dans un et un seul *bin*. Cette modélisation a été proposée par Kantorovich (1960).

5.1.3 Méthodes de résolution

▶ Heuristique Next Fit (NF)

Dans cette méthode on ne considère qu'un *bin* ouvert à la fois. Les objets sont traités selon un ordre donné. Les objets sont rangés successivement dans le *bin* ouvert tant qu'il y a de la place pour l'objet a_i en cours, sinon le *bin* en cours est fermé et un nouveau *bin* est ouvert. La complexité temporelle peut être linéaire en fonction du nombre d'objets à placer. Par contre, le fait de ne considérer qu'un seul *bin* à la fois cause beaucoup de perte d'espaces exploitables.

► Heuristique First Fit (FF)

Initialement un seul *bin* est considéré, et les objets sont traités selon un ordre donné. Quand il n'y a plus de place dans le premier *bin* pour ranger l'objet en cours, un deuxième *bin* est alors ouvert mais sans fermer le premier. Dans une étape intermédiaire où on dispose de k *bins* ouverts numérotés de 1 à k selon l'ordre de leur première utilisation, un objet a_i en cours est rangé dans le *bin* du plus faible numéro qui peut le contenir. Dans le cas où aucun *bin* ne peut contenir a_i un nouveau *bin* ($k + 1$) est alors utilisé sans fermer les autres. L'ordre selon lequel on traite les objets est crucial pour la qualité de la solution.

► Heuristique Best Fit (BF)

Comme dans l'heuristique FF, les algorithmes BF laissent les *bins* toujours ouverts. Cependant, le choix du *bin* dans lequel l'objet a_i en cours va être placé dépend des valeurs des gaps c'est-à-dire des hauteurs non utilisées présentes dans les *bins*. Ainsi, a_i sera placé dans le *bin* qui présente le moindre gap parmi les *bins* qui peuvent le contenir.

5.2 Le bin-packing en deux dimensions (BPP-2D)

Le problème de *bin-packing* en deux dimensions est une généralisation du problème classique en une dimension. La majorité des travaux existants dans ce domaine se concentre sur les problèmes de *bin-packing* rectangulaires qui cherchent à déterminer le nombre minimum de grands rectangles identiques, nécessaires pour ranger un ensemble de petits rectangles. Ce problème est un NP-difficile car il généralise le problème classique en une dimension (BPP-1D) connu comme un problème NP-difficile (Hong et al., 2014). Plus formellement, le problème de *bin-packing* en deux dimensions (BPP-2D) est défini de la façon suivante : étant donné un ensemble de n objets rectangulaires $A = \{a_1, \dots, a_n\}$ de dimensions connues et un nombre illimité de rectangles identiques (*bins*) de dimensions plus larges que celles des objets. Nous notons (w_i, h_i) les dimensions d'un objet a_i appartenant à l'ensemble des rectangles à ranger A , et (W, H) les dimensions du *bin*. Le problème consiste à déterminer le nombre minimum de *bins* à utiliser pour ranger l'ensemble de tous les objets sans déborder les *bins* et sans aucun chevauchement (Gomez et Terashima-Marín, 2018 ; Abeysooriya et al., 2018 ; Scheithauer, 2018).

La nature très riche de cette problématique a donné naissance à plusieurs classifications dans la littérature. Nous distinguons notamment la contrainte du rangement orthogonal et la contrainte de l'orientation :

- Les objets sont rangés de telle façon que leurs arêtes soient parallèles à celles des *bins* qui les contiennent, on parle d'un rangement orthogonal. Dans la **Figure I-12**, les deux objets a_3 et a_6 violent cette contrainte.

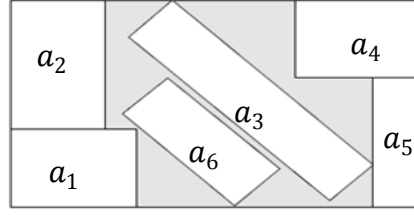
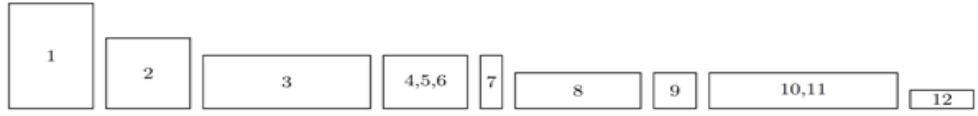
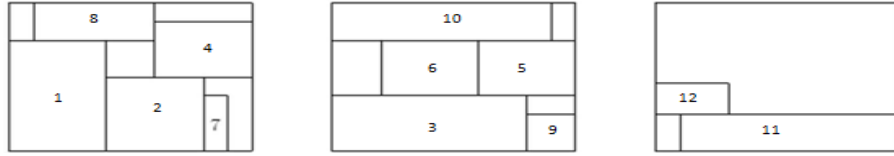


Figure I-12. Exemple d'un rangement non orthogonal

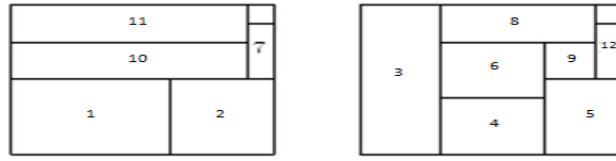
- Les objets peuvent être à orientation fixe (le cas orienté) ou bien ils peuvent être tournés de 90 degrés (le cas non-orienté). Un exemple de solution d'une instance BPP-2D dans le cas orienté et dans le cas non-orienté est donné dans la **Figure I-13**.



(a) Instance de BPP-2D



(b) Bin-Packing dans le cas orienté



(a) Bin-Packing dans le cas non-orienté

Figure I-13. La contrainte de l'orientation dans le BPP-2D

5.3 Le bin-packing en trois dimensions (BPP-3D)

Le problème de *bin-packing* en trois dimensions consiste à placer un ensemble S de n objets de dimensions $(w_i, h_i, d_i), i = 1, \dots, n$, dans un nombre minimum de conteneurs identiques $R_j, j = 1, \dots, m$, de dimensions (W, H, D) où W (resp. w_i) est la largeur, H (resp. h_i) la hauteur et D (resp. d_i) la profondeur des conteneurs (resp. des objets) tout en respectant certaines contraintes (**Figure I-14**) (Hifi et al., 2014 ; Martello et al., 2000 ; Mahvash et al., 2018).

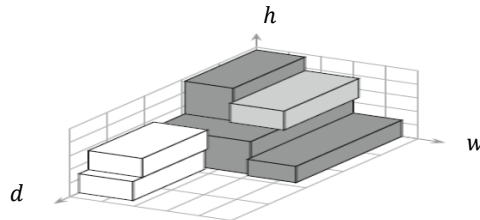


Figure I-14. Exemple d'un packing en 3D

Plusieurs variantes du BPP-3D existent, nous citons particulièrement :

► **Problème de chargement de conteneur**

Dans ce problème (*Container Loading Problem*, (CLP)), nous disposons d'un conteneur de largeur et de longueur finies mais de hauteur infinie. L'objectif est de placer les boîtes à l'intérieur du conteneur de telle sorte que la hauteur de l'espace occupé soit minimale.

► **Problème de chargement de sac-à-dos en 3D**

Dans ce problème, (*Knapsack Loading Problem*, (KLP)), nous disposons d'un ensemble de boîtes (de trois dimensions) ayant chacune une valeur différente. L'objectif est de choisir et placer un sous-ensemble de ces boîtes dans un conteneur unique de dimension finie afin de maximiser la valeur totale des boîtes choisies. Si la valeur d'une boîte correspond à son volume, on minimise alors l'espace résiduel.

5.3.1 Méthodes de résolution

Comme les problèmes de *bin-packing* en 3D constituent une extension naturelle des problèmes en deux dimensions, plusieurs techniques de résolution font appel à des adaptations d'algorithmes connus en deux dimensions ou réduisent le problème en sous-problèmes en deux dimensions. Nous rapportons ici les travaux les plus importants.

À notre connaissance, la première méthode exacte pour le BPP-3D a été proposée dans (Martello et al., 2000), elle consiste en deux phases. Lors de la première phase, un arbre d'énumération est généré afin d'affecter les boîtes aux conteneurs. Dans la seconde phase, une méthode exacte est appelée pour résoudre les problèmes à un seul conteneur obtenus durant la première phase. Aussi, George et Robinson (1980) ont proposé la première heuristique pour le problème en 3D. Cette heuristique insère les boîtes dans un ordre prédéterminé en partant du fond du conteneur (par exemple, de la boîte la plus volumineuse à la moins volumineuse). Deux autres nouvelles heuristiques, appelées « *Deepest-Bottom-Left-Fill* » (DBLF) et « *Maximum Touching Area* » (MTA) ont été proposées. Les boîtes sont préalablement triées et insérées une à la fois. L'heuristique DBLF choisit la position réalisable « la plus en bas, la plus profonde et la plus à gauche ». Dans le cas de MTA, la position choisie pour la boîte courante est celle qui maximise la surface qu'elle touche. Plusieurs autres méthodes existent dans la littérature (Mahvash et al., 2018 ; Paquay et al., 2018).

6 Conclusion

Pour faire face à l'énorme croissance du volume des échanges de conteneurs dans le monde, tous les processus opérationnels de manutention des conteneurs doivent être correctement optimisés. Ce chapitre introductif est consacré à la présentation de certaines notions et concepts liés aux problèmes du transport conteneurisé, notamment, la multimodalité, les défis environnementaux des différents modes de transport et le problème

d'optimisation des plans de chargement et d'arrimage des conteneurs (CSPP). Nous avons présenté un état de l'art général des différentes recherches pour améliorer l'efficacité et réduire les coûts liés au CSPP. Cependant, ces études sont concentrées généralement sur le transport maritime entre les terminaux à conteneurs des ports maritimes, elles abordent rarement le problème dans le cas du transport fluvial. Dans les chapitres suivants, nous traitons le CSPP dans deux contextes différents pour montrer l'influence des plans de chargement sur l'efficacité du transport, le premier est lié au cas du transport routier et le deuxième est donné dans le cas du transport fluvial.

Chapitre II

Le *shifting* dans le problème de voyageur de commerce : de la modélisation à la résolution

1 Introduction et travaux antérieurs

En raison de leur grande applicabilité pratique, les problèmes de routage de véhicules (*Vehicle Routing Problem*, (VRP)) ont été étudiés de manière intensive dans la littérature. Connus comme l'un des problèmes d'optimisation combinatoire classiques, ils sont un élément clé de la logistique de distribution. Cependant, de nos jours, les entreprises de transport sont en forte concurrence, non seulement par rapport aux coûts liés à la logistique, mais aussi par rapport à la différenciation des services qui peuvent être influencés par la contrainte de mouvement improductifs (*shiftings*) qui reste toujours un aspect pratique rarement abordé dans la littérature.

L'élimination du réarrangement et du déchargement temporaire des marchandises est très important en cas de transport d'éléments volumineux, lourds et fragiles ou de matières dangereuses. En effet, on peut faire face à des cas d'application dans le quotidien où les problèmes de chargement et d'arrimage des conteneurs (*Container Stowage Planning Problem*, (CSPP)) et de voyageur de commerce (*Traveling Salesman Problem*, (TSP)) sont combinés. Ces problèmes sont rencontrés dans plusieurs entreprises de transport qui livrent de gros articles tels que des meubles ou des appareils électroménagers. Ainsi, le but de ce chapitre est d'étudier une nouvelle variante des problèmes de routage que nous appelons *Shifting Traveling Salesman Problem*, (ShTSP) (El Yaagoubi et al., 2018). Le ShTSP se pose naturellement dans l'acheminement d'un véhicule qui porte un ensemble de produits placés d'une façon empilée et qui dispose d'un point d'accès unique pour le chargement et le déchargement. Ce véhicule peut également être schématisé par un porte-conteneurs ou une barge (El Yaagoubi et al., 2016), où chaque pile du navire/barge est concernée par le ShTSP.

La suite de ce chapitre est organisée comme suit. La deuxième section décrit le ShTSP formellement et identifie ses différentes caractéristiques. Dans la troisième section, nous proposons un modèle mathématique. Ensuite, dans la quatrième section, nous adaptons l'algorithme de colonie de fourmis dans sa version séquentielle et parallèle pour résoudre le ShTSP, l'ensemble des résultats numériques sont rapportés et commentés.

Le CSPP est présent dans la plupart des problèmes de transport de marchandises et en particulier dans les problèmes du transport maritime et fluvial. Avriel et al. (2000) ont prouvé que le problème de planification de chargement et d'arrimage des conteneurs est NP-complet dans sa version décision en montrant qu'il est lié au problème de coloration de graphe (*Circle Graphs Coloring Problem*, (CGCP)), connu pour être NP-complet. Ceci implique qu'il est très peu probable de trouver une solution optimale en un temps de traitement raisonnable. Depuis les années quatre-vingts, de nombreux chercheurs ont étudié le CSPP qui peut être modélisé comme un problème d'optimisation combinatoire (Shields, 1984 ; Aslidis, 1989 ; Wilson et Roach, 1999 ; Wilson et al., 2001 ; Vis et De Koster, 2003 ; Steenken et al., 2004 ; Günther et Kim, 2006 ; Stahlbock et Voß, 2008). Un plan d'arrimage implique le positionnement d'un ensemble de conteneurs dans un navire et dont l'emplacement exact de chaque conteneur est décrit par une combinaison de numéro de ligne, numéro de baie et numéro de rang. Parmi les principaux objectifs d'un bon plan d'arrimage, citons la réduction du temps de manipulation des conteneurs, la garantie de la stabilité du navire et l'optimisation de l'utilisation des grues (Wilson et Roach, 2000).

Parallèlement à ces études, le CSPP est associé à plusieurs autres problèmes de la littérature. Nous nous référons notamment au problème d'attribution des postes à quai (*Berth Allocation Problem*, (BAP)) (Cordeau et al., 2005). Nous mentionnons également le problème du déplacement dynamique des conteneurs (*the Dynamic Container Relocation Problem*, (DCRP)) (Hakan Akyüz et Lee, 2014) et le problème de routage et de planification des navires (*Ship Routing and Scheduling Problem*, (SRSP)) (Christiansen et al., 2004). Pour plus de détails, nous citons les travaux de Steenken et al. (2004), Stahlbock et Voß (2008) et Douma et al. (2009).

Dans un deuxième contexte, le problème de voyageur de commerce (*Traveling Salesman Problem*, (TSP)) est situé au cœur des problèmes de routage et figure parmi les problèmes les plus importants, les plus étudiés et les plus populaires de l'optimisation combinatoire. De plus, en raison de sa simplicité, le TSP peut modéliser diverses applications. Il reste un problème stimulant ayant motivé de nombreuses publications dont le but est de l'examiner de manière plus efficace. Le TSP classique peut être simplement défini comme la recherche du trajet le plus court entre un ensemble de villes de telle façon à couvrir toutes les villes une seule fois chacune. Le problème étant NP-complet, en effet, en augmentant le nombre de villes n , le temps de calcul de la solution optimale est exponentiel en n , sous la conjecture $P \neq NP$ (Garey et Johnson, 1979).

Depuis son introduction, de nombreuses versions approfondies du TSP ont été étudiées. De plus en plus de contraintes sont associées au TSP classique assurant ainsi un meilleur alignement des problèmes résultants avec les applications réelles. Ces versions approfondies incluent le TSP avec collecte et livraison (*Pickup and Delivery TSP*, (PDTSP)), avec un chargement LIFO (*Last In, First Out*) où la visite d'un nœud de livraison est possible seulement si la charge à livrer se situe au sommet de la pile (Carrabs et al., 2007 ; Cordeau et al., 2010), avec un chargement FIFO (*First In, First Out*) (Erdogan et al., 2009) et celui avec plusieurs piles (Petersen et Madsen, 2009), ainsi que d'autres problèmes de tournées de véhicules (*Vehicle Routing Problems*, (VRP)) avec contraintes de chargement (Iori et Martello, 2010).

Dans la version classique du TSP, il y a un coût fixe associé à chaque arc, pour voyager le long de cet arc ou pour le desservir. À notre connaissance, peu d'efforts ont été consacrés à l'étude du TSP avec un coût de service variable. La plupart des travaux connexes peuvent être trouvés dans (Tagmouti et al., 2011), où le coût du service sur un arc requis est une fonction de la date de début du service. De plus, Tagmouti et al. (2007) ont présenté une autre étude dans laquelle ils ont étudié un problème de routage d'arc avec des contraintes de capacité et des coûts de service dépendant du temps, où un sous-ensemble d'arcs doit être desservi à un coût qui dépend de la date de début du service, étant donné que le coût est une fonction linéaire du temps. En outre, Vander Wiel et Sahinidis (1996) ont présenté un algorithme pour résoudre le problème de voyageur de commerce dépendant du temps (*the Time-Dependent Traveling Salesman Problem*, (TDTSP)), dans lequel le coût du trajet entre deux villes dépend de la distance entre les villes et de l'ordre dans le trajet. Ichoua et al. (2003) ont proposé un modèle dépendant du temps pour un problème de tournée de véhicules avec fenêtres de temps (*Vehicle Routing Problem with Time Windows*, (VRPTW)), basé sur des vitesses de déplacement dépendant du temps, ce qui satisfait à l'hypothèse du premier entré, premier sorti. Il existe un très grand nombre de travaux dans la littérature sur le TSP, nous renvoyons le lecteur intéressé aux travaux de Bektas (2006), Bigras et al. (2008) et Iori et Martello (2010).

À notre connaissance, le ShTSP n'a pas encore été étudié dans la littérature, mais plusieurs autres variantes comprenant à la fois des collectes et des livraisons ont été étudiées, comme le problème de voyageur de commerce avec collecte et livraison et coût de manipulation (*the Traveling Salesman with Pickups, Deliveries, and Handling Costs*, (PDTSPH)) (Battarra et al., 2010 ; Erdogan et al., 2012). Ce problème est une nouvelle variante des problèmes de *the one-to-many-to-one Single Vehicle Pickup and Delivery Problems*, (SVPDP), intégrant les coûts de manutention encourus lors de la réorganisation du chargement chez le client, et où chaque client demande un service de collecte, un service de livraison ou les deux, c'est-à-dire que deux types d'articles sont pris en compte, ceux transportés du dépôt vers les clients et ceux transportés des clients vers le dépôt. Cela implique que le véhicule quitte le dépôt transportant toutes les livraisons, rend visite à chaque client une et une seule fois et retourne au dépôt transportant tous les articles ramassés, tout en respectant les contraintes de capacité. D'autres variantes existent :

- ▶ Le problème de collecte et de livraison avec fenêtres de temps et politique de chargement du dernier entré, premier sorti (*the Pickup and Delivery Problem with Time Windows and Last In, First Out Loading*) (Cherkesly et al., 2015). Dans ce travail, la règle LIFO signifie que lorsqu'un point de collecte est visité, l'article collecté est placé en haut de la pile et ne peut être livré que s'il se trouve dans cette position ; cette règle garantit qu'aucune manipulation supplémentaire (ou *shifting*) n'est requise pour décharger un article d'un véhicule,
- ▶ Le problème de collecte et de livraison avec fenêtres de temps et piles multiples (*the Pickup and Delivery Problem with Time Windows and Multiple Stacks*) (Cherkesly et al., 2016) interdit également les opérations supplémentaires de manipulation,

- ▶ Le problème de voyageur de commerce avec collecte et livraison et coût de manipulation (*the Pickup and Delivery Traveling Salesman Problem with Handling costs*) (Veenstra et al., 2017a), où un seul véhicule doit transporter des charges de nombreuses origines vers de nombreuses destinations ; le chargement et le déchargement du véhicule s'effectuent dans le cadre de la politique LIFO, mais cette fois-ci, une pénalité est associée à chaque opération de manipulation supplémentaire,
- ▶ Le problème de collecte et de livraison avec fenêtres de temps et opérations de manutention (*the Pickup and Delivery Problem with Time Windows and Handling operations*) (Veenstra et al., 2017b), où deux politiques de manutention ont été définies. La première permet uniquement les opérations de manipulation obligatoires et la seconde permet les manipulations obligatoires et préventives, c'est-à-dire que tous les articles peuvent être manipulés pour éviter des manipulations dans des nœuds suivants.

Le présent travail étudie la notion de *shifting* dans le TSP, (*Shifting Traveling Salesman Problem*, (ShTSP)), qui est une nouvelle variante des problèmes de transport combinant les fameux problèmes de TSP et de CSP. Le ShTSP se présente naturellement dans le transport de produits volumineux, lourds, dangereux ou fragiles dans un seul véhicule avec une seule pile où tous les produits sont rangés dans un ordre prédéfini en fonction de leur poids, de leur fragilité et de leur stabilité. La pile a un seul point d'accès pour le déchargement du fret, ce qui signifie que le déchargement de chaque produit est effectué conformément à la politique LIFO, de telle sorte qu'un certain nombre de produits doivent être déchargés pour atteindre les produits en dessous, puis rechargés à nouveau. Notre objectif est de rechercher un circuit optimal qui prenne en compte le coût de *shifting* représenté par l'enlèvement temporaire des produits dans le véhicule, causé par les opérations de déchargement et de rechargement ayant lieu chez chaque client du circuit. Nous proposons un modèle mathématique en tant que programme non linéaire mixte, puis nous résolvons le problème en proposant trois méthodes : la première est une méthode naïve exacte, la deuxième consiste à adapter la méta-heuristique des colonies de fourmis séquentielles et la troisième introduit une nouvelle adaptation des fourmis parallèles, les deux méta-heuristiques étant testées pour un certain nombre d'instances de caractéristiques différentes à partir des données de référence de TSPLIB⁷. Les résultats de calcul montrent l'efficacité de la version améliorée de l'algorithme, qui repose sur le concept parallèle, pour les instances de petite et de grande taille.

2 Description du problème

On considère un véhicule pour livrer des produits rangés dans une pile avec un ordre prédéfini, de telle sorte que pour décharger un certain produit, il est nécessaire de décharger tous les produits qui se trouvent au-dessus puis les recharger à nouveau, ce qui rend un produit directement accessible uniquement s'il se trouve en haut de la pile (**Figure II-1**).

⁷ TSPLIB est une bibliothèque d'exemples d'instances pour le TSP et d'autres problèmes connexes de différents types provenant de différentes sources.

Le problème consiste à trouver une tournée qui minimise à la fois le coût de déplacement du véhicule et le coût de *shifting*.

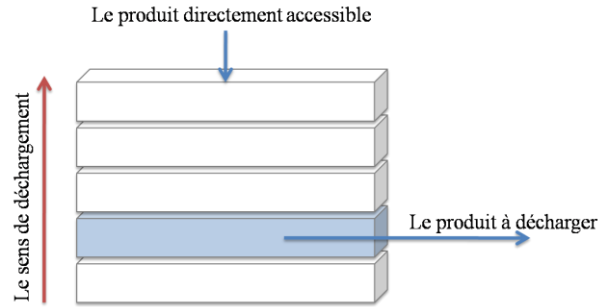


Figure II-1. Pile de cinq produits

Le ShTSP peut être formellement défini sur un graphe complet $G = (V, A)$ avec :

- ▶ N : le nombre de clients et de produits
- ▶ c_0 : le dépôt,
- ▶ $V^* = \{c_1, \dots, c_N\}$: l'ensemble de clients à visiter,
- ▶ $V = V^* \cup \{c_0\}$: l'ensemble des sommets,
- ▶ $A = \{(c_i, c_j), 0 \leq i, j \leq N, i \neq j\}$: l'ensemble des arêtes,
- ▶ $P = \{p_1, \dots, p_N\}$: l'ensemble des produits empilés,
- ▶ d_{ij} : une valeur non négative attribuée à chaque arc $(c_i, c_j), \forall i, j \in \{0, \dots, N\}$ pour représenter le coût de déplacement correspondant. Nous notons que $d_{ij} = d_{ji}, \forall i, j \in \{0, \dots, N\}$.

Les principales hypothèses du ShTSP sont :

- ▶ Chaque produit p_j est destiné à un et un seul client $c_i, \forall i, j \in \{1, \dots, N\}$,
- ▶ Chaque client c_i demande un et un seul produit $p_j, \forall i, j \in \{1, \dots, N\}$,
- ▶ Tous les produits se trouvent au dépôt c_0 ,
- ▶ Les produits $p_j, \forall j \in \{1, \dots, N\}$ sont rangés selon un ordre prédéfini en fonction de leur fragilité, de leur poids et de leur stabilité,

Comme illustré dans la **Figure II-2**, les produits peuvent être empilés l'un à côté de l'autre (horizontalement) ou l'un au dessus de l'autre (verticalement). Le ShTSP est particulièrement intéressant lorsque le coût de *shifting* est comparable à celui de la distance supplémentaire parcourue, causée par la politique LIFO de la pile. Par exemple, pour délivrer un ensemble de produits à l'intérieur d'une ville, il arrive parfois que le véhicule a le choix entre deux clients à visiter, le premier est très proche en termes de temps de déplacement par rapport au deuxième (il peut arriver, par exemple, dans 3 minutes au premier client et dans 5 minutes au deuxième client et entre les deux clients il y'a 4 minutes), alors que le produit du deuxième client est situé en haut de la pile (pour le décharger il faut 0 minutes de *shiftings*) et celui du premier client est en bas de la pile (pour le décharger il faut 4 minutes de *shiftings*), c'est-à-dire, si le véhicule choisit le premier client le coût de *shifting* sera 4 minutes qui est comparable à celui du déplacement s'il choisit le deuxième client (5

minutes). Si le véhicule choisit le premier client, il aura un coût total de 11 minutes et dans le cas inverse il aura 9 minutes. Plus de détails sont donnés dans la suite.

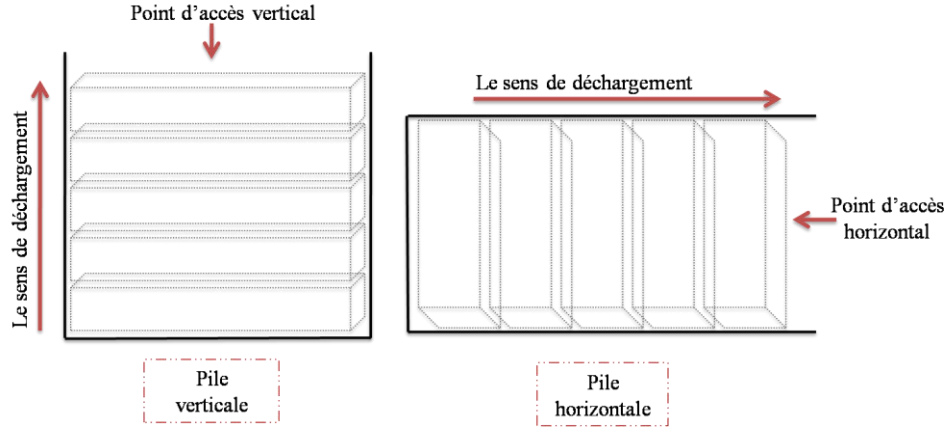


Figure II-2. Point d'accès vertical et horizontal

Une solution de ShTSP est une séquence des clients à servir $c_i, \forall i \in \{1, \dots, N\}$ commençant et se terminant par le dépôt c_0 . Plus précisément, comme illustré dans la **Figure II-3**, un exemple d'une tournée réalisable avec 5 clients peut être représenté comme $(c_0, c_4, c_2, c_1, c_3, c_5, c_0)$. Nous notons que les deux solutions $(c_0, c_4, c_2, c_1, c_3, c_5, c_0)$ et $(c_0, c_5, c_3, c_1, c_2, c_4, c_0)$ sont équivalentes en termes de coût de déplacement total, cependant, elles représentent deux solutions totalement différentes en termes de coût de *shiftings* total, car $d_{ij} = d_{ji}, \forall i, j \in \{0, \dots, N\}$ mais $sh_{ij} \neq sh_{ji}$.

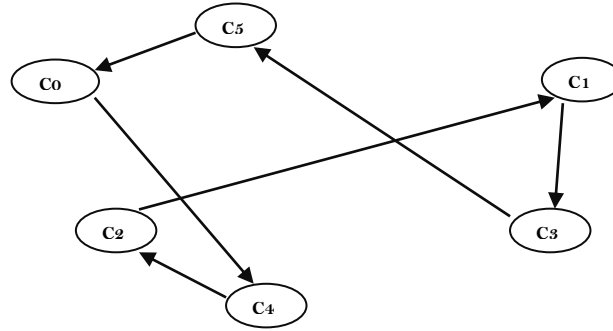


Figure II-3. Illustration d'une tournée de véhicule

Pour bien décrire notre problème, nous considérons un véhicule avec une pile verticale constituée d'un ensemble fini $I = \{1, \dots, N\}$ de N niveaux de placement (**Figure II-4**), chaque niveau $i \in I$ représente un produit p_i à livrer identifié par sa destination c_i . La pile ne peut être accédée que par le haut, de sorte que le déchargement de chaque produit est effectué conformément à la politique LIFO.

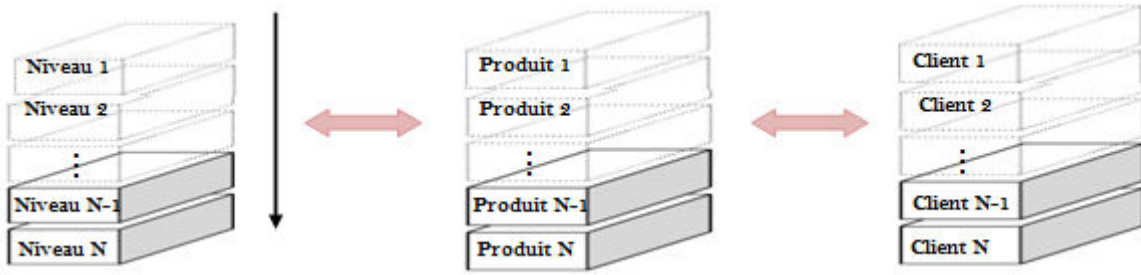


Figure II-4. Piles produits, niveaux, clients

Comme mentionné précédemment, nous supposons que le produit $i \in I$ est destiné à un seul client c_i , donc, à chaque client qui suit demandant une livraison, pour servir le client c_i , nous devons d'abord décharger tous les produits j (non encore desservis) tels que $j < i$ ($i, j \in I$), ce qui signifie, décharger tous les produits qui empêchent le déchargement du produit concerné i . Ensuite, après avoir livré le produit i , les produits déchargés doivent être rechargés dans le véhicule pour continuer le trajet (**Figure II-5**). Chacun de ces mouvements supplémentaires est appelé *shifting*. Plus précisément, un *shifting* est défini comme étant le retrait temporaire de produits qui sont disposés en dessus du produit à livrer et leur placement à nouveau dans le véhicule une fois retiré.

À titre d'illustration, nous considérons un petit exemple où le véhicule doit desservir cinq clients. La **Figure II-5** montre une solution de la tournée qui commence par le client c_5 , nécessitant donc le déchargement des produits aux niveaux 1, 2, 3 et 4, pour servir le client c_5 et ensuite leur rechargement à nouveau dans le même ordre avant de continuer la tournée, et ainsi de suite jusqu'à la fin de la tournée.

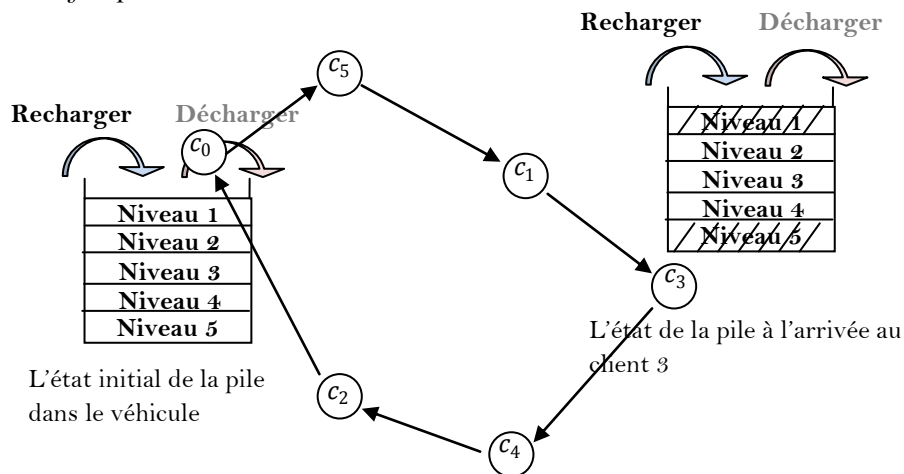


Figure II-5. Exemple d'une solution réalisable du ShTSP

Pour simplifier, nous définissons les deux coûts (le coût distance et le coût *shifting*) en termes d'unité de temps minute. Chacun des coûts de déchargement (u) et de rechargement (r) du produit $i, \forall i \in \{1, \dots, N\}$ est égal à trois minutes. La **Figure II-6** illustre les cinq états de la pile du véhicule une fois arrivé chez un client $c_i, \forall i \in \{1, \dots, N\}$, et ce, dans le même ordre du tour présenté dans la **Figure II-5**. Dans chaque état, le coût *shifting* (dénnoté par Coût_Sh)

correspond à la somme des opérations supplémentaires de chargement et de rechargement des produits bloquants le produit à servir, et le coût de service (dénnoté $Coût_Sv$) correspond à la somme du coût *shifting* et le coût de déchargement du produit concerné. Ainsi, le temps de service total durant la tournée sera égal à 51 minutes dont le temps total de *shifting* est de 36 minutes. Dans cet exemple, nous pouvons constater que plusieurs mouvements improductifs doivent obligatoirement être effectués au niveau de plusieurs clients.

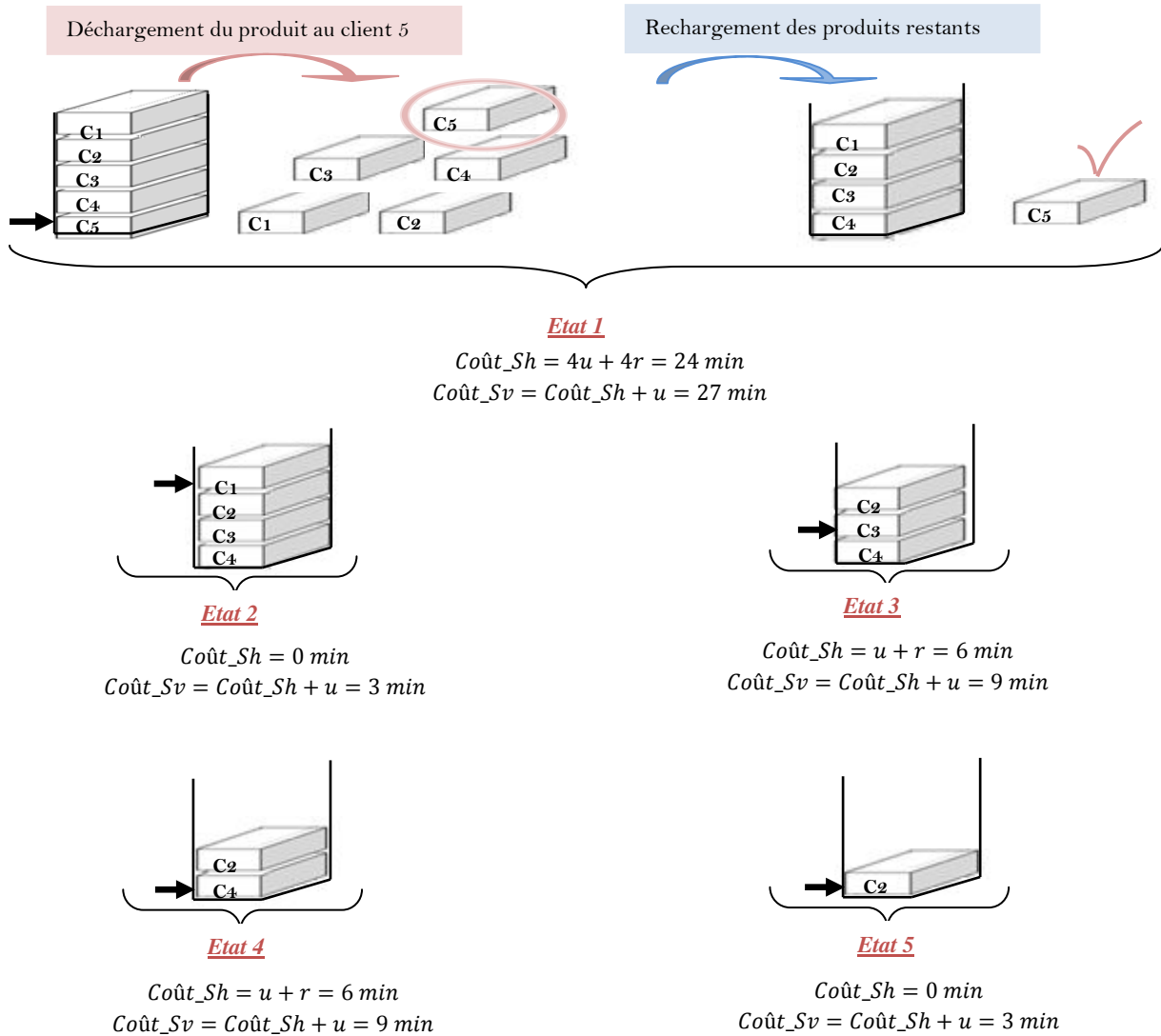


Figure II-6. Illustration de chaque état de la pile pendant toute la tournée

Notons que la distance totale parcourue et le nombre de *shiftings* effectués ne sont pas toujours corrélés, dans le sens qu'ils ne varient généralement pas de la même manière. En effet, minimiser la distance pourrait augmenter le coût total de *shiftings* et vice versa. Dans la **Figure II-7**, nous illustrons deux solutions possibles (a) et (b) qui varient de manière différente. Par rapport à la tournée représentée par la solution (b), la tournée (a) représente une très bonne solution en termes de temps du trajet ($Coût_Dis_{(a)} < Coût_Dis_{(b)}$) :

$Coût_Dis_{(a)} = T(c_0, c_5) + T(c_5, c_1) + T(c_1, c_3) + T(c_3, c_4) + T(c_4, c_2) + T(c_2, c_0) = 27 \min$
 $Coût_Dis_{(b)} = T(c_0, c_1) + T(c_1, c_2) + T(c_2, c_3) + T(c_3, c_4) + T(c_4, c_5) + T(c_5, c_0) = 56 \min$
 $T(c_i, c_j)$ est le temps de trajet requis pour se déplacer entre les deux clients c_i et c_j .
 Toutefois, la tournée (a) représente une solution moins bonne en termes de coûts de *shiftings* par rapport à la solution (b), qui représente la solution optimale en termes de temps de *shiftings*: $Coût_Sh_{(b)} = 0 \min < Coût_Sh_{(a)} = 36 \min$ (voir la **Figure II-6** pour plus de détails de calcul).

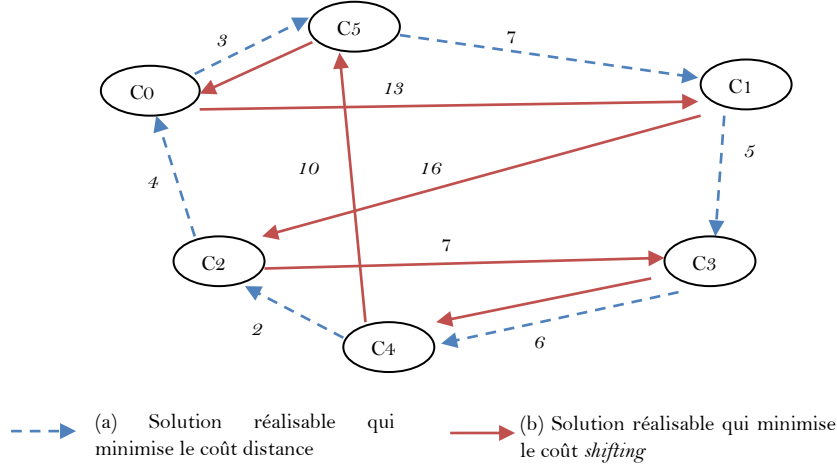


Figure II-7. Les deux objectifs (minimiser le coût distance et minimiser le coût *shifting*) ne varient pas dans le même sens

Dans la plupart des cas, les coûts de service sont soit ignorés ou supposés constants. Cependant, dans la pratique, on peut facilement constater que les coûts de service varient en fonction de plusieurs facteurs qui dépendent naturellement de l'ordre des clients visités lors de la tournée. Dans notre problème, le coût de service requis pour chaque client n'est pas fixé à l'avance, mais dépend de la position de ce client dans la tournée. Nous notons que, parfois, le coût de *shifting* est supérieur au coût de distance, car le service d'un client nécessite plus de temps et d'efforts qu'un simple déplacement le long de l'arc. Ainsi, la principale contribution de ce travail est d'introduire une variante du TSP classique appliqué dans le CSPP afin de déterminer le tour optimal du véhicule, en tenant en compte les mouvements supplémentaires des produits en les déchargeant et les rechargeant chez chaque client visité. Par conséquent, notre objectif est de minimiser le coût total de la tournée d'un seul véhicule avec une seule pile, y compris la distance totale parcourue et le coût total du service, qui est représenté par le coût de *shiftings* causés par le déchargement et le rechargement de tous les produits bloqués dans la pile.

3 Formulation mathématique du ShTSP

Dans cette section, nous présentons une modélisation non linéaire mixte pour résoudre le ShTSP. D'abord, nous présentons quelques hypothèses et terminologies qui seront utilisées dans la formulation ultérieurement.

Les hypothèses de base utilisées dans ce chapitre sont les suivantes:

- ▶ Le plan de chargement du véhicule est connu à l'avance et est donné en fonction de la stabilité et du poids de chaque produit,
- ▶ Le véhicule a exactement une pile verticale composée de plusieurs niveaux de placement,
- ▶ L'ordre de déchargement des produits dans le véhicule doit être effectué par le haut ou par le côté. Les produits ne peuvent être accédés que par le haut, suivant la politique de déchargement dernier entré, premier sorti,
- ▶ Chaque client demande la livraison d'un seul produit. En d'autres termes, le nombre de clients à servir est le même que celui des produits stockés dans le véhicule,
- ▶ Un seul produit peut être rangé dans chaque niveau de la pile et doit être retiré chez le client correspondant.

Les notations suivantes sont utilisées tout au long du chapitre pour formuler le modèle:

- ▶ N : le nombre de clients à visiter (égal au nombre de produits à livrer)
- ▶ M : une constante positive suffisamment grande
- ▶ δ : une constante positive assez petite
- ▶ d_{ij} : la distance entre les clients c_i et c_j
- ▶ u_i : le coût de déchargement du produit i
- ▶ r_i : le coût de rechargement du produit i
- ▶ $b_{ij} = \begin{cases} 1 & \text{si } i > j \\ 0 & \text{sinon} \end{cases}$

Avec ces notations, une formulation mathématique sous forme de programme non linéaire mixte du ShTSP est construite en introduisant les variables ci-dessous, pour $i, j \in \{0, \dots, N\}$:

$$x_{ij} = \begin{cases} 1 & \text{si le produit } j \text{ est livré immédiatement après le produit } i \\ 0 & \text{sinon} \end{cases}$$

$$y_{ij} = \begin{cases} 1 & \text{si le produit } i \text{ est livré avant le produit } j \\ 0 & \text{sinon} \end{cases}$$

Notons que $y_{ij} = 1$ si le produit i est livré « **non nécessairement immédiatement** » avant le produit j .

s_{ij} = coût de *shifting* lors du déchargement du produit j
sachant que le véhicule vient de servir le client c_i

$$s_{0j} = \sum_{i=1}^j u_i + \sum_{i=1}^{j-1} r_i \quad \forall j \in \{1, \dots, N\} \quad (\text{II-1})$$

$$s_{i0} = 0 \quad \forall i \in \{1, \dots, N\} \quad (\text{II-2})$$

Les formules (II-1) définissent le coût de *shifting* au début de la tournée, il est calculé en fonction du coût de déchargement et de rechargement de tous les produits arrimés en haut du produit j en incluant le coût de déchargement de j . Les formules (II-2) indiquent qu'à la fin de la tournée, les frais de *shiftings* sont nuls, c'est-à-dire qu'il n'y a pas d'opération de

déchargement ou de rechargement des produits. Et pour le cas $i \neq 0$ et $j \neq 0$, le coût de *shifting* s_{ij} peut être calculé par les trois contraintes suivantes:

$$s_{ij} = (1 - b_{ij}) \left[\sum_{\substack{k=0 \\ k \neq i \\ k \neq j}}^N s_{ki} x_{ki} - u_i + u_j + \sum_{k=i+1}^{j-1} (u_k + r_k)(1 - y_{kj}) \right] \\ + b_{ij} \left[\left(s_{0i} - \sum_{k=j+1}^{i-1} (u_k + r_k) \right) x_{0i} \right. \\ \left. + \sum_{\substack{k=1 \\ k \neq i \\ k \neq j}}^N \left(s_{ki} - \sum_{v=j+1}^{i-1} (u_v + r_v)(1 - y_{vj}) \right) x_{ki} - u_i - r_j \right] \\ \forall i, j \in \{1, \dots, N\} \quad \text{avec } i \neq j+1, j \neq i+1 \text{ et } i \neq j \quad (\text{II-3})$$

$$s_{ij} = \sum_{\substack{k=0 \\ k \neq i, j}}^N s_{ki} x_{ki} - u_i - r_j \quad \forall i, j \in \{1, \dots, N\} \text{ tel que } i = j+1 \quad (\text{II-4})$$

$$s_{ij} = \sum_{\substack{k=0 \\ k \neq i, j}}^N s_{ki} x_{ki} - u_i + u_j \quad \forall i, j \in \{1, \dots, N\} \text{ tel que } j = i+1 \quad (\text{II-5})$$

Les formules (II-3), (II-4) et (II-5) sont exprimées sous forme récursive; elles indiquent le coût de *shifting* nécessaire pour décharger le produit j sachant que le véhicule vient juste de visiter le client c_i , c'est-à-dire, le voyageur vient de livrer le produit i , de manière à prendre en compte l'historique de tous les clients visités avant c_i selon l'ordre suivi jusqu'à la fin de la tournée partielle. En d'autres termes, elles permettent d'éliminer tous les produits déjà livrés avant le produit j . Ceci dit, nous avons proposé une nouvelle façon de modéliser les coûts de *shifting* nécessaires pour livrer chaque produit i au client correspondant c_i , ces coûts ne sont pas fixés à l'avance mais dépendent de l'ordre des clients visités avant c_i . L'idée clé de cette forme récursive est d'avoir la possibilité de connaître à chaque instant l'historique de chaque visite non encore complétée.

$$\delta x_{ij} \leq s_{ij} \leq M x_{ij} \quad \forall i \in \{0, \dots, N\}, \forall j \in \{1, \dots, N\}, i \neq j \quad (\text{II-6})$$

Pour expliquer les contraintes (II-6), nous supposons que le véhicule visite le client c_j immédiatement après le client c_i , dans ce cas $x_{ij} = 1$, donc le coût de *shifting* pour décharger le produit j est supérieur à une constante positive δ , sinon le coût de *shifting* serait égal à zéro. De plus, compte tenu de la présentation des niveaux de pile illustrés au niveau de la **Figure II-4**, ces constantes positives δ et M peuvent être définies, respectivement, comme le meilleur scénario et le pire scénario parmi les solutions possibles de *shifting*, tels que

$\delta = \min_{i,j \in \{0, \dots, N\}} \{s_{ij}\}$ ce qui correspond au coût minimal de déchargement du produit qui se trouve dans le niveau le plus haut de la pile, autrement dit $\delta = \min_{k \in \{1, \dots, N\}} \{u_k\}$ alors que $M = \max_{i,j \in \{0, \dots, N\}} \{s_{ij}\}$ correspond au coût de déchargement du produit qui se trouve dans le niveau le plus bas de la pile de manière à ce que tous les produits au-dessus ne soient pas encore servis. En termes simples $M = s_{0N}$. Par conséquent, les contraintes (II-6) peuvent être remplacées par les contraintes suivantes (II-6)' :

$$x_{ij} \min_{k \in \{1, \dots, N\}} \{u_k\} \leq s_{ij} \leq x_{ij} s_{0N} \quad \forall i \in \{0, \dots, N\}, \forall j \in \{1, \dots, N\}, i \neq j \quad (\text{II-6})'$$

$$x_{ij} + x_{ji} \leq 1 \quad \forall i, j \in \{0, \dots, N\}, i < j \quad (\text{II-7})$$

Les contraintes (II-7) indiquent que si le produit j est délivré immédiatement après le produit i , alors il faut que $x_{ji} = 0$, ce qui élimine le cas contraire.

$$y_{ij} + y_{ji} = 1 \quad \forall i, j \in \{0, \dots, N\}, i < j \quad (\text{II-8})$$

Les contraintes (II-8) assurent que pour chaque paire de clients c_i et c_j , le véhicule servira l'un de ces clients avant l'autre. En d'autres termes, si le client c_i précède le client c_j lors d'un tour, alors c_j ne peut pas précéder c_i sur ce même tour.

$$y_{ij} \geq x_{ij} \quad \forall i, j \in \{0, \dots, N\}, i \neq j \quad (\text{II-9})$$

Les contraintes (II-9) définissent la relation entre les variables de décision en faisant en sorte que si le client c_i précède directement le client c_j dans un tour ($x_{ij} = 1$), alors c_j ne peut pas précéder c_i dans ce même tour et si $x_{ij} = 0$ alors y_{ij} peut prendre 0 ou 1.

$$y_{ij} \geq y_{ik} + y_{kj} - 1 \quad \forall i, j, k \in \{0, \dots, N\}, i \neq j, j \neq k, k \neq i \quad (\text{II-10})$$

Les contraintes (II-10) imposent au client c_i de précéder le client c_j lors de la tournée, si le client c_k précède le client c_j et que le client c_i précède le client c_k . Elles assurent l'ordre de déchargement des produits des clients c_i , c_j et c_k .

$$\sum_{\substack{i=0 \\ i \neq j}}^N x_{ij} = 1 \quad \forall j \in \{0, \dots, N\} \quad (\text{II-11})$$

$$\sum_{\substack{j=0 \\ j \neq i}}^N x_{ij} = 1 \quad \forall i \in \{0, \dots, N\} \quad (\text{II-12})$$

Les contraintes (II-11) et (II-12) garantissent que chaque client est visité exactement une et une seule fois par le véhicule.

$$\sum_{i \in Q} \sum_{j \in Q} x_{ij} \leq |Q| - 1 \quad \forall Q \text{ sous tour de } \{1, \dots, N\} \quad (\text{II-13})$$

Les contraintes (II-13) éliminent les sous-tours.

$$x_{ij}, y_{ij} \in \{0, 1\} \quad \forall i, j \in \{0, \dots, N\} \quad (\text{II-14})$$

$$s_{ij} \geq 0 \quad \forall i, j \in \{0, \dots, N\} \quad (\text{II-15})$$

Les contraintes (II-14) et (II-15) spécifient les définitions de variables. Notez que les contraintes (II-14) peuvent également indiquer qu'à chaque client un service partiel n'est pas autorisé.

Ainsi, la fonction objectif du modèle mathématique peut être définie comme suit:

$$\text{Min} \left(\sum_{i \neq j}^N x_{ij} d_{ij} \quad , \quad \sum_{i \neq j}^N s_{ij} \right) \quad (\text{II-16})$$

Cette fonction objectif (II-16) vise à minimiser le coût total de la tournée du véhicule qui consiste à minimiser deux fonctions, la distance totale parcourue et le coût total de *shifting* défini par les coûts de déchargement et de rechargement de chaque produit à chaque client du trajet.

4 Résolution du ShTSP

Dans cette section, nous adaptons la méta-heuristique de colonie de fourmis (*Ant Colony Optimization*, ACO) de deux manières différentes pour résoudre le ShTSP. Nous exploitons une colonie de fourmis afin de modifier l'ordre dans lequel les produits sont livrés à leur destination, de sorte que la distance parcourue et l'emplacement de chaque produit influencent la qualité de chaque solution obtenue. Tout d'abord, nous présentons une adaptation de l'algorithme ACO ajusté au ShTSP, puis nous résolvons notre problème avec une méthode naïve exacte qui explore toutes les solutions de manière exhaustive, sur des problèmes test générés aléatoirement, pour prouver l'efficacité de notre adaptation ACO. Aussi, nous nous sommes concentrés sur la manière d'accélérer l'obtention des solutions par un algorithme ACO séquentiel, en proposant un algorithme de colonie de fourmis parallèles adapté à notre problème. Enfin, nous affichons les résultats numériques des nouvelles instances générées à partir de la bibliothèque TSPLIB⁸ pour chaque algorithme.

4.1 Principe de l'algorithme de colonie de fourmis (ACO)

L'algorithme de colonie de fourmis est un processus stochastique qui construit une solution en ajoutant des composants à des solutions partielles. Ce processus prend en compte deux facteurs: le premier est une heuristique sur l'instance du problème et le second est les traces de phéromones qui changent dynamiquement pour répéter l'expérience acquise par les agents. Les fourmis construisent donc des solutions en se déplaçant sur un graphe $G = (V, E)$ où V est l'ensemble des nœuds et E connecte les composantes de V . Les contraintes du problème sont implémentées directement dans les règles de mouvement des fourmis. Ainsi, le concept général de cette méta-heuristique est basé sur la simulation du

⁸ TSPLIB est une bibliothèque d'exemples d'instances pour le TSP et d'autres problèmes connexes de différents types provenant de différentes sources.

comportement de plusieurs agents travaillant ensemble à la recherche d'une meilleure solution en utilisant un moyen naturel de communication qu'est la phéromone. Les lecteurs intéressés par une présentation complète de la méta-heuristique ACO peuvent trouver une référence précieuse et générale dans (Dorigo et Blum, 2005).

4.2 L'adaptation de l'algorithme de colonie de fourmis (ACO)

Pour résoudre le ShTSP, nous adaptons la méta-heuristique ACO selon les étapes suivantes. D'abord, chaque fourmi k commence sa tournée à partir du dépôt c_0 et choisit aléatoirement un client c_i à visiter en premier. Ensuite nous calculons l'information heuristique $(\eta_{ij})_{j \in \{0, \dots, N\}}$ qui correspond au client c_i choisi en utilisant la formule suivante:

$$\eta_{ij} = \begin{cases} \frac{1}{s_{ij} + d_{ij}} & \forall j \in L_i^k \\ 0 & \text{sinon} \end{cases} \quad (\text{II-17})$$

où L_i^k indique la liste des candidats (clients qui ne sont pas encore visités) de la fourmi k au nœud i et s_{ij} est calculé selon les formules (II-3), (II-4) et (II-5). Cette information heuristique est utilisée pour guider le choix des fourmis vers les clients proches (en termes de distance) dont les indices sont aussi petits que possible correspondant aux produits placés en haut de la pile, c'est-à-dire, le coût de déchargement du produit correspondant est faible en terme de *shifting*. Nous préférons les clients qui ont à la fois la plus faible distance de trajet et le plus faible coût de *shifting*. Dans l'étape suivante, nous calculons la probabilité de transition pour choisir le client qui suit parmi l'ensemble des candidats du client choisi c_i (à l'étape précédente du calcul), selon la formule suivante:

$$p_{ij}^k = \begin{cases} \frac{(\tau_{ij})^\alpha \cdot (\eta_{ij})^\beta}{\sum_{l \in L_i^k} (\tau_{il})^\alpha \cdot (\eta_{il})^\beta} & \text{si } j \in L_i^k \\ 0 & \text{sinon} \end{cases} \quad (\text{II-18})$$

Les deux paramètres α et β définissent l'importance relative entre la valeur de la phéromone τ_{ij} et l'information heuristique η_{ij} .

Le dépôt de phéromones peut modifier considérablement le mode de convergence de l'algorithme. D'un point de vue purement naïf, nous pouvons déposer la même quantité de phéromones sur chaque chemin. Les fourmis engagées dans de longs chemins vont déposer moins de phéromones car elles essayeront moins de chemins. Au contraire, les fourmis engagées sur les chemins les plus courts essaieront rapidement d'autres chemins. Naturellement, les chemins les plus courts auront plus de phéromones que d'autres. Cependant, nous pouvons utiliser deux méthodes pour mettre à jour la valeur de la phéromone : nous choisissons la meilleure solution de toutes les fourmis, puis nous effectuons une mise à jour locale de $(\tau_{ij})_{i,j}$ pour chaque arc de la solution choisie. Puis, nous effectuons une mise à jour globale de la phéromone pour tous les arcs du graphe suivant la formule ci-dessous:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (\text{II-19})$$

Où ρ est le coefficient d'évaporation choisi entre $]0,1[$, m est le nombre de fourmis dans la colonie et $\Delta\tau_{ij}^k$ est calculé selon la formule suivante :

$$\Delta\tau_{ij}^k = \begin{cases} \frac{q}{L^k} & \text{si } (i,j) \text{ fait partie de la solution de la fourmi } k \\ 0 & \text{sinon} \end{cases} \quad (\text{II-20})$$

Où L^k est le coût total de la solution k et q est un paramètre fixe de l'algorithme.

L'algorithme 1 indique le squelette général de l'adaptation de la méta-heuristique de la colonie de fourmis séquentielles pour résoudre le ShTSP.

Algorithme 1: ACO-séquentielles pour le ShTSP.

Entrées

```

m ← nombre de fourmis
iterNumber ← nombre d'itérations
τ0 ← la quantité de phéromone initialisée à un nombre réel positif
α, β, ρ, q et iter ← 1,
pour chaque arc  $(i,j)$  faire
     $\tau_{ij} = \tau_0$ 
fin pour
tant que ( $\text{iter} \leq \text{iterNumber}$ ) faire
    pour  $k = 1$  jusqu'à  $m$  faire
        tant que (fourmi  $k$  n'a pas encore terminé toute la construction de la solution) faire
            construire la solution Solk(iter) selon les formules (II-17) et (II-18)
            calculer le coût de Solk(iter)
        fin tant que
    fin pour
    si une meilleure solution est trouvée alors
        mettre à jour la meilleure solution trouvée: Sol*
    fin si
    mettre à jour la quantité de phéromone de Sol*
    pour tout arc  $(i,j)$  faire
        mettre à jour la quantité de phéromone τij selon la formule (II-19)
    fin pour
     $\text{iter} \leftarrow \text{iter} + 1$ 
fin tant que

```

4.3 Algorithme direct

Pour évaluer notre algorithme ACO, nous avons établi plusieurs tests sur différentes instances générées aléatoirement, nous avons ensuite comparé les résultats obtenus avec un algorithme direct naïf (Algorithme 2), qui est une méthode exacte basée sur combinatoricslib. Cette dernière utilise une librairie java qui permet de générer des permutations, combinaisons et autres séquences combinatoires.

Algorithme 2: Algorithme direct pour le ShTSP.

En commençant par le dépôt 0, chercher l'ensemble des permutations possibles pour tous les clients en utilisant `combinatoricslib`

pour chaque permutation p **faire**

calculer le coût de p

si une meilleure solution est trouvée **alors**

Mettre à jour la meilleure solution trouvée: p^*

fin si

fin pour

Le principal inconvénient de cette méthode exacte est qu'elle ne peut être utilisée que pour un nombre de clients assez petit, même avec une machine plus puissante.

4.4 Adaptation de l'algorithme de colonie de fourmis parallèles

Bien qu'il soit important de trouver de bonnes solutions, le temps de calcul reste un indicateur précieux qui doit être pris en compte dans les applications du monde réel. Dans cette section, nous proposons une nouvelle adaptation de la méta-heuristique ACO en utilisant le concept de parallélisme au niveau des fourmis, comme le montre la **Figure II-8**.

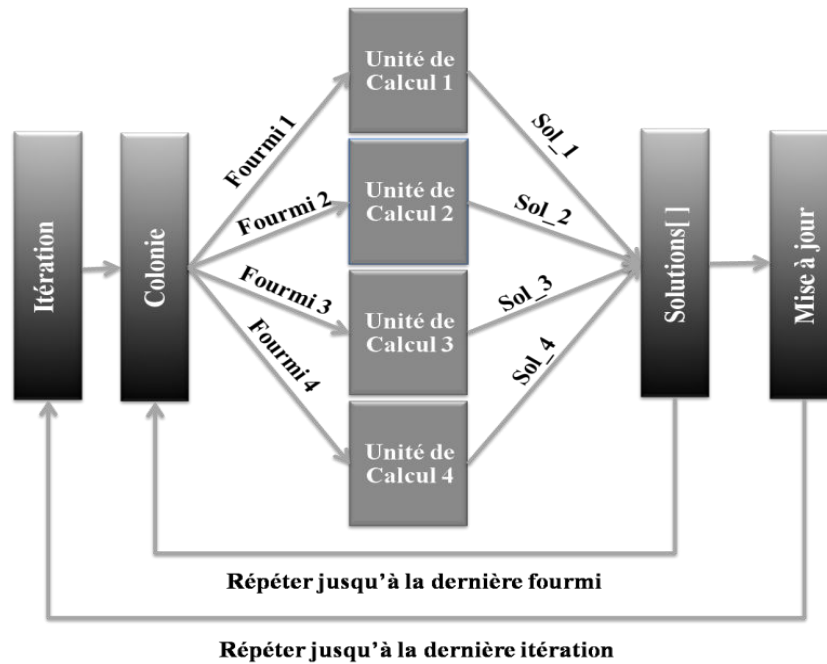


Figure II-8. L'algorithme d'ACO-parallèles

Dans notre contexte, le parallélisme vise à diviser le temps de calcul de chaque colonie en lançant un processus pour chaque fourmi; il permet donc, d'optimiser l'utilisation des ressources informatiques afin d'optimiser l'efficacité de l'algorithme.

Algorithm 3: ACO-parallèles pour le ShTSP.**Entrée**

```

m ← nombre de fourmis
iterNumber ← nombre d'itérations
 $\tau_0$  ← la quantité initiale de phéromone
 $\alpha, \beta, \rho, q$  et iter ← 1
pour tout arc  $(i, j)$  faire
     $\tau_{ij} = \tau_0$ 
fin pour
pour iter = 1 jusqu'à iterNumber faire
    Parallèlement, construire des solutions Sol(iter)[p] pour un ensemble de p fourmis
    jusqu'à la dernière fourmi
    si une meilleure solution est trouvée alors
        mettre à jour la meilleure solution: Sol*
    fin si
    mettre à jour la quantité de phéromone de Sol*
    pour tout arc  $(i, j)$  faire
        mettre à jour la quantité de phéromone  $\tau_{ij}$ 
    fin pour
fin pour

```

4.5 Réglage des paramètres

Les paramètres de l'algorithme sont nombreux, nous citons le nombre m de fourmis, le paramètre ρ d'affaiblissement des phéromones, le paramètre q qui intervient dans le calcul de la quantité de phéromone déposée pour chaque fourmi et les paramètres α et β qui interviennent quant à eux dans le calcul de la probabilité de transition. Nous avons essayé plusieurs valeurs pour chaque paramètre en fixant les autres. Pour le paramètre m , nous avons essayé les valeurs 20, 50, 100 et 200. Pour le paramètre q , nous avons testé les valeurs 1, et 100 et nous avons constaté que les meilleurs résultats ont été obtenus pour $q = 1$. Pour ce qui est du paramètre ρ , les meilleurs résultats ont été obtenus pour $\rho = 0.7$ et $\rho = 0.5$. En ce qui concerne α et β , nous avons constaté que ces deux paramètres jouent un rôle important dans la qualité de la solution, mais nous ne sommes pas arrivés à trouver une valeur optimale qui convient à une classe d'instances toute entière. Nous avons essayé différentes valeurs entre 1 et 10 et nous avons rapporté la meilleure solution retenue.

4.6 Résultats expérimentaux

Dans cette section, nous rapportons les résultats numériques de notre problème résolu avec la méthode exacte, la méta-heuristique ACO et celle ACO-parallèles. Les approches de résolution ont été implémentées avec Java en utilisant NetBeans 8.0. Par défaut, toutes les expériences ont été menées sur un Intel® Core™ i5-4570 CPU @ 3,20 GHz.

4.6.1 Génération des instances

Comme il n'existe pas d'instances de référence dans la littérature qui combine le CSPP et le TSP, nous envisageons de générer des instances adaptées à notre problème. Le ShTSP a du sens dans des contextes où les coûts de distance et les coûts totaux de *shiftings* sont dans la même échelle. Nous avons donc testé nos algorithmes sur des instances présentant cette caractéristique. Nous avons généré de nouveaux benchmarks, basés sur la bibliothèque TSPLIB, dans lesquels les coûts de *shifting* sont comparables aux coûts distances. Chaque ensemble de problèmes est généré avec différents coûts de déchargement/rechargement définis sur 3, 7, 10, 12, 15, 20, 25 et 30 unités de temps par produit. Notez qu'une unité de temps équivaut à une unité de distance euclidienne. TSPLIB est disponible sur l'adresse: <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>. Les informations indiquant: le nom de l'instance originale extraite de la bibliothèque TSPLIB (*Inst_TSPLIB*) ainsi que le nombre de clients ($|V|$), le type de l'instance (*Type*), les coûts de déchargement et de rechargement ($U \times R$) et les nouveaux noms des instances adaptées au ShTSP (*Inst_Sh*), sont données dans les deux tableaux, **Tableau II-1** et **Tableau II-2**.

Tableau II-1. Description des instances avec petit $|V|$

<i>Inst_TSPLIB</i>	$ V $	<i>Type</i>	$U \times R$	<i>Inst_Sh</i>
gr17	17	Matrix	15x15	gr17_sh15
			25x25	gr17_sh25
			30x30	gr17_sh30
gr21	21	Matrix	15x15	gr21_sh15
			25x25	gr21_sh25
			30x30	gr21_sh30
gr24	24	Matrix	7x7	gr24_sh7
			10x10	gr24_sh10
			12x12	gr24_sh12
fri26	26	Matrix	7x7	fri26_sh7
			10x10	fri26_sh10
swiss42	42	Matrix	3x3	swiss42_sh3
			7x7	swiss42_sh7
dantzig42	42	Matrix	3x3	dantzig42_sh3
			7x7	dantzig42_sh7
hk48	48	Matrix	15x15	hk48_sh15
			25x25	hk48_sh25
			30x30	hk48_sh30
gr48	48	Matrix	10x10	gr48_sh10
			12x12	gr48_sh12
			15x15	gr48_sh15
			25x25	gr48_sh25
eil51	51	EUD_2D	3x3	eil51_sh3
			7x7	eil51_sh7
berlin52	52	EUD_2D	15x15	berlin52_sh15
			25x25	berlin52_sh25
			30x30	berlin52_sh30

Le **Tableau II-1** résume les instances de petite et moyenne taille. La première colonne indique le nom des instances du TSPLIB utilisées pour créer de nouvelles instances, *Inst_Sh*, adaptées à notre problème en ajoutant les coûts de déchargement (U) et de rechargement (R)

qui sont indiqués en minutes dans la quatrième colonne. Le **Tableau II-2** décrit les grandes instances suivant la même structure du **Tableau II-1**.

Tableau II-2. Description des instances avec grand $|V|$

<i>Inst_TSPLIB</i>	<i>/V/</i>	<i>Type</i>	<i>UxR</i>	<i>Inst_Sh</i>
st70	70	EUD_2D	3x3	st70_sh3
eil76	76	EUD_2D	3x3	eil76_sh3
rat99	99	EUD_2D	3x3	rat99_sh3
kroA100	100	EUD_2D	3x3	kroA100_sh3
			7x7	kroA100_sh7
			10x10	kroA100_sh10
			12x12	kroA100_sh12
			15x15	kroA100_sh15
kroB100	100	EUD_2D	20x20	kroA100_sh20
			3x3	kroB100_sh3
			7x7	kroB100_sh7
			10x10	kroB100_sh10
			12x12	kroB100_sh12
kroC100	100	EUD_2D	15x15	kroB100_sh15
			20x20	kroB100_sh20
			3x3	kroC100_sh3
			7x7	kroC100_sh7
			10x10	kroC100_sh10
kroD100	100	EUD_2D	12x12	kroC100_sh12
			15x15	kroC100_sh15
			20x20	kroC100_sh20
			3x3	kroD100_sh3
			7x7	kroD100_sh7
kroE100	100	EUD_2D	10x10	kroD100_sh10
			12x12	kroD100_sh12
			15x15	kroD100_sh15
			20x20	kroD100_sh20
			3x3	kroE100_sh3
eil101	26	EUD_2D	7x7	kroE100_sh7
			10x10	kroE100_sh10
			12x12	kroE100_sh12
			15x15	kroE100_sh15
			20x20	kroE100_sh20
lin105	105	EUD_2D	3x3	eil101_sh3
			3x3	lin105_sh3
			7x7	lin105_sh7
			10x10	lin105_sh10
			12x12	lin105_sh12
			15x15	lin105_sh15

De nouvelles instances supplémentaires ont été générées aléatoirement pour étudier les performances de notre algorithme ACO et pour comparer son efficacité par rapport à la méthode directe. Ces instances sont divisées en trois classes générées aléatoirement et distinguées par le coût du déchargement et du rechargement d'un seul produit dans le véhicule.

- ▶ c1: le coût de déchargement et de rechargement d'un seul produit $U=R=5\text{min}$
- ▶ c2: le coût de déchargement et de rechargement d'un seul produit $U=R=10\text{min}$
- ▶ c3: le coût de déchargement et de rechargement d'un seul produit $U=R=15\text{min}$

Chaque classe se compose de trois groupes distingués par le nombre de clients à servir ($|V| = 5$, $|V| = 7$ et $|V| = 10$).

4.6.2 Résultats numériques et discussion

Les résultats des instances des trois classes c1, c2 et c3 sont présentés avec une description détaillée dans le **Tableau II-3**. Les trois premières colonnes spécifient les attributs des groupes d'instances: le nom de la classe (*Class*), le nombre de clients à visiter ($|V|$) et ($[min, max]$) qui donne les valeurs des distances de chaque groupe. Chaque ligne représente les résultats obtenus pour des instances constituant un groupe dans une classe. La méthode directe donne les solutions présentées dans la colonne (*Méthode Directe*). Nous donnons le trajet explicite de chaque solution, sans considérer le retour au dépôt, dans la quatrième colonne (*DTourPath*), le coût de *shifting* dans la cinquième colonne (*DSh_C*), le coût de service total dans la sixième colonne (*DSv_C*) qui est exprimé par une sommation du coût de *shifting* et le coût de déchargement obligatoire dans le cas idéal (c'est-à-dire, le coût de déchargement de chaque produit à son client en supposant qu'il est placé en haut de la pile), $DSv_C = DSh_C + U * |V| - 1$, le coût distance dans la septième colonne (*DTrav_C*) et le coût total de la solution dans la huitième colonne (*DTotal_C*). Alors que les résultats de l'adaptation de l'algorithme ACO sur les mêmes instances générées sont résumés dans la colonne (*ACO*), nous donnons le chemin explicite (*ATourPath*), le coût total de la solution (*ATotal_C*) en utilisant 50 fourmis et le CPU correspondant à l'exécution de l'algorithme en millisecondes (*ACPU*). Enfin, nous montrons l'écart de notre adaptation ACO par rapport à la méthode exacte (*Méthode Directe*) dans la dernière colonne (*Gap %*) donné par $Gap = \frac{ATotal_C - DTotal_C}{DTotal_C} \times 100$.

Tableau II-3. Résultats numériques des classes c1, c2 et c3 trouvés par l'ACO et comparés par la Méthode Directe

Class	V	[min,max]	Méthode Directe					ACO			Gap %
			DTourPath	DSh_C	DSv_C	DTrav_C	DTotal_C	ATourPath	ATotal_C	ACPU (ms)	
c1	5	[3,50]	[0, 1, 2, 3, 4]	0	20	89	109	[0, 1, 2, 3, 4]	109	46	0,00
	7	[5,70]	[0, 1, 6, 2, 4, 3, 5]	50	80	97	177	[0, 1, 2, 4, 3, 5, 6]	186	54	5,08
	10	[7,100]	[0, 2, 6, 1, 9, 3, 8, 4, 5, 7]	130	175	180	355	[0, 2, 4, 1, 6, 3, 8, 9, 5, 7]	387	94	9,01
c2	5	[9,90]	[0, 1, 3, 2, 4]	20	60	131	191	[0, 1, 3, 2, 4]	191	39	0,00
	7	[10,140]	[0, 4, 6, 1, 3, 2, 5]	160	220	222	442	[0, 4, 6, 1, 3, 2, 5]	442	54	0,00
	10	[15,200]	[0, 1, 6, 2, 3, 5, 4, 7, 8, 9]	100	190	457	647	[0, 1, 6, 2, 3, 5, 4, 7, 8, 9]	647	102	0,00
c3	5	[10,140]	[0, 1, 2, 4, 3]	30	90	234	324	[0, 1, 2, 4, 3]	324	31	0,00
	7	[20,200]	[0, 2, 1, 4, 3, 5, 6]	60	150	312	462	[0, 2, 1, 4, 3, 5, 6]	462	54	0,00
	10	[30,280]	[0, 1, 2, 3, 7, 4, 5, 8, 6, 9]	120	255	700	955	[0, 1, 2, 3, 7, 4, 5, 8, 6, 9]	955	101	0,00

En se référant au **Tableau II-3**, nous pouvons conclure que l'algorithme ACO donne de très bons résultats pour les instances de petites tailles. L'ACO donne des solutions optimales pour 7 instances parmi 9 indiquées en gras au niveau de la colonne (*Gap*), et pour les deux instances restantes l'ACO s'est éloigné de l'optimalité par une moyenne de 7.04 %. Par conséquent, nous pouvons espérer que notre algorithme donnera des solutions satisfaisantes pour les instances de plus grande taille.

Bien qu'il soit important de trouver de bonnes solutions, le temps de calcul est un indicateur très important qui doit être pris en compte dans les applications du monde réel. Dans le **Tableau II-4**, nous montrons les résultats numériques de l'adaptation de l'algorithme ACO et de l'algorithme ACO-parallèles obtenus en résolvant nos nouvelles instances présentées dans le **Tableau II-1**. Dans le **Tableau II-5**, nous donnons les résultats numériques pour des instances plus grandes qui sont présentées dans le **Tableau II-2**. Notez que chaque ligne représente la meilleure solution en cinquante exécutions. Dans la colonne (*ACO*), ainsi que dans la colonne (*ACO-parallèles*), nous présentons les résultats obtenus en utilisant 50 et 100 fourmis, nous rapportons le coût total de la solution (*TCost*) et le temps CPU en millisecondes (*CPU*).

Tableau II-4. Résultats numériques de l'adaptation de l'ACO et de l'ACO-parallèles sur les instances du **Tableau II-1**

<i>Instance</i>	<i>ACO</i>				<i>ACO- Parallèles</i>			
	<i>50 fourmis</i>		<i>100 fourmis</i>		<i>50 fourmis</i>		<i>100 fourmis</i>	
	<i>TCost</i>	<i>CPU (ms)</i>	<i>TCost</i>	<i>CPU (ms)</i>	<i>TCost</i>	<i>CPU (ms)</i>	<i>TCost</i>	<i>CPU (ms)</i>
gr17_sh15	3374	203	3374	391	3374	15	3374	30
gr17_sh25	3878	203	3878	281	3878	15	3878	28
gr17_sh30	4231	141	4231	297	4231	15	4231	28
gr21_sh15	4927	219	4927	422	4927	22	4927	44
gr21_sh25	5531	203	5531	422	5531	22	5531	44
gr21_sh30	5858	218	5858	422	5858	21	5858	43
gr24_sh7	2431	266	2431	532	2431	30	2431	59
gr24_sh10	2847	281	2847	531	2847	30	2847	59
gr24_sh12	3084	266	3084	515	3084	29	3084	59
fri26_sh7	1134	297	1134	593	1134	33	1134	63
fri26_sh10	1209	297	1209	594	1209	33	1209	63
swiss42_sh3	2630	828	2630	1641	2630	96	2630	190
swiss42_sh7	3019	843	3019	1625	3019	95	3019	187
dantzig42_sh3	839	812	839	1641	839	94	839	187
dantzig42_sh7	966	813	966	1640	966	94	966	187
hk48_sh15	23737	1047	23060	2109	23144	131	23060	263
hk48_sh25	26063	1062	26063	2079	26063	132	26063	262
hk48_sh30	27942	1047	27942	2078	27942	132	27942	261
gr48_sh10	11488	1094	11488	2157	11488	127	11488	253
gr48_sh12	11680	1109	12050	2187	11680	127	11680	253
gr48_sh15	12245	1078	12245	2157	12245	126	12245	252
gr48_sh25	15173	1078	15173	2187	15334	126	15173	251
eil51_sh3	1293	1203	1293	2344	1293	142	1293	282
eil51_sh7	1673	1188	1673	2359	1673	142	1673	281
berlin52_sh15	19419	1218	19419	2531	19419	156	19419	308
berlin52_sh25	19600	1266	19600	2516	19600	156	19600	307
berlin52_sh30	19792	1250	19792	2516	19792	155	19792	309

Selon les résultats obtenus dans le **Tableau II-4**, nous pouvons confirmer que, pour les petites instances, lorsque le nombre de fourmis augmente, dans les deux algorithmes ACO et ACO-parallèles, nous obtenons de meilleurs résultats, en termes de $TCost$, mais seulement pour quelques instances (représenté en gras : **hk48_sh15**, **gr48_sh12** et **gr48_sh25**). Cependant, dans le **Tableau II-5**, qui représente les résultats des instances de plus grande taille, nous obtenons de meilleurs résultats pour presque tous les cas étudiés (représentés en gras). De plus, lorsque le nombre de fourmis augmente, les performances de l'ACO-parallèles augmentent également, en termes de temps d'exécution, pour les instances de petite taille et de grande taille.

Tableau II-5. Résultats numériques de l'adaptation de l'ACO et de l'ACO-parallèles sur les instances du **Tableau II-2**

<i>Instance</i>	<i>ACO</i>				<i>ACO- Parallèles</i>			
	<i>50 fourmis</i>		<i>100 fourmis</i>		<i>50 fourmis</i>		<i>100 fourmis</i>	
	<i>TCost</i>	<i>CPU (ms)</i>	<i>TCost</i>	<i>CPU (ms)</i>	<i>TCost</i>	<i>CPU (ms)</i>	<i>TCost</i>	<i>CPU (ms)</i>
st70_sh3	2568	3406	2568	6905	2568	661	2595	1320
eil76_sh3	2133	4125	2133	8187	2133	785	2133	1573
rat99_sh3	2792	7016	2792	14033	2792	1390	2792	2777
kroA100_sh3	38888	7187	38888	14015	38888	1439	38888	2882
kroA100_sh7	53538	7188	53538	13805	53616	1434	53538	2871
kroA100_sh10	57851	7234	57851	13985	58839	1432	57851	2863
kroA100_sh12	62236	7188	61855	14187	61855	1431	61198	2862
kroA100_sh15	66450	7188	66773	14126	66450	1430	66280	2863
kroA100_sh20	71414	7312	71414	14328	71414	1430	72232	2856
kroB100_sh3	40362	7251	40362	14469	40362	1437	40362	2870
kroB100_sh7	49989	7203	49836	14375	49836	1434	49836	2865
kroB100_sh10	59727	7250	59396	14391	59396	1433	59396	2860
kroB100_sh12	61399	7219	61399	14391	61624	1431	61399	2859
kroB100_sh15	66425	7203	66425	14422	66425	1431	66425	2856
kroB100_sh20	73814	7203	73323	14312	73814	1429	73323	2853
kroC100_sh3	37593	7234	37593	14485	37593	1436	37593	2869
kroC100_sh7	49428	7219	48990	14406	49428	1433	48990	2864
kroC100_sh10	56505	7219	56505	14484	56505	1432	56505	2860
kroC100_sh12	58923	7203	57266	14407	58923	1431	57266	2859
kroC100_sh15	63884	7203	63396	14406	63884	1430	63396	2856
kroC100_sh20	70809	7235	70809	14360	70809	1430	70809	2855
kroD100_sh3	39257	7219	38858	14375	39257	1435	38858	2870
kroD100_sh7	52028	7219	51424	14344	52028	1433	51424	2862
kroD100_sh10	58029	7234	58029	14406	58029	1432	58029	2859
kroD100_sh12	61187	7219	61187	14391	61187	1431	61187	2859
kroD100_sh15	64584	7203	64033	14344	64584	1430	64033	2856
kroD100_sh20	71203	7219	71203	14328	71203	1428	71203	2853
kroE100_sh3	40904	7204	40852	14375	40904	1438	40852	2871
kroE100_sh7	50961	7234	50255	14375	50961	1433	50255	2863
kroE100_sh10	59157	7219	59157	14407	59157	1430	59067	2860
kroE100_sh12	64329	7265	63541	14344	64329	1430	63541	2860
kroE100_sh15	65348	7235	63621	14344	65348	1433	63621	2859
kroE100_sh20	74006	7203	73853	14390	73853	1430	73853	2855
eil101_sh3	2315	7343	2236	14625	2315	1453	2236	2902
lin105_sh3	23193	7984	23151	15922	23193	1587	23151	3173
lin105_sh7	29317	7954	28916	15875	29317	1587	28916	3171
lin105_sh10	29737	7953	29737	15875	29737	1585	29737	3167
lin105_sh12	31151	7953	30268	15891	31151	1584	30268	3165
lin105_sh15	34907	7938	34482	15906	34907	1584	34482	3165

En outre, nous présentons une comparaison en termes du temps CPU, des deux approches ACO et ACO-parallèles, appliquées aux instances du **Tableau II-1** en utilisant 100 fourmis (**Figure II-9**). Nous notons que ces résultats ont été obtenus sur un Intel® Core™ i5-4570 CPU @ 3,20 GHz dotée de quatre unités de calcul. Alors que dans la **Figure II-10**, nous avons testé nos algorithmes dans un Intel® Core™ i7-4770 CPU @ 3,40 GHz qui a huit unités de calcul. Les deux diagrammes (**Figure II-9** et **Figure II-10**) confirment la performance du concept de parallélisme dans notre algorithme en montrant le grand écart entre le temps de calcul de la version séquentielle (ACO) et celle parallèle (ACO-parallèles). En particulier, une diminution importante du temps de calcul dans l'ACO-parallèles est observée à partir de l'instance **swiss42_sh3** avec 42 clients, spécifiquement lors de l'utilisation de huit unités de calcul.

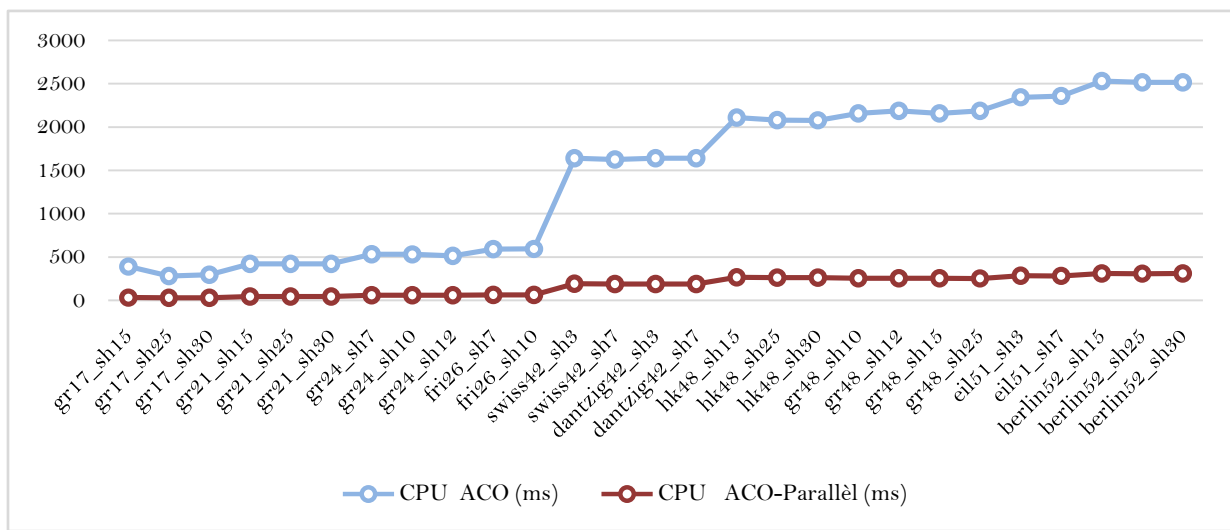


Figure II-9. Comparaison du temps CPU de l'ACO et de l'ACO-parallèles, les deux appliqués aux instances du **Tableau II-1** avec 100 fourmis en utilisant quatre unités de calcul

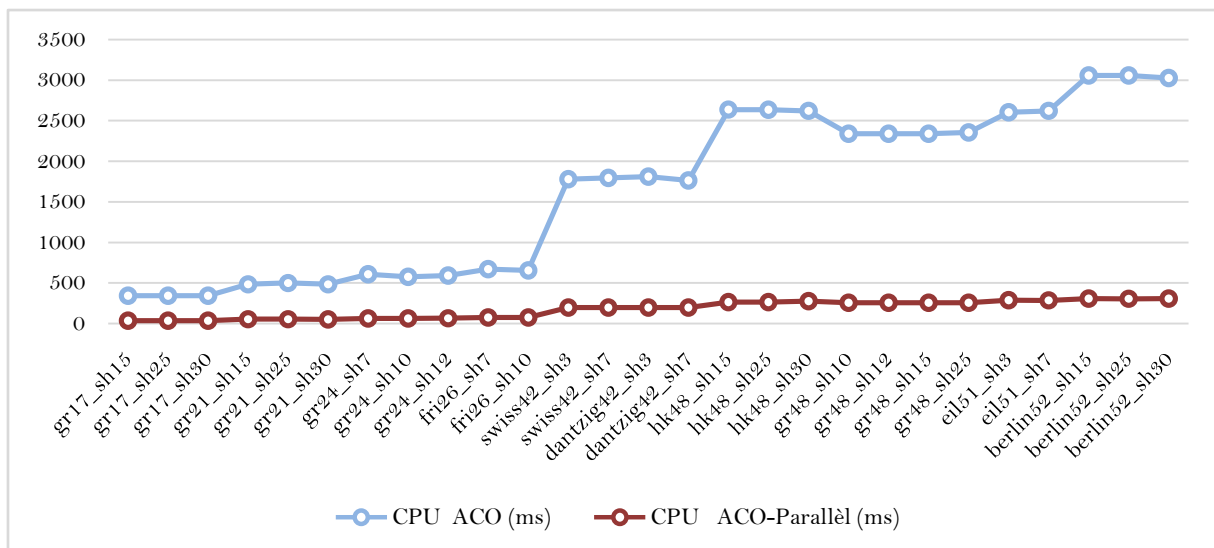


Figure II-10. Comparaison du temps CPU de l'ACO et de l'ACO-parallèles, les deux appliqués aux instances du **Tableau II-1** avec 100 fourmis en utilisant huit unités de calcul

Pour montrer la relation entre les performances de l'algorithme ACO-parallèles et le nombre de unités de calcul, nous avons tracé, sur la **Figure II-11**, la progression du temps CPU de l'ACO-parallèles adapté aux instances du **Tableau II-2** avec 100 fourmis en utilisant quatre unités de calcul, puis, huit unités de calcul. Nous pouvons confirmer que l'algorithme ACO-parallèles montre plus d'efficacité lorsqu'il est exécuté dans une machine avec plus d'unités de calcul.

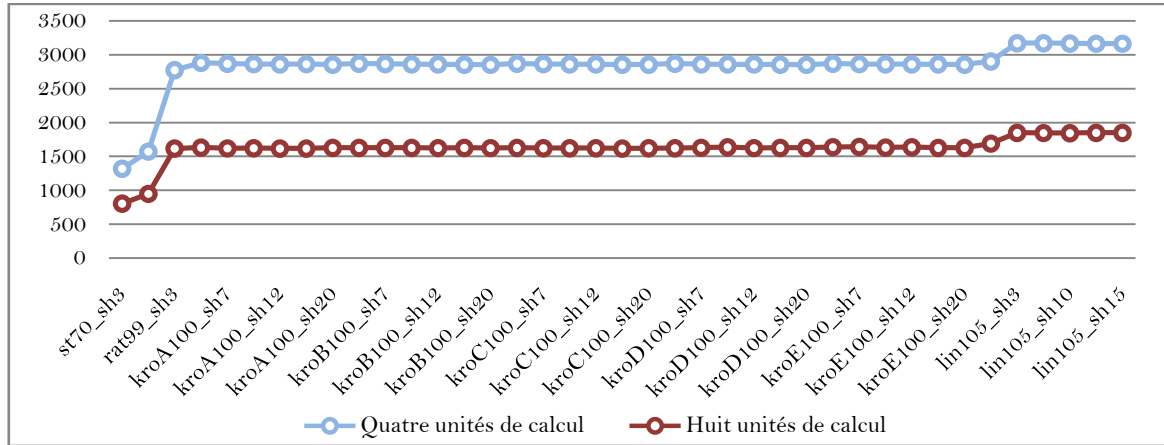


Figure II-11. Comparaison du temps CPU (ms) de l'ACO-parallèles adapté aux instances du **Tableau II-2** avec 100 fourmis en utilisant quatre puis huit unités de calcul

Pour conclure et en se référant à tous les résultats donnés ci-dessus, nous pouvons confirmer que l'algorithme ACO-parallèles donne d'excellents résultats en moyenne pour toutes les instances par rapport à l'algorithme ACO séquentiel. Cela nous permet de réduire le temps de calcul de plus de 80%, en particulier pour les grandes instances. De plus, nous confirmons que la variation du temps CPU est fortement corrélée à la variation du nombre des unités de calcul dans la machine.

5 Conclusion

Dans ce chapitre, nous avons introduit le *shifting* dans le problème de voyageur de commerce (*Shifting Traveling Salesman Problem*, (ShTSP)), le transport d'un véhicule avec une seule pile composée d'un ensemble fini de niveaux avec prise en compte du coût *shifting*. Nous avons défini ce problème en tant que nouvelle variante du problème de voyageur de commerce (TSP) associé au problème de chargement et d'arrimage des conteneurs (CSPP). Etant donné un plan de chargement généré selon plusieurs critères, le ShTSP consiste à déterminer le parcours optimal du véhicule en tenant compte des mouvements supplémentaires des produits chargés causés par la politique LIFO. Nous avons proposé un modèle mathématique non linéaire mixte, puis nous avons adapté l'algorithme de colonie de fourmis séquentielles et l'algorithme de colonies de fourmis parallèles pour le résoudre. En fin, nous avons montré que la version parallèle de l'algorithme des colonies de fourmis est très efficace en termes de temps d'exécution relativement à sa version séquentielle. Le prochain chapitre sera réservé à l'optimisation des plans de chargement des conteneurs dans le cas du transport fluvial tout en sachant la tournée à l'avance.

Chapitre III

Optimisation du problème de chargement et d'arrimage des conteneurs avec coût de *shifting* non-uniforme : le cas d'une seule barge

1 Introduction

Avec l'augmentation des flux de fret et des problèmes de congestion routière et ferroviaire, le transport fluvial se présente comme une alternative intéressante pour résoudre les problèmes liés aux coûts de transport et ceux liés à la pollution, et cela, lorsque la zone de transport est dotée d'une liaison fluviale ou maritime. Pour permettre la fluidité du transport fluvial des conteneurs, de règles et stratégies strictes concernant les services des barges et les processus de manutention des conteneurs dans les ports sont primordiales. Dans ce qui suit, nous désignerons par transport par barge, le transport fluvial et même le transport fluvio-maritime.

Dans les ports à conteneurs, le retard des barges peut survenir à différentes zones et peut être causé par plusieurs facteurs (Tanaka et Tierney, 2018). Or, les principales activités affectant le temps que passe la barge dans le quai du terminal fluvial sont les processus de déchargement et de chargement des conteneurs. Bien qu'il existe une relation entre les deux processus, ils sont essentiellement réalisés en deux tâches indépendantes, le chargement étant le plus difficile et le plus sensible à l'efficacité des opérations dans les ports ultérieurs (Imai et al., 2002). En effet, un bon plan de chargement dans le port initial permet la diminution du temps de déchargement des conteneurs dans leur port de destination, et par conséquent, la réduction du temps global du transport. Un autre facteur crucial à prendre en considération lors de la planification des plans de chargement est celui de la stabilité de la barge.

Les conteneurs sont empilés selon le principe « dernier entré, premier sorti », de telle sorte que pour décharger un certain conteneur, il est nécessaire de décharger tous les conteneurs qui se trouvent au-dessus, puis les recharger à nouveau, générant ainsi des mouvements « *shifting* » qui entraînent des frais supplémentaires ainsi qu'une perte de temps dans le port. Le problème qui se pose est donc de déterminer l'arrangement le plus convenable possible des conteneurs dans la barge afin de minimiser à la fois le nombre de *shiftings* et maximiser les mesures de stabilité. Nous notons le problème dans le cas d'une seule barge par le 3D-

BCSPP, (*3D-Barge Container Stowage Planning Problem*), problème d'optimisation des plans de chargement et d'arrimage des conteneurs dans la barge.

Dans ce chapitre, nous nous intéressons à un cas inspiré par des problèmes réels, où les coûts de chargement/déchargement sont non-uniformes, c.-à-d., ils diffèrent d'un port à un autre. En effet, chaque déplacement de conteneur via une grue de quai induit une unité de frais de déplacement qui dépend de la politique de charge du terminal à conteneurs local. Les études précédentes supposent que chaque mouvement de conteneur consomme un coût uniforme dans tous les ports et se concentrent donc sur la minimisation du nombre total de *shiftings* ou du temps de rotation du navire (Delgado et al., 2012 ; Parreño et al., 2016 ; Ambrosino et al., 2017). À notre connaissance, le seul travail qui a considéré le cas des coûts non-uniformes, est celui de Zhang et al., (2018) ; cependant, les auteurs ont traité le problème de planification de l'arrimage en considérant des conteneurs de même taille (20 pieds) à charger dans une seule baie de navire pour un itinéraire de transport à ports multiples, dans le but de minimiser les frais de *shiftings* totaux.

L'objectif de ce chapitre est de traiter le problème de *shifting* dans le cas non-uniforme en considérant les deux objectifs : minimiser le coût total de *shiftings* et maximiser la stabilité de la barge en considérant les mesures de la stabilité longitudinale et transversale. Le cas où le plan de chargement et d'arrimage des conteneurs appliqué à un système multi-barges avec un coût de *shifting* uniforme fera l'objet du chapitre IV. Le 3D-BCSPP est un problème NP-difficile, puisqu'il généralise le CSPP classique qui est lui-même NP-difficile (Avriel et al., 2000).

Le reste de ce chapitre est organisé comme suit. La deuxième section propose une brève étude comparative sur la performance de la durabilité actuelle des principaux modes de transport. La troisième section décrit le problème formellement et identifie ses différentes caractéristiques, notamment, la structure physique de la barge, la politique du « dernier entré, premier sorti », la notion des frais non-uniformes dans les ports, la notion de stabilité, de fragilité et de taille mixte des conteneurs. Ensuite, un modèle mathématique approprié est présenté dans la quatrième section. Puis, une série d'exemples pratiques basée sur des données réelles résolue par le solveur CPLEX est présentée dans la cinquième section afin de valider le modèle proposé. Enfin, avant de conclure, une étude numérique comparative est rapportée en analysant l'influence de la nature de l'instance et des frais de *shiftings* (dans le cas non-uniforme et uniforme) sur la planification de l'arrimage.

2 Transport vert : transport par barge comme alternative écologique aux autres modes du transport intermodal

En tant que mode massifié, la voie fluviale dispose aussi d'avantages importants par rapport à la route, souvent communs avec le rail (Van Slobbe, 2002 ; Frémont et al., 2009) :

- Face à des trafics importants, elle permet d'éviter des phénomènes de congestion dans les agglomérations portuaires, sur les corridors qui desservent l'*hinterland*, ainsi que dans les agglomérations de l'*hinterland*,

- ▶ Avec des coûts fixes élevés et des coûts variables faibles, la voie fluviale peut offrir des coûts par kilomètre et à l'EVP. Par rapport à la route et au rail, la voie d'eau est le mode qui offre la plus grande efficacité énergétique et les coûts les plus faibles de la tonne au kilomètre. Un convoi fluvial peut transporter de 250 à 500 EVP contre 60 à 90 pour le rail et 2 seulement pour la route,
- ▶ C'est un mode fiable. Les infrastructures ne sont pas saturées et le plus souvent dédiées aux transports de marchandises, éliminant ainsi le risque d'accidents. La marchandise est déposée sur des barges qui empruntent des canaux de navigation naturels ou artificiels,
- ▶ Face à la route, dont les externalités négatives sont de moins en moins acceptées, elle permet de diminuer considérablement l'empreinte environnementale du transport des marchandises. La barge est une alternative modale pour répondre à la pression environnementale (ECMT, 2006).

L'utilisation des barges comme moyen de transport intermodal est donc fortement favorisée dans les régions avec accès facile au réseau de voies navigables. Cependant, le processus de transport des conteneurs par barges n'est toujours pas optimal, car ces dernières passent souvent plus de temps que nécessaire au port. Pour que le transport de marchandises par voie navigable coûte moins que les autres modes de transport, il faut impérativement optimiser leurs opérations de manutention.

3 Description du problème d'optimisation des plans de chargement et d'arrimage des conteneurs dans la barge

Étant donné une barge avec sa tournée bien définie, décrite par une séquence de ports à visiter ayant des frais de déchargement/rechargement non-uniformes, le problème 3D-BCSPP consiste à définir le plan d'arrimage pour un ensemble donné de conteneurs, qui diffèrent par la taille, le poids, la fragilité et par le port de destination ; tous les conteneurs doivent être chargés à bord, de telle sorte que l'instabilité de la barge et le montant total des frais de déchargement soient minimaux.

3.1 Structure physique de la barge

La barge à conteneurs est une forme de transport intérieur à fond plat où les conteneurs sont empilés et remorqués vers leur destination via des voies navigables intérieures. La **Figure III-1** représente un plan schématisé pour une barge à conteneurs dans lequel les vues d'en-haut, transversale et longitudinale sont représentées.

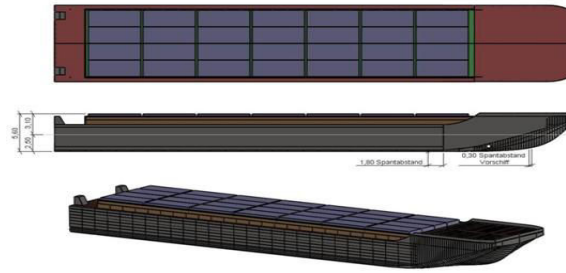


Figure III-1. Vues d'en-haut, transversale et longitudinale

Cette représentation figure une structure cellulaire qui est organisée en un système de trois dimensions appelé BAROTI (en anglais, *Bay, Row, Tier*). Le lecteur pourra se référer à (Ambrosino et al., 2004) pour une description bien détaillée. Ce système est principalement conçu pour améliorer la fonction de stockage des conteneurs, qui, cependant, impose une forte restriction dans l'ordre de chargement/déchargement des conteneurs transportés (Imai et al., 2002 ; hu et Cai, 2017). Dans le système BAROTI, la barge est divisée en baies. Chaque baie est ensuite organisée en piles de conteneurs. Chaque pile a une hauteur définie par un nombre maximal de niveaux. La **Figure III-2** montre une illustration du système BAROTI de la barge.

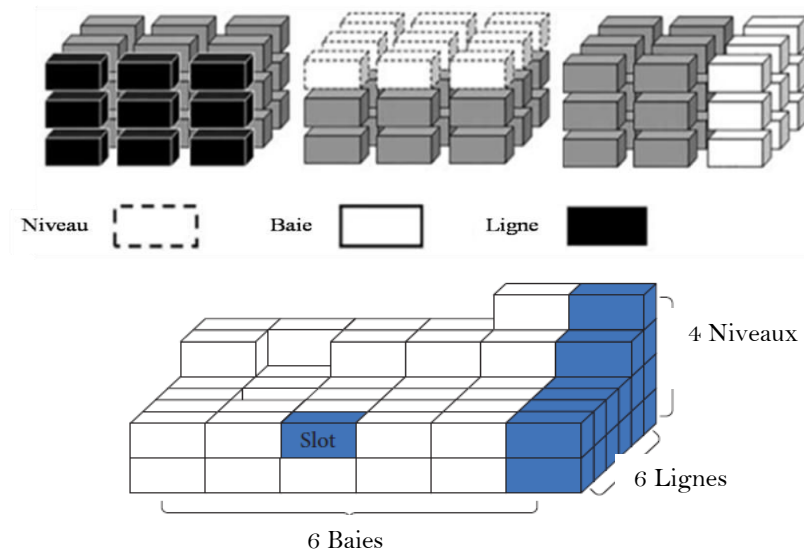


Figure III-2. Le système BAROTI

Dans la suite de ce chapitre, nous considérons une barge constituée de L baies rectangulaires, chaque baie illustre les sections transversales de la barge et elle est numérotée de la proue à la poupe par l'indice $j = 1, \dots, L$. Les lignes parcourent la longueur de la barge et elles sont numérotées du côté bâbord vers le côté tribord par $i = 1, \dots, l$. Les niveaux sont les couches de conteneurs, numérotées de bas en haut et indicés par $h = 1, \dots, Niv$ (**Figure III-3**).

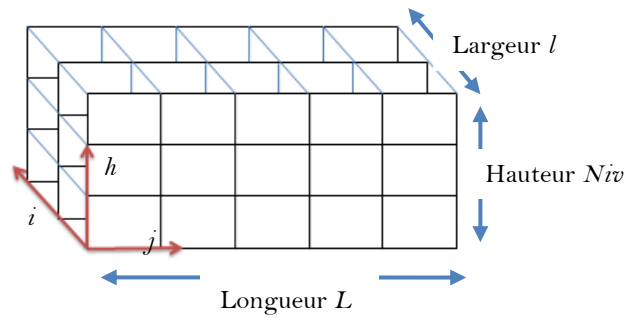


Figure III-3. Structure cellulaire générale d'une barge à conteneurs

La **Figure III-4** montre la disposition de chaque baie de la barge qui se compose d'un certain nombre de piles divisées en de petits emplacements nommés slots ou positions. Le slot dont la ligne i , la baie j et le niveau h est étiqueté (i, j, h) . Un slot peut contenir un seul conteneur de 20 pieds, alors qu'un conteneur de 40 pieds occupe des emplacements de deux conteneurs de 20 pieds. Toutes les piles ont une limite de hauteur commune mesurée en conteneurs de taille 20 pieds ; aucun conteneur ne peut être empilé au-dessus de la hauteur maximale.

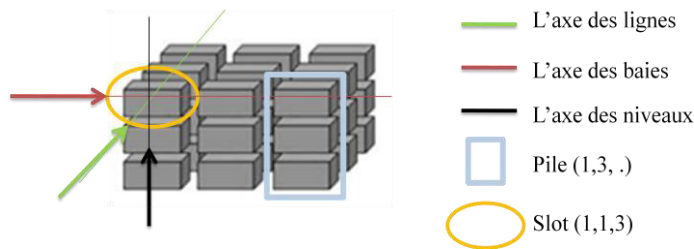


Figure III-4. Illustration des slots de la barge

Nous rappelons que la capacité de la barge est mesurée en unités équivalentes à vingt pieds (EVP, en anglais, TEU), un conteneur standard de 40 pieds égale à 2 EVP. La **Figure III-5** représente un plan d'ensemble pour une barge de 208 EVP ($lxLxNiv=4x13x4$) indiquant le nombre de slots disponibles et leur emplacement selon les vues longitudinale et d'en haut.

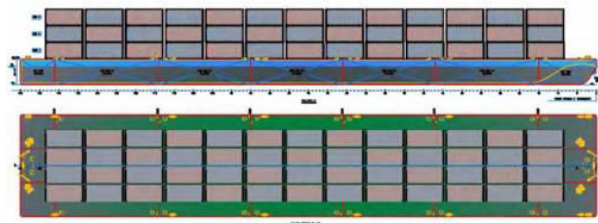


Figure III-5. Plan d'ensemble des conteneurs sur une barge de 208 EVP

3.2 La politique du « dernier entré, premier sorti » (LIFO)

Comme mentionné précédemment, la structure donnée de la barge impose que les conteneurs ne peuvent être atteints qu'en déchargeant tout conteneur empilé au-dessus, et cela, conformément à la politique du « dernier entré, premier sorti » (*Last In, First Out*, (LIFO)). Afin d'accéder à un certain conteneur, la grue est positionnée sur la baie concernée dans laquelle se trouve le conteneur cible. La grue est déplacée vers la pile dans laquelle se trouve ce dernier et son épandeur est abaissé pour atteindre le conteneur. Enfin, le conteneur

est branché et ramassé pour être retiré de son emplacement actuel (**Figure III-6**). Dans cet objectif, les mouvements réalisables par la grue de quai sur la barge sont définis sur la base de la structure intrinsèque LIFO des piles (Expósito-Izquierdo et al., 2017).

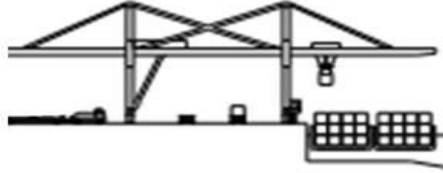


Figure III-6. La politique LIFO appliquée en haut de la barge

Il existe, donc, deux cas de déchargement (Azevedo et al., 2018 ; Araújo et al., 2016), le premier lorsque les conteneurs à décharger sur un port donné se trouvent en haut de la pile et le second lorsque les conteneurs à décharger sur un port donné sont bloqués par un ou plusieurs conteneurs devant rester à bord de la barge. Ces derniers doivent être déchargés, puis rechargés une fois que tous les conteneurs de la pile pour ce port ont été déchargés. Ainsi, en essayant de trouver un plan de rangement approprié pour les conteneurs, les planificateurs doivent prendre en compte la séquence des ports à visiter. Par exemple, si les ports d'escale sont P1, P2 et P3 : pour le déchargement du port P1, le planificateur doit tenir compte de ne pas choisir une position de rangement pour un conteneur ayant comme destination le port P2 ou P3 au-dessus du conteneur destiné au port P1. Sinon, ce mouvement improductif de déchargement et de rechargement diminue la performance des ports, car plus un tel blocage se produit, plus le navire doit rester dans le port, ce qui consomme plus de temps et d'argent (Ding et Chou, 2015).

Comme le montre la **Figure III-7**, la solution du 3D-BSPP devrait produire un plan d'arrimage étroitement lié à la structure cellulaire-rectangulaire de l'espace de chargement qui est discrétisé en EVP selon les trois dimensions (au sens de la longueur, de la poutre et de la profondeur de la barge).

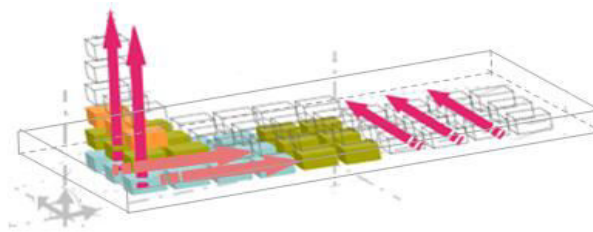


Figure III-7. Illustration du plan d'arrimage du conteneur sur la barge

3.3 Les frais non-uniformes des opérations de manipulation des conteneurs

La barge visite généralement une série de ports en fonction de son programme d'itinéraire fixe et, à chaque port, les conteneurs peuvent être chargés, déchargés ou repositionnés. Dans le présent travail, nous supposons que la barge commence sa tournée au port 0, dans lequel, le plan de chargement est généré, et visite par la suite tous les autres

ports d'escale successivement où seules les opérations de déchargement sont autorisées (**Figure III-8**).

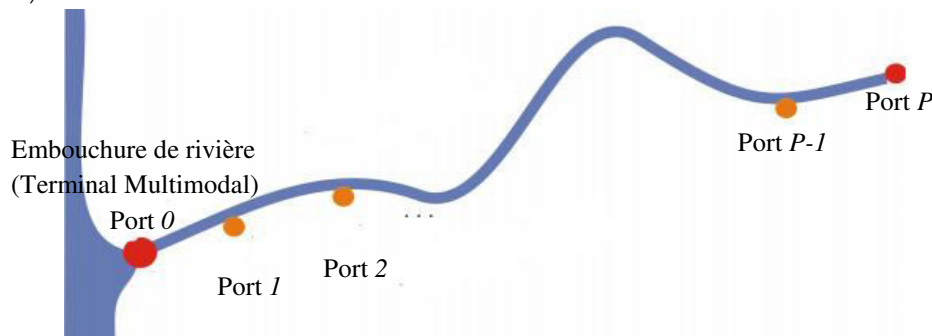


Figure III-8. Un exemple de transport le long d'une rivière avec P ports à visiter. (Source : adaptée à partir de (Alfandari et al., 2019))

Les charges de manipulation et de déplacement des conteneurs sont un facteur important qu'il faut prendre en compte pour améliorer les performances du port. La plupart des travaux antérieurs dans la littérature visent à minimiser le nombre total des mouvements *shiftings* dans chaque port et donc à minimiser le temps qu'un navire passe dans le port, autrement dit, ces travaux supposent que chaque mouvement de conteneur consomme un coût identique dans tous les ports (Chapitre I, section 4.2). Cependant, comme nous l'avons déjà mentionné, nous nous intéressons dans ce chapitre, à un cas inspiré par des problèmes réels, où les mouvements de conteneurs dans différents ports entraînent des coûts différents (Zhang et al., 2018). Par exemple, le coût de mouvement des conteneurs au port de Ningbo-Zhoushan en Chine peut coûter environ 49,5 RMB par conteneur, alors qu'il est de 100 RMB au port de Guangzhou en Chine (Zhang et al., 2018), et dans d'autres ports, il entraîne environ 200 USD (Dubrovsky et al., 2002). Chaque déplacement de conteneur via une grue de quai induit une unité de frais de déplacement qui dépend de la politique du port.

Supposons qu'une barge envisage de visiter $D (\geq 2)$ ports, elle quitte le premier port (port 0) et visite séquentiellement les ports $1, 2, \dots, P$. Initialement, il n'y a aucun conteneur dans la barge avant le chargement au premier port de départ (port 0). Notez qu'il n'y a pas de frais de *shiftings* de conteneurs au port 0 car il n'y a que des opérations de chargement. De même, au dernier port P , tous les conteneurs restants sur la barge doivent être déchargés et il n'y a pas de frais de *shiftings* non plus. Cependant, à chaque port $i = 1, \dots, (P-1)$, tous les conteneurs destinés au port i sont déchargés et un coût significatif des opérations de *shifting* peut être considéré.

Ce chapitre aborde le problème de la planification de l'arrimage des barges (3D-BSPP) dans le cas où les mouvements des conteneurs dans différents ports entraînent des coûts non-uniformes. Ainsi, l'optimisation des frais totaux de *shiftings* est considérée parmi les principaux objectifs traités dans ce travail. Il est également important de noter que d'autres critères doivent être observés, tels que la stabilité de la barge, le poids limite de chaque pile, la taille des conteneurs, etc. (Ambrosino et al., 2006 ; Avriel et al., 1998 ; Wilson et Roach, 2000).

3.4 Notion de stabilité de la barge

La position du conteneur dans une pile donnée affecte fortement la stabilité de la barge. Cette notion de stabilité de la barge indique sa tendance à rester dans l'état d'équilibre ou sa capacité à retrouver sa position d'équilibre initial lorsqu'elle est agitée par l'action des vagues, du vent, etc. Comme mentionné dans le Chapitre I, dans la littérature, il existe deux manières différentes de traiter la stabilité du navire dans le modèle mathématique : sous forme de fonction objectif (Fan et al., 2010 ; Imai et al., 2006), ou sous forme de contraintes (Ambrosino et al., 2006 ; Delgado et al., 2012 ; Dubrovsky et al., 2002 ; Monaco et al., 2014 ; Pacino et al., 2012). Certains chercheurs ont imposé la stabilité en tant que contraintes, et cela lorsque le problème de chargement et d'arrimage des conteneurs est traité en deux phases (*Master Planning Problem*, (MPP) et *Slot Planning Problem*, (SPP)), mais n'expliquaient pas, par exemple, comment la satisfaction des contraintes imposées lors de la planification des baies (MPP) dans la première phase serait assurée après l'application de la planification des emplacements (SPP) dans la deuxième phase, et vice versa (Delgado et al., 2012). Parfois, pour de grandes instances, même pour le problème de planification des baies (MPP), il est difficile de considérer la stabilité comme une contrainte (Ambrosino et al., 2010). Sinon, la prise en compte des mesures de stabilité à l'aide d'une fonction objectif, telle que considérée dans ce chapitre, a conduit Azevedo et al. (2018) et Arajo et al. (2016) à le désigner comme étant le problème de planification de chargement du porte-conteneurs en 3D (en anglais, *3D Container ship Loading Plan Problem*, (CLPP)). Comme l'a montré Tavares de Azevedo et al. (2014), la priorité accordée à l'instabilité du navire dans l'objectif à optimiser pourrait avoir un impact considérable sur le plan de chargement des conteneurs en augmentant le nombre de mouvements inutiles lors de leur déchargement. Cependant, cette limitation pratique est indispensable lors de l'exécution d'un plan d'arrimage.

L'une des notions d'instabilité des barges est la mauvaise répartition du poids des conteneurs dans la barge qui pourrait réduire sa durée de vie et causer des incidences de désarrimage. En règle générale, tout navire porte-conteneurs devient instable si la répartition du poids dans le navire est déséquilibrée. Notons que, contrairement aux porte-conteneurs océaniques, les plus petits navires comme les barges sont beaucoup plus sensibles à la stabilité et aux contraintes structurelles d'arrimage. Par exemple, les opérations de chargement/déchargement d'un grand nombre de conteneurs vers/depuis une seule baie, positionnés vers l'un des coins de la barge, provoquent un déséquilibre qui pourrait potentiellement avoir des effets négatifs sur la stabilité totale de la barge. Parmi ces effets, mentionnons les coûts énormes encourus lorsque les exploitants de la barge tentent de la stabiliser en utilisant des machines et des équipements disponibles sur la barge elle-même, tels que les citernes de ballastage, qui permettent de répartir les problèmes de stabilité. Cependant, ce processus est déconseillé, car cela augmenterait le tonnage de la barge, augmentant ainsi la consommation de carburant.

En règles générales, les mesures de stabilité sont liées aux caractéristiques du navire, telles que l'assiette et la distance métacentrique (Imai et al., 2006), les limites de hauteur et de poids dans une pile (Delgado et al., 2012 ; Pacino et al., 2012), et l'équilibre vertical (Cruz-Reyes et al., 2013). Une revue récente a fourni une classification des articles précédents (Monaco et al., 2012). Nous renvoyons également le lecteur à (Pacino, 2012) et (Derrett,

1999) pour une description plus détaillée des exigences de stabilité et des règles d'empilement.

Dans ce chapitre, nous étudions la stabilité de la barge en se basant sur les mesures de l'assiette et de la gîte (**Figure III-9**). Ces deux mesures sont considérées en connaissant la répartition du poids des conteneurs dans la barge et la position du centre de gravité lorsque la barge n'a pas encore de cargaison (poids léger). Dans les sous-sections suivantes, une description détaillée de ces facteurs d'évaluation liés à la stabilité de la barge est donnée. Par la suite, nous introduisons, en détails, la notion de fragilité des conteneurs qui jouera un rôle important dans la répartition verticale équilibrée du poids des conteneurs dans la barge.

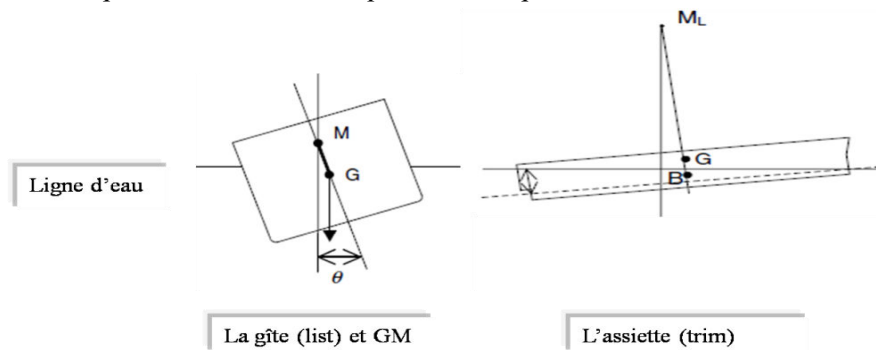


Figure III-9. Les facteurs de stabilité

3.4.1 Stabilité transversale : la gîte

Dans une simple illustration dans la **Figure III-10**, nous considérons une balance avec plusieurs poids suspendus de ses deux côtés, chaque poids est loin du centre de la balance par une distance « d ». Afin de déterminer quel côté est le plus lourd et de faire pencher la balance, il faut calculer le moment⁹ total des deux côtés qui est donné par la somme du produit de chaque poids ajouté/supprimé par sa distance par rapport à l'axe de la médiane. Même si les poids sont différents, l'équilibre est assuré si la somme des moments par rapport à l'axe de rotation est nulle.

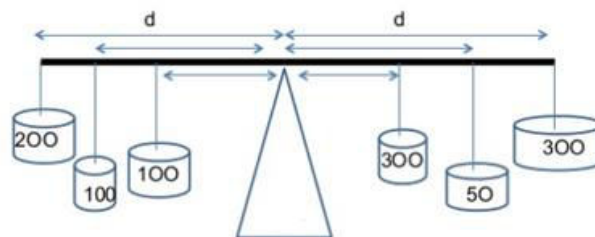


Figure III-10. Illustration de la stabilité d'une balance.

Source : <https://www.slideshare.net/pcorsano/active-management-of-vessel-stability>

Semblable à ce concept, nous définissons la gîte de la barge (**Figure III-11**). La gîte est l'angle qui se produit lorsque la barge se penche vers l'un de ses deux côtés, bâbord ou tribord, à cause de l'effet d'une force interne telle que les ballasts, la distribution de la

⁹ L'aptitude d'une force à faire tourner un système mécanique autour d'un point donné

cargaison et les opérations de *shiftings* (en regardant vers l'avant de la barge, le bâbord et le tribord se réfèrent, respectivement, aux côtés gauche et droit).

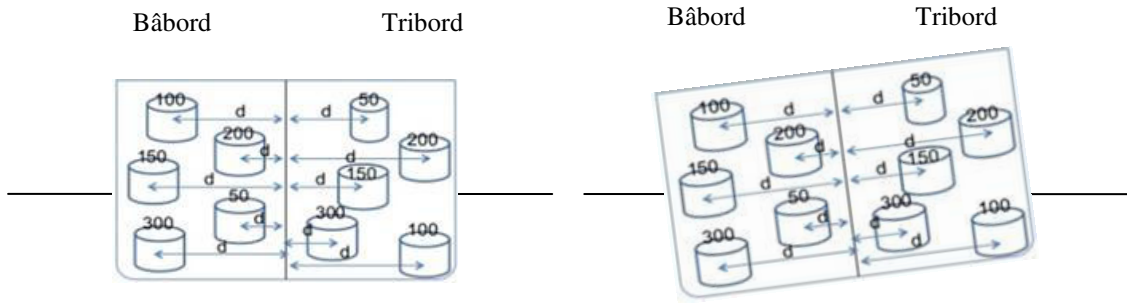


Figure III-11. Stabilité transversale d'une barge. Source : <https://www.slideshare.net/pcorsano/active-management-of-vessel-stability>

Nous avons adapté la formulation proposée par Imai et al. (2002, 2006), pour calculer l'angle qui se produit lorsque la barge est listée de chaque côté.

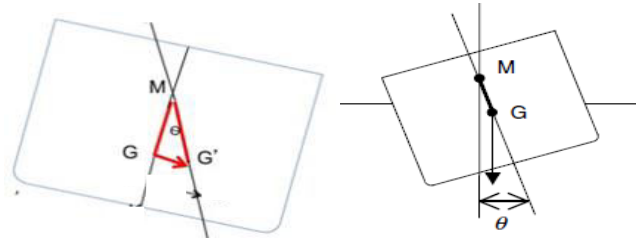


Figure III-12. La gîte et la distance métacentrique de la barge

La gîte, telle qu'illustrée dans la **Figure III-12**, causée par le chargement des conteneurs dans la barge, est mesurée par $\tan \theta$ qui est calculée à l'aide de l'équation (III-1) :

$$\tan \theta = \frac{GG'}{GM} \quad (\text{III-1})$$

Où θ est l'angle de la gîte.

Pour calculer GG' , il faut calculer le moment total de la gîte qui est exprimé par l'équation (III-2):

$$\text{Moment total de la gîte} = |\text{moment tribord} - \text{moment bâbord}| \quad (\text{III-2})$$

Où le moment de chaque côté tribord et bâbord est défini par la somme du produit du poids de chaque conteneur par la distance horizontale correspondante (par rapport au centre vertical).

$$GG' = \frac{\text{moment total}}{\text{poids total}} \quad (\text{III-3})$$

L'angle de la gîte de la barge s'exprime alors par l'équation (III-4), obtenue en insérant l'équation (III-3) dans (III-1).

$$\tan \theta = \frac{\sum_n w_n * l w_n}{\Delta_T * GM} \quad (\text{III-4})$$

Où w_n est le poids du conteneur n , $l w_n$ est la distance horizontale entre le centre vertical de la barge et le centre de gravité du conteneur n , Δ_T est le poids total qui signifie le déplacement de la barge après le chargement des conteneurs, calculé selon l'équation (III-5) et GM est la distance entre le centre de gravité de la barge (G) et le métacentre (M) (plus de détails sont donnés dans le Chapitre IV).

$$\Delta_T = \Delta_S + \sum_n w_n \quad (\text{III-5})$$

Où Δ_S est le déplacement de la barge sans cargaison.

3.4.2 Stabilité longitudinale : l'assiette

En stabilité longitudinale, l'assiette peut être considérée comme l'équivalent de la gîte en stabilité transversale. Donc d'une manière semblable à la définition de la stabilité transversale, nous définissons la stabilité longitudinale comme étant l'inclinaison longitudinale de la barge qui désigne l'état d'équilibre de la barge par rapport au plan horizontal, autrement dit, elle désigne la différence entre le tirant d'eau¹⁰ de l'arrière et le tirant d'eau de l'avant de la barge, comme indiquée dans la **Figure III-13**. Pendant le chargement/déchargement d'une barge, les poids ajoutés/retirés de la poupe et de la proue de la barge permettent de l'incliner des deux côtés (la proue est la partie avant de la barge, et l'autre extrémité de la barge est la poupe).

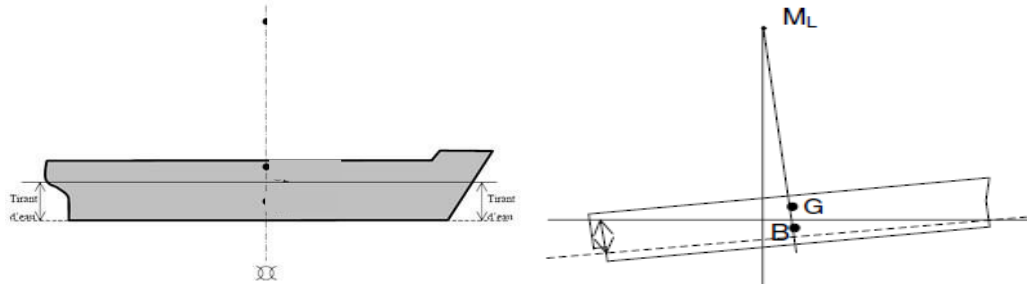


Figure III-13. L'assiette longitudinale de la barge

L'assiette est donnée, donc, par l'équation (III-6):

$$t = \frac{L \sum_n w_n * l l_n}{\Delta_T * GM_L} \quad (\text{III-6})$$

Où L est la longueur de la barge, GM_L est la distance entre le centre de gravité (G) et le métacentre longitudinal (M_L) et $l l_n$ est la longueur horizontale entre le centre de gravité du conteneur n et le centre de flottaison B de la barge qui est le point d'application des forces de flottaison, c'est-à-dire, le centre géométrique du volume immergé (volume du fluide déplacé). Selon la **Figure III-13**, le métacentre se situe très haut lorsque l'assiette se produit; par conséquent, pratiquement, nous supposons que $GM_L = BM_L$, où BM_L est défini par l'équation

¹⁰ La hauteur de la partie immergée du bateau qui correspond à la distance verticale entre la flottaison et le point le plus bas de la coque, généralement la quille.

(III-7) et représente la distance entre le centre de flottaison (B) et M_L avec W est la largeur de la barge.

$$BM_L = \frac{W * L^3}{12\Delta_T} \quad (III-7)$$

Etant donné que la barge est une boîte rectangulaire, l'assiette est exprimée par l'équation (III-8), obtenue en insérant l'équation (III-7) dans (III-6).

$$t = \frac{\sum_n 12w_n * l_n}{W * L^2} \quad (III-8)$$

Comme le plan d'arrimage correspond à la position de chaque conteneur à bord de la barge, seul $w_n * l_n$ et $w_n * l_n$ dépendent de l'emplacement du conteneur à bord. Cela nous permet de ne minimiser que $w_n * l_n$ pour la fonction objectif de la gîte et que $w_n * l_n$ pour la fonction objectif de l'assiette. Nous définissons ces valeurs en se basant sur la description des coordonnées de la structure physique de la barge et des conteneurs comme suit.

3.5 Coordonnées des positions de la barge

Pour la définition du 3D-BCSPP, nous considérons les conteneurs secs ayant des dimensions extérieures conformes aux normes ISO de 20 et 40 pieds de longueur, la barge est donc discrétisée par des conteneurs de 20 pieds, c.-à-d., chaque position dans la barge a les mêmes dimensions qu'un conteneur de 20 pieds. Pour déterminer les coordonnées de chaque position dans la barge, nous considérons l'unité « pieds » comme la plus petite unité de discrétisation. Comme illustré dans la **Figure III-14**, la hauteur standard des conteneurs de 20' et de 40' est de 8'6" et leur largeur standard est de 8'0". Pour simplifier le calcul des coordonnées en unité « pieds », nous supposons que la hauteur et la largeur d'un conteneur (20' et 40') sont toutes les deux égales à 8'0".

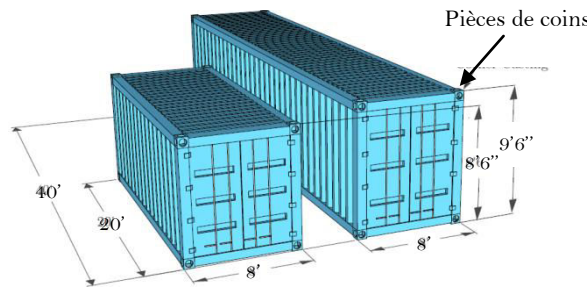


Figure III-14. Dimensions des conteneurs les plus courants (adaptée par Delgado (2013) à partir de (Pacino, 2012))

3.5.1 Les positions dans la barge

Dans la **Figure III-15**, Niv 1 représente le fond de la barge et Niv 3 représente le sommet de la barge. La pile 1 représente la proue de la barge et la pile 3 représente la poupe. Ainsi, la position (1, 1, 1) représente le coin bas-gauche de la première ligne de la barge.

Nous notons par la suite (a_i, b_j, c_h) les coordonnées de la position (i, j, h) , $\forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}$ et $\forall h \in \{1, \dots, Niv\}$.

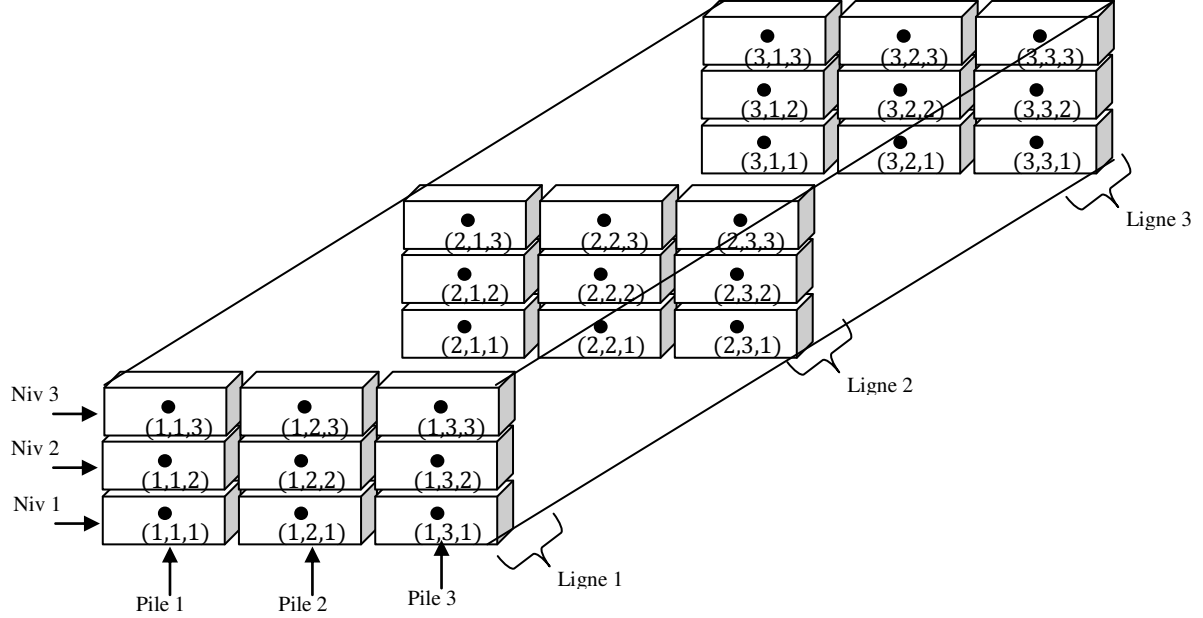


Figure III-15. Les positions dans la barge selon la largeur, la longueur et la hauteur

3.5.2 Le centre de gravité de chaque position dans la barge

Nous supposons que chaque conteneur a le même centre de gravité, c'est-à-dire que le poids est imposé au centre géométrique du conteneur le long des trois axes, même si le centre de gravité d'un conteneur dépend de son poids total et de sa distribution de masse. Selon Imai et al. (2006), l'hypothèse adaptée est considérée valide car la plupart des conteneurs sont remplis dans de petits emballages contenant des marchandises et ont leur centre de gravité global situé au milieu.

Nous indiquons par $C_{(i,j,h)}$ le centre géométrique de la position (i, j, h) , $\forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}$, et G_c le centre de gravité du conteneur c (**Figure III-16**). Notons que dans le cas où la position (i, j, h) est vide, le terme « centre géométrique C » de cette position est confondu avec le terme « centre de gravité G ».

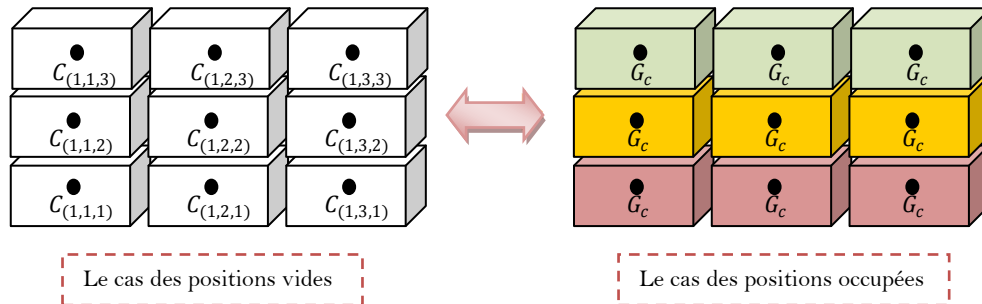


Figure III-16. Le centre de gravité et le centre géométrique des positions

3.5.3 Les coordonnées du centre de gravité de la barge

Nous supposons que le point R représente le point de repère de coordonnées (0,0,0) situé au coin bas-gauche du niveau 0 de la barge, comme illustré dans la **Figure III-17**, et que le centre de gravité de la barge G se trouve dans le centre géométrique du niveau 0 de la barge dont la position est $(i, j, h) = (\frac{l}{2}, \frac{L}{2}, 0)$ qui correspond aux coordonnées $(a_G, b_G, c_G) = (\frac{8l}{2}, \frac{20L}{2}, 4) = (4l, 10L, 4)$, avec l le nombre maximal d'EVP dans la barge selon l'axe i et L le nombre maximal d'EVP dans la barge selon l'axe j , comme illustré dans la **Figure III-18**.

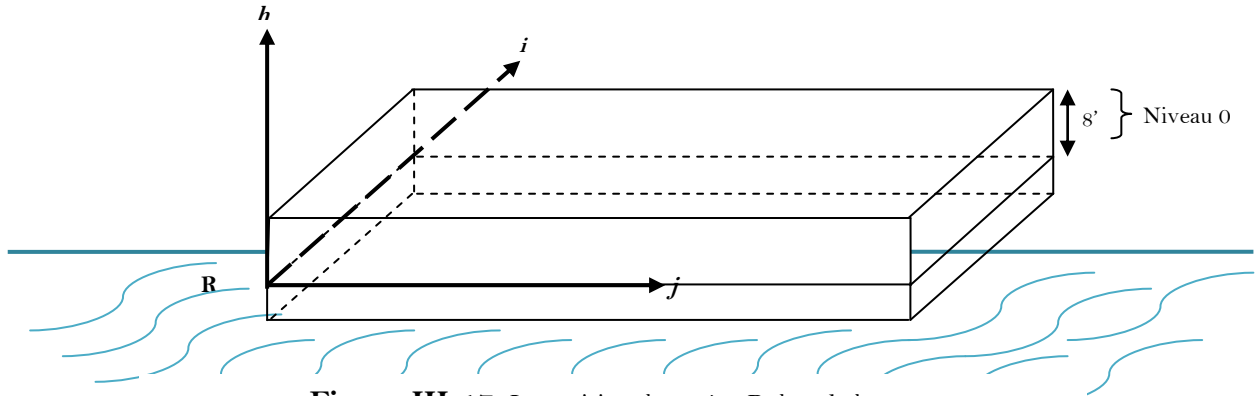


Figure III-17. La position du repère R dans la barge

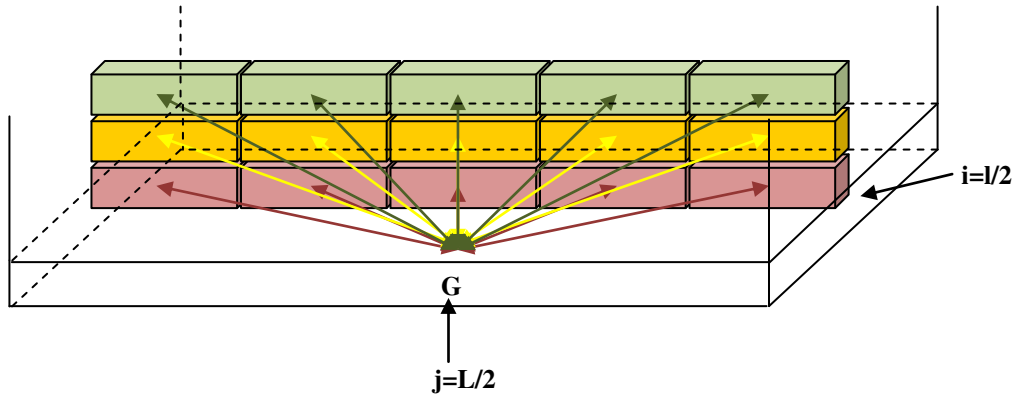


Figure III-18. Les coordonnées du centre de gravité G de la barge

Prenons un exemple d'une barge avec une largeur $l=3$ EVP, une longueur $L=5$ EVP, et une hauteur Niv=3 EVP (**Figure III-19**), les coordonnées de G sont $(\frac{24}{2}, \frac{100}{2}, 4) = (12, 50, 4)$.

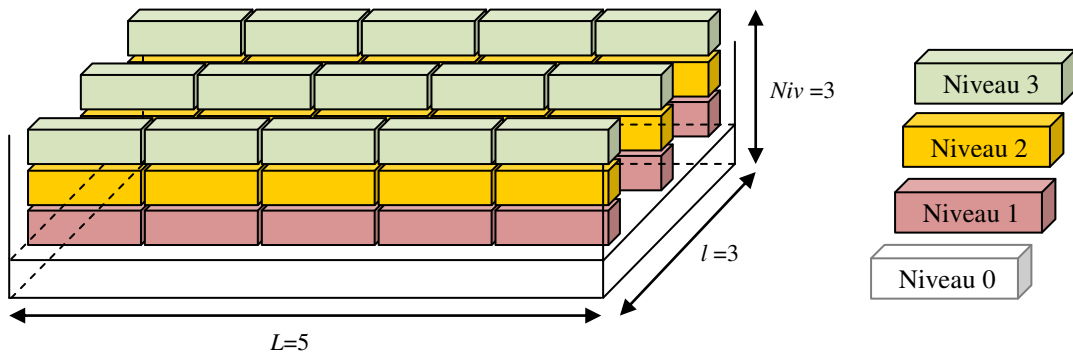


Figure III-19. Exemple d'une barge de capacité 45 EVP

3.5.4 Les coordonnées de chaque position dans la barge selon l'axe i (avec j et h fixes)

Selon la **Figure III-20**, les coordonnées de chaque position $(i, 1, 1), \forall i \in \{1, \dots, 3\}$ sont calculées d'une façon explicite, comme suit :

$$\begin{aligned}(a_1, b_1, c_1) &= (0, 0, 0) + (4, 10, 8 + 4) = (4, 10, 12) \\ (a_2, b_1, c_1) &= (0, 0, 0) + (4, 10, 12) + (8, 0, 0) = (12, 10, 12) \\ (a_3, b_1, c_1) &= (0, 0, 0) + (4, 10, 12) + (8, 0, 0) + (8, 0, 0) = (20, 10, 12)\end{aligned}$$

3.5.5 Les coordonnées de chaque position dans la barge selon l'axe j (avec i et h fixes)

De même, les coordonnées de chaque position $(1, j, 1), \forall j \in \{1, \dots, 3\}$ sont calculées comme suit :

$$\begin{aligned}(a_1, b_1, c_1) &= (0, 0, 0) + (4, 10, 8 + 4) = (4, 10, 12) \\ (a_1, b_2, c_1) &= (0, 0, 0) + (4, 10, 12) + (0, 20, 0) = (4, 30, 12) \\ (a_1, b_3, c_1) &= (0, 0, 0) + (4, 10, 12) + (0, 20, 0) + (0, 20, 0) = (4, 50, 12)\end{aligned}$$

3.5.6 Les coordonnées de chaque position dans la barge selon l'axe h (avec i et j fixes)

De même, les coordonnées de chaque position $(1, 1, h), \forall h \in \{1, \dots, 3\}$ sont calculées comme suit :

$$\begin{aligned}(a_1, b_1, c_1) &= (0, 0, 0) + (4, 10, 8 + 4) = (4, 10, 12) \\ (a_1, b_1, c_2) &= (0, 0, 0) + (4, 10, 12) + (0, 0, 8) = (4, 10, 20) \\ (a_1, b_1, c_3) &= (0, 0, 0) + (4, 10, 12) + (0, 0, 8) + (0, 0, 8) = (4, 10, 28)\end{aligned}$$

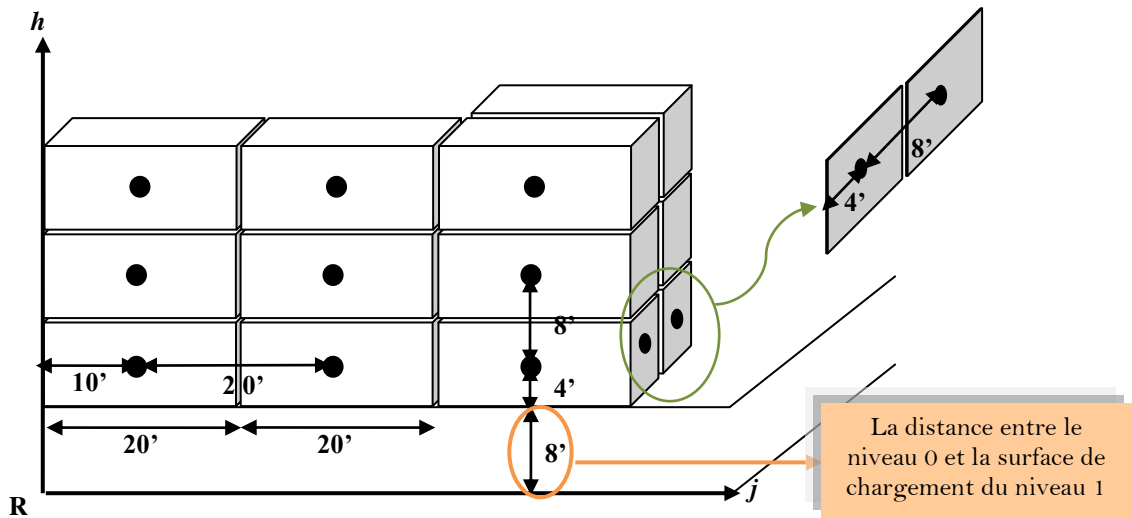


Figure III-20. Les coordonnées des positions dans la barge en 3D

Alors, dans le cas général, les coordonnées (a_i, b_j, c_h) de chaque position (i, j, h) dans la barge $\forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}$ sont calculées selon la formule suivante :

$$\begin{aligned}(a_i, b_j, c_h) &= (4 + 8(i - 1), 10 + 20(j - 1), 12 + 8(h - 1)) \\ (a_i, b_j, c_h) &= (8i - 4, \quad 20j - 10, \quad 8h + 4)\end{aligned}$$

3.5.7 Distance entre le centre de gravité de la barge et la position du conteneur

Pour calculer la distance entre deux points X et Y de l'espace, nous avons utilisé la norme 2 :

$$\begin{aligned}d: \mathbb{R}^3 \times \mathbb{R}^3 &\rightarrow \mathbb{R}^+ \\ (X, Y) &\rightarrow d(X, Y) = \|Y - X\|_2\end{aligned}$$

Telle que :

$$d(X, Y) = d((i, j, h), (i', j', h')) = \|(i, j, h) - (i', j', h')\|_2 = \sqrt{(i - i')^2 + (j - j')^2 + (h - h')^2}$$

Où (i, j, h) et (i', j', h') représentent les coordonnées de X et Y respectivement.

En prenant l'exemple illustré dans la **Figure III-18**, la distance entre G et la position (i, j, h) , $\forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}$ est calculée comme suit :

$$\begin{aligned}d((a_G, b_G, c_G), (a_i, b_j, c_h)) &= \sqrt{(a_G - a_i)^2 + (b_G - b_j)^2 + (c_G - c_h)^2} \\ &= \sqrt{(12 - a_i)^2 + (50 - b_j)^2 + (c_h)^2}\end{aligned}$$

3.6 Notion de fragilité des conteneurs et son influence sur le poids limite de chaque pile

Lors du chargement des conteneurs dans une barge, on doit respecter un ensemble de contraintes, dont les contraintes de poids :

- ▶ Chaque pile a un poids qui ne doit pas être dépassé, sinon cela peut endommager la structure de la barge ou d'autres conteneurs,
- ▶ Les conteneurs les plus lourds doivent être placés dans les niveaux inférieurs et les conteneurs les plus légers dans les niveaux supérieurs (Delgado et al., 2012), sinon, les charges sur les niveaux inférieurs peuvent devenir excessives et les conteneurs peuvent s'effondrer et désarrimer.

Malheureusement, il n'existe pas de définition universelle de ce qui constitue un conteneur lourd ou léger, et cela peut devenir un problème de jugement selon (AIMU, 2018). En conséquence, dans notre étude, nous introduisons une autre manière de représenter ces contraintes de poids, par la notion de fragilité, en considérant le cas où chaque pile possède une fragilité qui ne doit pas être dépassée par le poids total des conteneurs empilés.

Inspirée par le problème de *bin-packing* avec objets fragiles (Bansal et al., 2009), la notion de fragilité de conteneurs peut définir à la fois le poids limite de chaque pile de la barge et assurer l'ordre de chargement convenable des conteneurs selon leur poids. En d'autres termes, nous considérons une certaine tolérance au niveau de chargement des conteneurs selon leur poids. Ainsi, pour une pile donnée, si le poids w_n d'un conteneur n , à charger dans cette pile, dépasse le poids d'un conteneur déjà empilé, la solution est considérée réalisable à condition que w_n ne dépasse pas une certaine limite de résistance que nous appelons « la fragilité de la pile ». En conséquence, la restriction de la fragilité oblige souvent à sélectionner des conteneurs légers dans les niveaux supérieurs, ce qui aide à garantir la stabilité de la barge.

Selon notre connaissance, cette étude est la première à inclure les contraintes de taille des conteneurs avec le niveau de tolérance de poids de chaque pile qui n'est pas connu à l'avance, mais, dépend du paramètre de fragilité.

3.6.1 Calculer la fragilité de la pile

3.6.1.1 Fragilité d'un conteneur

Nous définissons la notion de fragilité des conteneurs en se basant sur la définition de la fragilité dans le problème de Bin-Packing avec Objets Fragiles (*Bin-Packing Problem with Fragile Objects*, (BPP-FO)) (Bansal et al., 2009). La fragilité d'un conteneur standard n désigne combien de poids (en tonnes métriques, (MT)) il peut supporter sur ses pièces de coins.

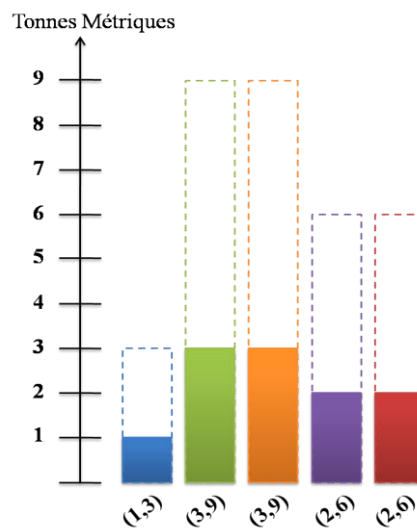


Figure III-21. La notion de fragilité d'un conteneur

La **Figure III-21** montre un ensemble de cinq conteneurs illustrés en une dimension. Chaque conteneur n est représenté par un couple (w_n, f_n) où w_n indique le poids du conteneur n et f_n sa fragilité. La fragilité d'un conteneur est représentée par un rectangle à bordure pointillé et son poids par un rectangle coloré.

3.6.1.2 Fragilité d'une pile

Nous supposons que la limite de poids de chaque pile est connue à l'avance et qu'elle est supérieure à la fragilité maximale de tous les conteneurs. Cependant, lors du processus de chargement de la barge, cette limite sera affectée par la fragilité de chaque conteneur chargé. Ainsi, nous introduisons la notion de limite de poids dynamique de chaque pile. À la lumière de ceci, nous définissons la notion de fragilité de la pile par le poids autorisé qui peut être ajouté dans la pile, en prenant en considération les conteneurs déjà empilés.

Plus formellement, soit N un ensemble de conteneurs à charger dans une seule barge constituée de P piles et soit $N(p_t)$ l'ensemble de conteneurs empilés dans la pile p_t , $\forall t \in \{1, \dots, P\}$ et $Top(n)$ l'ensemble de conteneurs empilés au dessus du conteneur n . Chaque conteneur $n \in N$ est caractérisé par son poids w_n et sa fragilité f_n . Nous supposons que $w_n \leq f_n$ ($\forall n \in N$). Notons par F_{p_t} la fragilité de la pile p_t , $\forall t \in \{1, \dots, P\}$, F_{p_t} peut être calculée selon la formule suivante :

$$F_{p_t} = \min_{n \in N(p_t)} \left(f_n - \sum_{n' \in Top(n)} w_{n'} \right) \quad (\text{III-9})$$

Dans ce contexte, une solution est considérée réalisable, si le poids du dernier conteneur empilé ne dépasse pas la fragilité mise à jour de la pile. Plus formellement, la condition suivante doit être vérifiée pour chaque pile p_t , pour que la solution soit réalisable :

$$F_{p_t} \geq 0 \quad \forall t \in \{1, \dots, P\} \quad (\text{III-10})$$

Notons que si deux piles consécutives p_1 et p_2 sont vides, et qu'un seul conteneur n de taille 40' y doit être empilé, alors la fragilité de chaque pile sera égale à : $F_{p_1} = F_{p_2} = f_n/2$. Dans le cas général, pour calculer la fragilité d'une pile dans laquelle un conteneur 40' est déjà empilé, la fragilité et le poids de ce conteneur doivent être divisés par 2.

La **Figure III-22** montre un exemple simplifié proposé d'une instance de 3D-BCSPP. Chaque conteneur n est illustré sous forme d'un rectangle coloré caractérisé par un triple (w_n, f_n, s_n) indiquant, respectivement, le poids w_n , la fragilité f_n et la taille s_n (où $s_n \in \{T, F\}$, T pour TEU et F pour FEU). Le nombre à l'intérieur de chaque rectangle indique l'identifiant du conteneur. Cette instance est composée de sept conteneurs à charger dans une barge de dimensions $L \times l \times Niv = 3 \times 2 \times 4$.

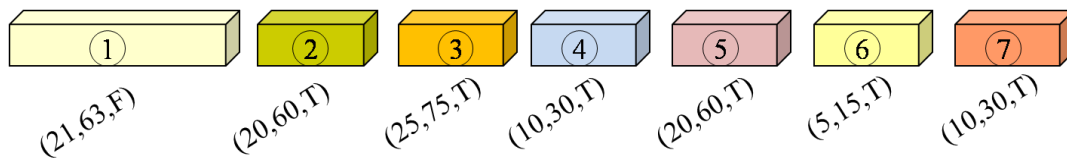


Figure III-22. Une instance d'un ensemble de conteneurs

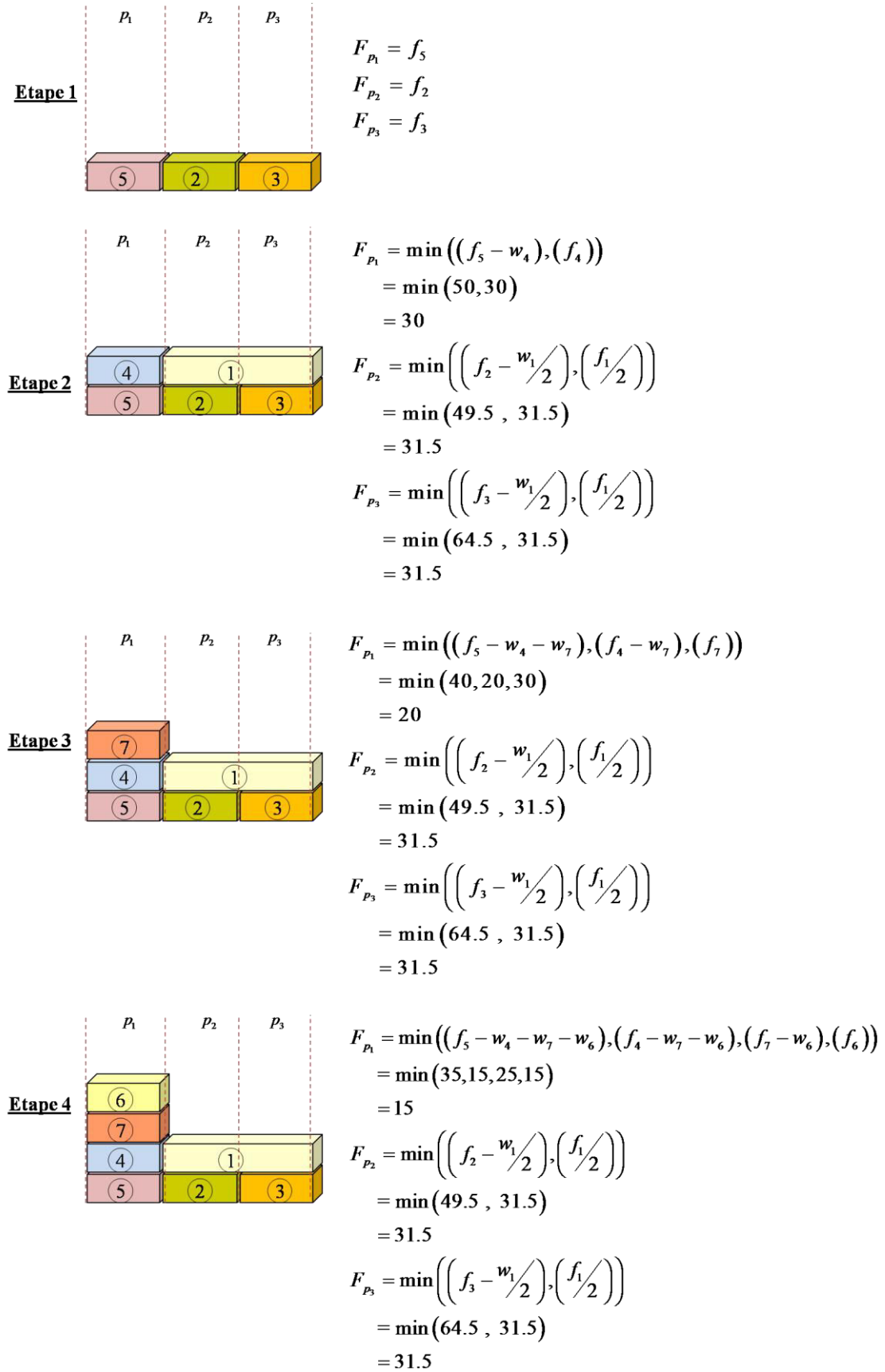


Figure III-23. Une solution réalisable de l'instance de la **Figure III-22** avec calcul explicite de la fragilité de chaque pile

Dans la **Figure III-23**, nous illustrons une solution réalisable de l'instance proposée dans la **Figure III-22**, nécessitant trois piles pour charger les sept conteneurs, la fragilité de chaque pile est calculée à chaque étape de chargement selon la formule (III-9). Nous remarquons qu'à la fin de chargement, nous pouvons ajouter un autre conteneur (s'il existe) dont le poids ne dépasse pas 15 MT dans la pile p_1 , de même, dans les deux piles p_2 et p_3 un conteneur 40' de poids inférieur à 31.5MT peut être ajouté. Cependant, dans la **Figure III-24**, une solution non réalisable est proposée. La fragilité de la pile p_1 après le chargement du conteneur 7 est égale à 20 MT, or le poids du conteneur 3 dépasse cette fragilité, d'où la non faisabilité de la solution.

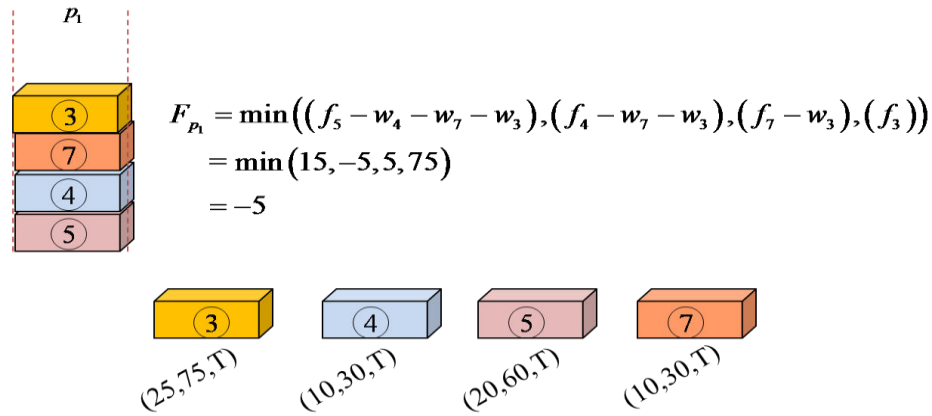


Figure III-24. Une solution non réalisable au sens de fragilité

Par conséquent, dans notre étude, une pile n'a pas de limite de poids maximal fixe et connue auparavant, mais son poids limite sera décidé en fonction de son contenu lors de la construction de la solution.

3.7 L'influence des tailles de conteneurs sur le plan de chargement

Outre les contraintes mentionnées précédemment, beaucoup d'autres contraintes ont un impact majeur sur la planification de l'arrimage, telles que les contraintes physiques du CSPP lorsque différentes tailles de conteneurs sont empilées ensemble. Ces contraintes doivent respecter les règles suivantes:

- ▶ Deux conteneurs ne peuvent pas être placés au même emplacement,
- ▶ Les emplacements assignés doivent former des piles (les conteneurs sont superposés dans les piles, ils ne peuvent pas rester suspendus dans l'air),
- ▶ La hauteur des conteneurs déjà placés dans la pile et la hauteur du nouveau conteneur à empiler ne doivent pas dépasser la limite de hauteur totale de la pile (Parreño et al., 2016). De même pour les deux autres dimensions (largeur et longueur),
- ▶ Un conteneur de 20 pieds ne peut pas être empilé sur un conteneur de 40 pieds, mais l'inverse est possible si la structure du navire le permet. Les planificateurs peuvent également charger un conteneur de 40 pieds au-dessus de deux conteneurs de 20

pieds, ce que l'on appelle un «arrimage russe» (« *russian stowage* ») ou un «arrimage mixte» (« *mixed stowage* ») (Pacino et Jensen, 2012).

La troisième et la dernière règle sont imposées à cause de la structure physique du conteneur. Chaque conteneur est équipé en haut et en bas de pièces de coins (**Figure III-14**). Ces pièces sont construites pour la stabilité de la pile, elles supportent le poids des conteneurs empilés les uns sur les autres, car ils sont conçus pour résister à une grande force (Parreño et al., 2016). Chacune des quatre pièces de coin inférieures d'un conteneur doit reposer sur les pièces de coin d'autres conteneurs ou prises du navire. Lorsque deux conteneurs de 20 pieds sont placés sur un conteneur de 40 pieds, les deux coins intermédiaires ne peuvent pas être utilisés et la pile est donc instable. Cela signifie que les conteneurs de 20 pieds ne peuvent pas être rangés sur des conteneurs de 40 pieds, en raison de leur manque de pièces de coins au centre.

4 Formulation mathématique du 3D-BCSPP

Dans cette section, nous présentons une formulation mathématique non linéaire en nombres entiers pour résoudre le problème de planification en 3D de plans de chargement et d'arrimage des conteneurs dans les barges. D'abord, nous présentons quelques hypothèses et terminologies qui seront utilisées dans la formulation mathématique ultérieurement.

4.1 Hypothèses du problème

Les hypothèses de base utilisées dans ce chapitre sont listées comme suit :

- ▶ La barge est initialement vide et elle a une capacité suffisante pour empiler tous les conteneurs tout au long de son parcours. Le nombre de conteneurs à arrimer est inférieur au nombre de positions disponibles ; cela signifie que ce travail n'est pas concerné par le problème de chargement et d'arrimage de conteneurs dans le cas sélectif,
- ▶ La barge est divisée en un ensemble fini de piles. Chaque pile est définie par la combinaison de chaque ligne avec chaque baie. Elle peut contenir jusqu'à un nombre maximal de conteneurs empilés verticalement,
- ▶ Chaque emplacement (ou position) peut accueillir un conteneur de 20 pieds. Un EVP est considéré comme l'unité de discrétisation des emplacements à l'intérieur de la barge,
- ▶ La barge commence sa tournée à partir du port initial pour lequel nous générons le plan d'arrimage, puis, elle visite successivement un nombre fini de ports connus à l'avance où seules les opérations de déchargement sont autorisées,
- ▶ Les frais de chargement/déchargement des conteneurs sont non-uniformes,
- ▶ Les coordonnées de chaque emplacement dans la barge sont données en « pieds »,
- ▶ Les conteneurs à arrimer sont de deux tailles différentes (40' et 20'),
- ▶ La fragilité de chaque conteneur est supérieure à son poids.

4.2 Modèle mathématique

4.2.1 Données et notations

Les notations suivantes sont utilisées tout au long du chapitre pour modéliser le problème :

4.2.1.1 Les indices

- ▶ n : l'indice des conteneurs
- ▶ i : l'indice des lignes de la barge ($i=1$ représente le côté bâbord de la barge)
- ▶ j : l'indice des baies de la barge ($j=1$ représente la proue de la barge)
- ▶ h : l'indice des niveaux de la barge ($h=1$ représente le niveau le plus bas de la barge)

4.2.1.2 Les paramètres d'entrée

N_1 : l'ensemble des conteneurs de taille 40'

N_2 : l'ensemble des conteneurs de taille 20'

N : l'ensemble de tous les conteneurs $N = N_1 \cup N_2$

L : longueur de la barge discrétisée en 20 pieds

l : largeur de la barge discrétisée en 8 pieds

Niv : le nombre maximal de niveaux dans la barge discrétisée en 8 pieds

ll_j : la longueur horizontale entre le centre de flotaison de la barge et la baie j

lw_i : la distance horizontale entre le centre vertical de la barge et la ligne i

w_n : le poids du conteneur n donné en tonne métrique

$f_{max} = \max_{j \in N} \{f_j\}$ la fragilité maximale parmi celles des conteneurs de l'instance considérée

f_n : la fragilité du conteneur n donnée en tonne métrique

d_n : le port de destination du conteneur n

fee_{d_n} : le coût de shifting du port de destination du conteneur n

$des_{n,n'} = \begin{cases} 1 & \text{si } d_n \text{ est strictement avant } d_{n'}, \\ 0 & \text{sinon} \end{cases}$

$Des = (des_{n,n'})_{1 \leq n, n' \leq |N|}$ la matrice qui définit la relation de succession entre n et n'

$y_n = \begin{cases} 1 & \text{si le conteneur } n \text{ est de taille } N_1 \\ 0 & \text{si le conteneur } n \text{ est de taille } N_2 \end{cases}$

4.2.2 Variables de décision

Avec ces notations et paramètres, une formulation mathématique non linéaire sous forme de programme en nombres entiers du 3D-BCSPP est construite en introduisant les trois variables suivantes :

$$V_{ijhn} = \begin{cases} 1 & \text{si le conteneur } n \text{ est chargé dans la position } (i, j, h) \text{ et } n \in N_2 \\ 0 & \text{sinon} \end{cases}$$

$$U_{ijhn} = \begin{cases} 1 & \text{si le conteneur } n \text{ est chargé dans la position } (i, j, h) \text{ et } n \in N_1 \\ 0 & \text{sinon} \end{cases}$$

sh_{ijhn} = le nombre de shiftings nécessaires pour décharger le conteneur n de la position (i, j, h) ,
 $sh_{ijhn} \in \mathbb{N}$

Nous notons que : $V : I \times L \times Niv \times N \rightarrow \{0,1\}$ et $U : I \times L \times Niv \times N \rightarrow \{0,1\}$
 $(i, j, h, n) \rightarrow V_{ijhn}$ et $(i, j, h, n) \rightarrow U_{ijhn}$

Où :

$$\sum_{i=1}^I \sum_{j=1}^L \sum_{h=1}^{Niv} V_{ijhn} = 0, \quad \forall n \in N_1 \quad (\text{III-11})$$

$$\sum_{i=1}^I \sum_{j=1}^L \sum_{h=1}^{Niv} U_{ijhn} = 0, \quad \forall n \in N_2 \quad (\text{III-12})$$

4.2.3 Fonctions objectifs

La première fonction objectif (f_1) du 3D-BCSPP consiste à réduire les frais des opérations *shiftings* durant toute la rotation de la barge en fonction du plan de chargement obtenu, elle peut être formulé selon l'équation (III-13) :

$$f_1 = \text{Min} \sum_{i=1}^I \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} sh_{ijhn} fee_{d_n} \quad (\text{III-13})$$

Tandis que la deuxième et la troisième fonctions objectifs (f_2 et f_3) sont liées à l'aspect de la stabilité de la barge qui est définie, respectivement, par le facteur de la gîte (stabilité transversale) et celui de l'assiette (stabilité longitudinale). Nous notons que la valeur de ces deux facteurs peut être négative ou positive, c'est la raison pour laquelle nous introduisons la valeur absolue dans la formulation mathématique de f_2 et f_3 .

$$f_2 = \text{Min} \left| \sum_{i=1}^{I/2} \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} w_n lw_i \left((1-y_n) V_{ijhn} + \frac{1}{2} y_n U_{ijhn} \right) - \sum_{i=I/2+1}^I \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} w_n lw_i \left((1-y_n) V_{ijhn} + \frac{1}{2} y_n U_{ijhn} \right) \right| \quad (\text{III-14})$$

$$f_3 = \text{Min} \left| \sum_{i=1}^I \sum_{j=1}^{L/2} \sum_{h=1}^{Niv} \sum_{n \in N} w_n ll_j \left((1-y_n) V_{ijhn} + \frac{1}{2} y_n U_{ijhn} \right) - \sum_{i=1}^I \sum_{j=L/2+1}^L \sum_{h=1}^{Niv} \sum_{n \in N} w_n ll_j \left((1-y_n) V_{ijhn} + \frac{1}{2} y_n U_{ijhn} \right) \right| \quad (\text{III-15})$$

La deuxième fonction objectif (III-14) consiste à la minimisation de la valeur absolue de la différence entre le moment tribord et le moment bâbord (**Figure III-25 (a)**). Alors que la troisième fonction objectif (III-15) consiste à la minimisation de la valeur absolue de la différence entre le moment proue et le moment poupe (**Figure III-25 (b)**).

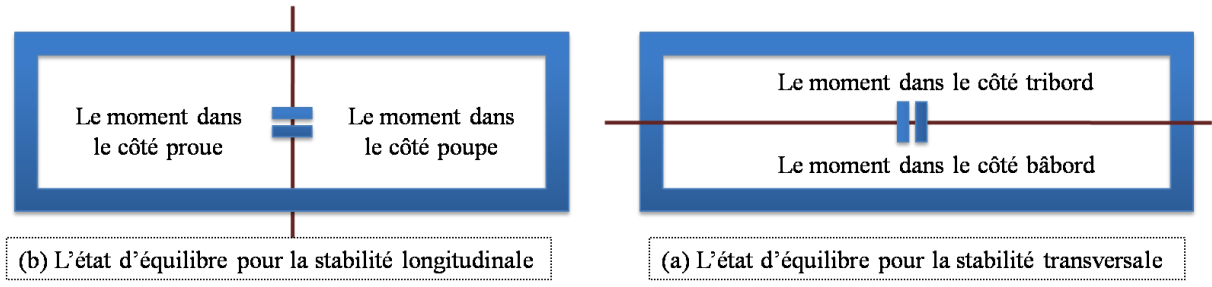


Figure III-25. La stabilité transversale et longitudinale de la barge

4.2.4 Calculer la valeur de ll_j et de lw_i

Compte tenu de la description de la structure physique de chaque emplacement dans la barge expliquée dans la section 3.1, et en se basant sur la définition des facteurs de stabilité longitudinale et transversale de la barge, nous formulons les paramètres ll_j et lw_i liés à la stabilité utilisée dans chaque fonction objectif.

D'abord, nous rappelons que, dans l'état d'équilibre de la barge, le centre de flottaison B se trouve au-dessous du centre de gravité G, les deux points se situent sur le même axe vertical de la barge, nous supposons que B est sous G par 4 pieds. Par ailleurs, les coordonnées de B sont $(a_B, b_B, c_B) = \left(\frac{8l}{2}, \frac{20L}{2}, -4\right) = (4l, 10L, -4)$. Ainsi, nous calculons la longueur horizontale entre le centre de flottaison B de la barge et la baie j , $\forall j \in \{1, \dots, L\}$ selon l'équation (III-16), et la distance horizontale entre le centre vertical de la barge et la ligne i , $\forall i \in \{1, \dots, l\}$ selon l'équation (III-17).

$$ll_j = \sqrt{(10L - 20j + 10)^2} = 10 * |L - 2j + 1| \quad \forall j \in \{1, \dots, L\} \quad (\text{III-16})$$

$$lw_i = \sqrt{(4l - 8i + 4)^2} = 4 * |l - 2i + 1| \quad \forall i \in \{1, \dots, l\} \quad (\text{III-17})$$

Par conséquent, les fonctions objectifs (III-14) et (III-15) peuvent être remplacées par (III-14)' et (III-15)' comme suit :

$$f_2 = \text{Min} \left| \begin{aligned} & \sum_{i=1}^{l/2} \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} 4 w_n |l - 2i + 1| \left((1 - y_n) V_{ijhn} + \frac{1}{2} y_n U_{ijhn} \right) \\ & - \sum_{i=l/2+1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} 4 w_n |l - 2i + 1| \left((1 - y_n) V_{ijhn} + \frac{1}{2} y_n U_{ijhn} \right) \end{aligned} \right| \quad (\text{III-14})'$$

$$f_3 = \text{Min} \left| \begin{aligned} & \sum_{i=1}^l \sum_{j=1}^{L/2} \sum_{h=1}^{Niv} \sum_{n \in N} 10 w_n |L - 2j + 1| \left((1 - y_n) V_{ijhn} + \frac{1}{2} y_n U_{ijhn} \right) \\ & - \sum_{i=1}^l \sum_{j=L/2+1}^L \sum_{h=1}^{Niv} \sum_{n \in N} 10 w_n |L - 2j + 1| \left((1 - y_n) V_{ijhn} + \frac{1}{2} y_n U_{ijhn} \right) \end{aligned} \right| \quad (\text{III-15})'$$

Les fonctions objectifs du modèle mathématique sont optimisées sous les contraintes présentées ci-après.

4.2.5 Contraintes

$$\sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} V_{ijhn} = 1 \quad \forall n \in N_2 \quad (\text{III-18})$$

Les contraintes (III-18) sont les contraintes célèbres d'affectation, elles assurent que tous les conteneurs de taille 20' sont chargés et chaque conteneur est chargé dans un et un seul emplacement (**Figure III-26**).

$$\sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} U_{ijhn} = 2 \quad \forall n \in N_1 \quad (\text{III-19})$$

$$U_{ij+1hn} U_{ij-1hn} \leq U_{ijhn} (1 - U_{ij-1hn}), \forall n \in N_1, \forall i \in \{1, \dots, l\}, \forall j \in \{2, \dots, L-1\}, \forall h \in \{1, \dots, Niv\} \quad (\text{III-20})$$

$$U_{ij+1hn} \geq U_{ijhn} (1 - U_{ij-1hn}), \forall n \in N_1, \forall i \in \{1, \dots, l\}, \forall j \in \{2, \dots, L-1\}, \forall h \in \{1, \dots, Niv\} \quad (\text{III-21})$$

$$U_{ijhn} \leq U_{ij+1hn} \quad \forall n \in N_1, \forall i \in \{1, \dots, l\}, j = 1, \forall h \in \{1, \dots, Niv\} \quad (\text{III-22})$$

$$U_{ijhn} \leq U_{ij-1hn} \quad \forall n \in N_1, \forall i \in \{1, \dots, l\}, j = L, \forall h \in \{1, \dots, Niv\} \quad (\text{III-23})$$

Les contraintes ((III-19) – (III-23)) imposent que tous les conteneurs de taille 40' doivent être chargés et que chaque conteneur est chargé dans deux emplacements consécutifs le long de l'axe j (**Figure III-26**). Ces contraintes sont divisées en deux phases: la première consiste à réserver deux positions (III-19), tandis que la seconde est destinée à préciser la consécutivité selon l'axe j des deux positions réservées ((III-20)- (III-23)).

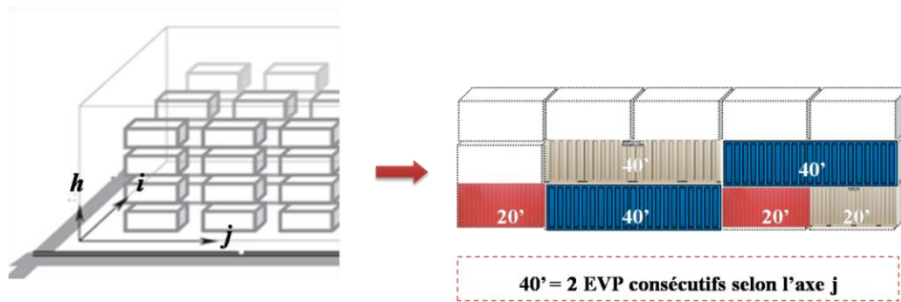


Figure III-26. Contraintes de tailles mixtes de conteneurs (a)

$$(h-1) \left(\sum_{n \in N} (y_n U_{ijhn} + (1-y_n) V_{ijhn}) \right) \leq \sum_{h'=1}^{h-1} \sum_{n' \in N} (y_{n'} U_{ijh'n'} + (1-y_{n'}) V_{ijh'n'}) \quad (\text{III-24})$$

$\forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{2, \dots, Niv\}$

Les contraintes (III-24) assurent que les conteneurs sont empilés les uns sur les autres, c'est-à-dire que si une position (i, j, h) est occupée, alors toutes les positions en dessous doivent également être occupées.

$$1 - \left(y_n U_{ijhn} + (1 - y_n) V_{ijhn} \right) \geq \sum_{\substack{n' \in N \\ n' \neq n}} \left(y_{n'} U_{ijhn'} + (1 - y_{n'}) V_{ijhn'} \right) \quad (III-25)$$

$\forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}$

Les contraintes (III-25) garantissent que si l'emplacement (i, j, h) est occupé, il doit être occupé par un et un seul conteneur.

$$\sum_{h=1}^{Niv} \sum_{n \in N} \left(y_n U_{ijhn} + (1 - y_n) V_{ijhn} \right) \leq Niv \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\} \quad (III-26)$$

Les contraintes (III-26) indiquent que le nombre de conteneurs chargés dans chaque pile ne dépasse pas le nombre maximal de niveaux « Niv » dans la barge. Elles imposent la limitation de hauteur de la barge pour une navigation sûre.

$$\sum_{j=1}^L \sum_{n \in N} \left(y_n U_{ijhn} + (1 - y_n) V_{ijhn} \right) \leq L \quad \forall i \in \{1, \dots, l\}, \forall h \in \{1, \dots, Niv\} \quad (III-27)$$

Les contraintes (III-27) indiquent que le nombre de conteneurs chargés dans chaque ligne ne dépasse pas la longueur « L » de la barge.

$$\sum_{i=1}^l \sum_{n \in N} \left(y_n U_{ijhn} + (1 - y_n) V_{ijhn} \right) \leq l \quad \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\} \quad (III-28)$$

Les contraintes (III-28) indiquent que le nombre de conteneurs chargés dans chaque baie ne dépasse pas la largeur « l » de la barge.

Comme le montre la **Figure III-27**, tous les conteneurs sont équipés de pièces de coins renforcées, et conçues pour résister à une force importante. Ils portent le poids des conteneurs empilés les uns sur les autres. Il convient de noter que les conteneurs de 20' ne peuvent pas être empilés sur les conteneurs 40' en raison de l'absence de pièces de coins au milieu de ce dernier.

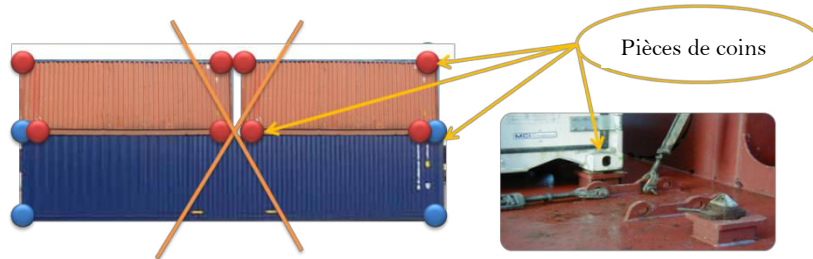


Figure III-27. Les pièces de coins des conteneurs 20' et 40'

$$\sum_{n \in N_2} V_{ijhn} \leq \sum_{n \in N_2} V_{ijh-1n} \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{2, \dots, Niv\} \quad (III-29)$$

Les contraintes (III-29) capturent le fait que deux conteneurs de 20' ne peuvent pas être empilés sur un conteneur de 40'. Mais, l'inverse est possible si la structure de la barge le permet. Les planificateurs peuvent également charger un conteneur de 40' au-dessus de deux unités de conteneur de 20'.

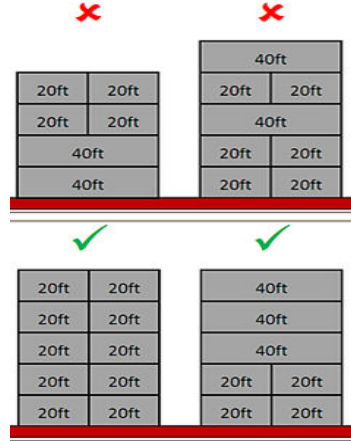


Figure III-28. La contrainte de pièces de coins

La **Figure III-28** montre quelques exemples illustrant la façon dont les conteneurs ne peuvent pas être chargés sans risque d'avoir un désarrimage.

$$\left(1 - \sum_{n \in N_2} V_{ijhn}\right) \left(\sum_{n' \in N_1} U_{ijh+1n'} + \sum_{n \in N_2} V_{ijhn} \right) + \sum_{n \in N_2} V_{ijhn} \left(\sum_{n' \in N_1} U_{ijh+1n'} + \sum_{n \in N_2} V_{ijhn} - 1 \right) \leq \sum_{n' \in N_2} V_{ij+1hn'} \left(\sum_{n \in N_2} V_{ijhn} \right) + \sum_{n'' \in N_1} U_{ijhn''} \left(1 - \sum_{n \in N_2} V_{ijhn} \right) \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L-1\}, \forall h \in \{1, \dots, Niv-1\} \quad (III-30)$$

Les contraintes (III-30) montrent que si l'emplacement (i, j, h) est occupé par un conteneur de taille 20' et que l'emplacement $(i, j, h + 1)$ est occupé par un conteneur de 40' alors l'emplacement $(i, j + 1, h)$ doit être occupé par un conteneur 20' (**Figure III-29**).

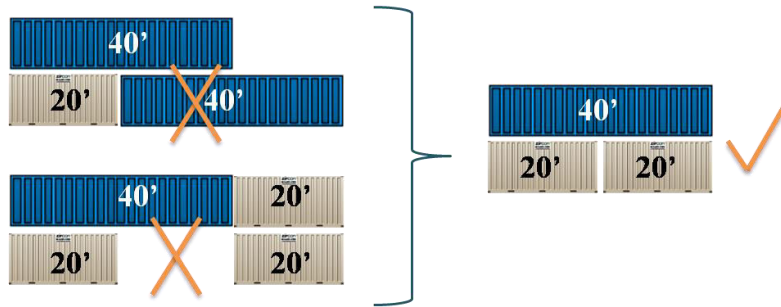


Figure III-29. Contraintes de tailles mixtes de conteneurs (b)

$$\sum_{h'=h+1}^{Niv} \sum_{n' \in N} w_{n'} \left(\frac{1}{2} y_{n'} U_{ijh'n'} + (1 - y_{n'}) V_{ijh'n'} \right) \leq f_{\max} + \frac{1}{2} y_n U_{ijhn} (f_n - 2f_{\max}) + (1 - y_n) V_{ijhn} (f_n - f_{\max}) \quad (\text{III-31})$$

$$\forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv-1\}$$

Les contraintes (III-31) sont les contraintes de fragilité qui assurent que les conteneurs doivent être chargés en fonction de leur poids. La fragilité de chaque conteneur placé dans une pile doit être supérieure ou égale à la somme des poids de tous les conteneurs placés au-dessus.

$$sh_{ijhn} = (1 - y_n) \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Niv} des_{n,n'} (y_{n'} U_{ijh'n'} + (1 - y_{n'}) V_{ijh'n'}) \right) (y_n U_{ijhn} + (1 - y_n) V_{ijhn})$$

$$+ \frac{1}{2} y_n \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Niv} des_{n,n'} (y_{n'} U_{ijh'n'} + (1 - y_{n'}) V_{ijh'n'}) \right) (y_n U_{ijhn} + (1 - y_n) V_{ijhn}) \quad (\text{III-32})$$

$$\forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}$$

Les contraintes (III-32) permettent de calculer le nombre de *shiftings* nécessaire pour décharger un conteneur n dans son port de destination. Le premier terme de ces contraintes indique que si n est un conteneur de 20', alors le nombre de *shiftings* nécessaires pour le décharger de sa position (i, j, h) est égal au nombre de conteneurs dans la pile (i, j) qui sont arrimés au dessus du niveau h du conteneur n et qui n'ont pas encore été déchargés à leurs destinations finales. Cependant, si le conteneur n est de taille 40', alors le nombre de *shiftings* requis pour le décharger de son emplacement (i, j, h) sera divisé par deux, comme indiqué dans le deuxième terme des contraintes (III-32). En effet, comme les conteneurs 40' ne peuvent pas avoir deux 20' au-dessus et comme la barge est discrétisée en EVP, un conteneur 40' occupe deux emplacements consécutifs de 20' le long de l'axe j ((i, j, h) et $(i, j+1, h)$). Ainsi, le *shifting* sera calculé pour l'emplacement (i, j, h) et aussi pour l'emplacement $(i, j+1, h)$, donc, si on ne le divise pas par 2, le nombre total de *shiftings* nécessaires pour décharger n dans son port de destination sera égale à deux fois le nombre réel (Figure III-30).

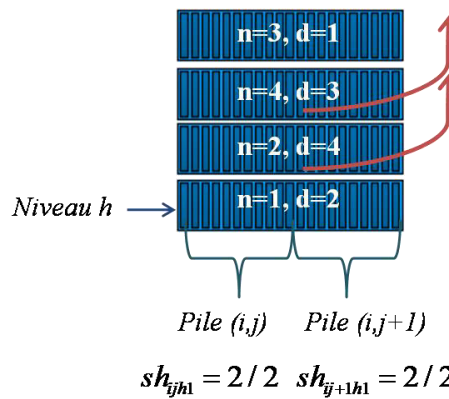


Figure III-30. Le *shifting* pour un conteneur 40'

Après simplification de (III-32), nous obtenons la contrainte (III-32)' :

$$sh_{ijhn} = \left(1 - \frac{1}{2} y_n\right) \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Niv} des_{n,n'} (y_{n'} U_{ijh'n'} + (1 - y_{n'}) V_{ijh'n'}) \right) (y_n U_{ijhn} + (1 - y_n) V_{ijhn}) \quad (III-32)'$$

$$\forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}$$

Les contraintes ((III-33)-(III-35)) spécifient les définitions des variables de décision utilisées dans le modèle.

$$sh_{ijhn} \in \mathbb{N} \quad \forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\} \quad (III-33)$$

$$U_{ijhn} \in \{0, 1\} \quad \forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\} \quad (III-34)$$

$$V_{ijhn} \in \{0, 1\} \quad \forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\} \quad (III-35)$$

Remarque : La fonction objectif (III-13) peut être réécrite en remplaçant sh_{ijhn} par sa valeur indiquée dans les contraintes (III-32)' comme suit :

$$f_1 = \text{Min} \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} fee_{d_n} \left(1 - \frac{1}{2} y_n\right) \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Niv} des_{n,n'} (y_{n'} U_{ijh'n'} + (1 - y_{n'}) V_{ijh'n'}) \right) (y_n U_{ijhn} + (1 - y_n) V_{ijhn}) \quad (III-36)$$

Si nous notons $Z_{ijhn} = y_n U_{ijhn} + (1 - y_n) V_{ijhn}$, alors nous pouvons donner un nouveau modèle mathématique en remplaçant la fonction objectif (III-36) par sa forme linéaire (III-37), et cela en introduisant une troisième variable $W_{ijhnh'n'}$ telle que $W_{ijhn} = Z_{ijhn} * Z_{ijh'n'}$ avec les contraintes (III-38) et (III-39):

$$f_1 = \text{Min} \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} fee_{d_n} \left(1 - \frac{1}{2} y_n\right) \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Niv} des_{n,n'} W_{ijhnh'n'} \right) \quad (III-37)$$

$$W_{ijhnh'n'} \geq Z_{ijhn} + Z_{ijh'n'} - 1 \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}, \forall n \in N, \forall h' \in \{1, \dots, Niv\}, \forall n' \in N \quad (III-38)$$

$$W_{ijhnh'n'} \leq \frac{1}{2} (Z_{ijhn} + Z_{ijh'n'}) \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}, \forall n \in N, \forall h' \in \{1, \dots, Niv\}, \forall n' \in N \quad (III-39)$$

Ce nouveau modèle permet de calculer avec précision le nombre de mouvements *shiftings*, au lieu de faire une estimation comme dans (Imai et al., 2006). De plus, le calcul des paramètres de stabilité de la barge se fait selon une fonction linéaire en utilisant les coordonnées des emplacements des EVP dans la barge au lieu d'une fonction non linéaire comme dans (Tavares de Azevedo et al., 2014).

5 Résolution et discussion des résultats

Le modèle mathématique proposé pour le 3D-BCSPP a été appliqué pour une série d'instances afin d'étudier la façon dont les plans d'arrimage pourraient être influencés par les mesures de stabilité et les mouvements *shiftings*. Des exemples d'applications du problème ont été générés, implémentés en langage OPL (*Optimization Programming Language*) et résolus à l'aide de CPLEX 12.2, un solveur commercial très courant utilisé souvent dans la communauté de la recherche opérationnelle. Or, en raison de ses limitations en ce qui concerne la complexité des problèmes, les expériences ont été réalisées sur des instances de petite / moyenne taille (selon les barges) dans une machine Intel® Core™ i7-47770 CPU @ 3.40GHz, RAM 6 Go. Nous notons que tous les paramètres du solveur CPLEX sont définis sur leurs valeurs par défaut. L'ensemble des instances ont été générées aléatoirement en se basant sur des données réelles, de manière à ce que toutes les conditions de faisabilité de base du problème soient satisfaites. Les résultats numériques sont présentés et discutés dans le but d'évaluer et d'analyser l'efficacité et la performance du modèle proposé.

5.1 L'étude de faisabilité des instances

Dans la plupart des cas, même si la capacité maximale de la barge est suffisante pour accueillir tous les conteneurs, les contraintes réelles du problème imposent un plan de chargement qui n'inclut pas forcément tous les conteneurs à charger. Autrement dit, pour une instance de N conteneurs à transporter avec T (EVP) en total, une barge de capacité maximale égale à T (EVP) peut ne pas être suffisante pour charger tous ces conteneurs. Cela dépend fortement de l'influence des contraintes de fragilité et de tailles mixtes des conteneurs lors du chargement.

5.1.1 Exemple explicatif

Soit N ($N=N_1 \cup N_2$, où $|N_1| = 6$ et $|N_2| = 8$) l'ensemble de conteneurs à charger dans une barge de capacité maximale de 20 EVP. Nous illustrons la barge sous forme de 5 piles avec une hauteur maximale égale à 4 niveaux comme indiqué dans la **Figure III-31**.

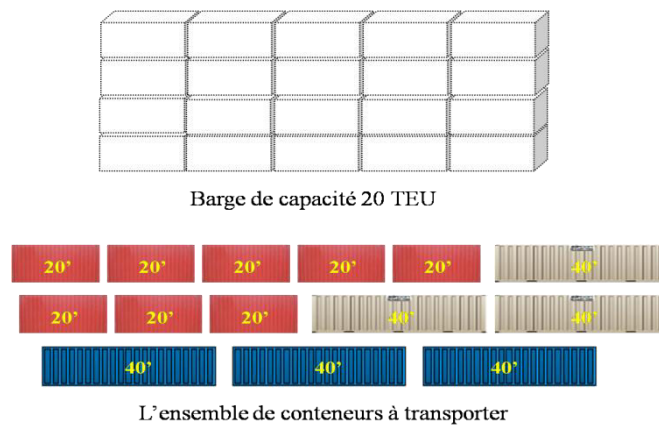


Figure III-31. Instance simplifiée du 3D-BCSPP

Nous supposons qu'à cause des contraintes de fragilité et de taille des conteneurs, deux conteneurs de 20' et un conteneur de 40' ne peuvent pas être chargés dans la barge, comme illustré dans la **Figure III-32**.

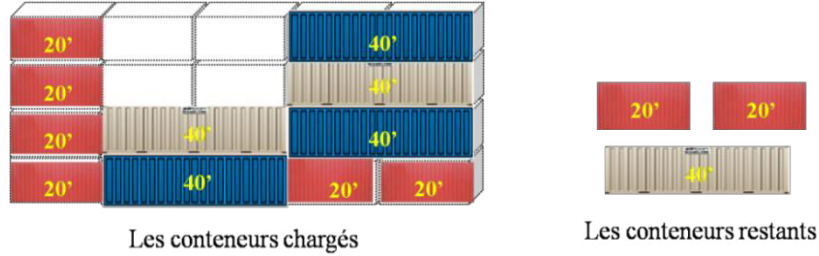


Figure III-32. Problème de faisabilité des instances du 3D-BCSPP

Par conséquent, il n'est pas possible d'arrimer tous l'ensemble N des conteneurs assignés à chaque emplacement d'une barge avec une capacité N EVP. Autrement dit, la condition d'avoir la capacité maximale de la barge égale au un nombre de conteneurs en EVP est une condition nécessaire et non pas suffisante pour transporter l'ensemble de tous les conteneurs, sinon, des décisions concernant la sélection des conteneurs à charger à bord doivent être prises, ce qui dépasse le cadre de notre étude. Par conséquent, seules les données de transport réalisables sont prises en compte tout au long de ce chapitre.

De plus, en étudiant l'ensemble de données utilisées dans des scénarios réels dans (Verdonck et al., 2014), nous pouvons supposer que, dans nos instances de test, chaque barge peut accueillir au maximum $\frac{3}{4}$ (75%) de sa capacité maximale (en EVP) (le quart restant servant comme tampon contre les contraintes de chargement prises en compte dans cette étude).

5.2 Génération des instances du 3D-BCSPP

En se basant sur des données réelles des flux fluviaux du projet HAROPA fournies par le Grand Port Maritime du Havre (GPMH) et sur quelques scénarios trouvés dans (Zhang et al., 2018) et (Hamedi, 2011), nous générons un ensemble d'instances de petite taille avec différentes combinaisons de capacités de barges et d'ensembles de conteneurs à charger. Les listes de tous les paramètres relatifs aux barges, aux conteneurs et aux ports à visiter sont générées d'une façon aléatoire pour chaque instance, de sorte que l'exigence de faisabilité de base du 3D-BCSPP soit satisfaite. Les informations liées aux ports sont données dans le **Tableau III-1**. Nous supposons que la ligne fluviale de la barge contient au maximum 15 ports et que le nombre de ports P d'une tournée varie de 5 à 15 avec l'incrément 5 ($P \in \{5, 10, 15\}$). Les trois sélections de ports à visiter sont indiquées dans la première colonne « *Selection_Ports* » où chaque sélection des P premiers ports du tableau est notée $S_p, \forall P \in \{5, 10, 15\}$ telle que $S_5 \subset S_{10} \subset S_{15}$. L'indice de chaque port dans chaque sélection est représenté dans la deuxième colonne « *Indice_Port* » et les frais de *shiftings* unitaires de chaque port sont listés dans la troisième colonne « *Frais_Sh_Port* ».

Tableau III-1. Paramètres des ports

<i>Selection_Ports</i>			<i>Indice_Port</i>	<i>Frais_Sh_Port</i>
S15	S5	S10	1	30
			2	15
			3	40
			4	50
			5	25
			6	80
			7	60
			8	70
			9	15
			10	30
			11	65
			12	35
			13	70
			14	100
			15	80

En outre, une série de 10 instances, correspondant à des barges de tailles réelles transportant des capacités allant de 16 à 54 EVP, ont été introduites. Les caractéristiques des barges considérées pour générer les instances finales de 3D-BCSPP sont indiquées dans le **Tableau III-2**. Pour chaque barge « *Barge* », le nombre de lignes « *l* », de baies « *L* » et de niveaux « *Niv* » sont indiqués dans les quatre premières colonnes. La cinquième colonne représente la capacité maximale de la barge « *CapM* » qui indique le nombre de positions en EVP disponibles ($CapM = l \times L \times Niv$). Tandis que, la dernière colonne est liée à la capacité réelle de la barge « *CapR* », elle représente le nombre maximal réel de conteneurs en EVP à charger par rapport à la capacité maximale de la barge (fixée à un maximum de 75% de *CapM*). Nous avons pris la partie entière dans le cas où la valeur *CapR* n'est pas un nombre entier. Nous rappelons qu'un conteneur 40' est compté comme deux conteneurs de 20'.

Tableau III-2. Caractéristiques des barges

<i>Barge</i>	<i>l</i>	<i>L</i>	<i>Niv</i>	<i>CapM</i>	<i>CapR (75%)</i>
B1	2	4	2	16	12
B2	2	5	2	20	15
B3	2	5	3	30	22
B4	2	4	4	32	24
B5	3	4	3	36	27
B6	3	3	4	36	27
B7	2	4	5	40	30
B8	2	5	4	40	30
B9	3	5	3	45	33
B10	3	6	3	54	40

Nous générons aléatoirement les caractéristiques des conteneurs à transporter selon les capacités « *CapR* » des barges données dans le **Tableau III-2**. Dans chaque instance, un nombre $|N|$ de conteneurs mixtes doit être transporté ($|N| = |N1| + |N2|$, où $|N1|$ indique le nombre des conteneurs 40' et $|N2|$ le nombre des 20'). Le poids de chaque conteneur a été généré aléatoirement dans le cadre des intervalles de poids. Ces intervalles sont divisés en trois sous intervalles « *L* », « *M* » et « *H* », indiquant respectivement le poids léger, le poids

moyen et le poids lourd, comme détaillé dans le **Tableau III-3**. Nous notons que le poids d'un conteneur 20' se situe généralement entre 3 et 29 tonnes métriques, cependant, le poids d'un conteneur 40' se situe entre 4 et 31 tonnes métriques. Nous estimons que la fragilité de chaque conteneur est égale à 3 fois son poids.

Tableau III-3. Les intervalles de poids des conteneurs

<i>Taille</i>	<i>L</i>	<i>M</i>	<i>H</i>
20'	[3-15]	[16-24]	[25-29]
40'	[4-15]	[16-24]	[25-31]

Les caractéristiques des conteneurs considérés sont rapportées dans le **Tableau III-4**. Pour chaque instance « *InstC* » indiquée dans la cinquième colonne, le nombre total de conteneurs « $|N|$ », le nombre de conteneurs de 20' « $|N2|$ », de 40' « $|N1|$ », et l'intervalle de poids correspondant « W » sont affichés respectivement dans les quatre premières colonnes.

Tableau III-4. Caractéristiques des conteneurs pour chaque instance

$ N $	$ N2 $	$ N1 $	W	$InstC$	$ N $	$ N2 $	$ N1 $	W	$InstC$
7	4	3	L/M	C1	15	7	8	L/M	C28
			M/H	C2				M/H	C29
			L/H	C3				L/H	C30
	6	1	L/M	C4		10	5	L/M	C31
			M/H	C5				M/H	C32
			L/H	C6				L/H	C33
	1	6	L/M	C7		3	12	L/M	C34
			M/H	C8				M/H	C35
			L/H	C9				L/H	C36
10	6	4	L/M	C10	16	1	15	L/M	C37
			M/H	C11				M/H	C38
			L/H	C12				L/H	C39
	2	8	L/M	C13		8	8	L/M	C40
			M/H	C14				M/H	C41
			L/H	C15				L/H	C42
	7	3	L/M	C16		10	6	L/M	C43
			M/H	C17				M/H	C44
			L/H	C18				L/H	C45
14	6	8	L/M	C19	20	5	15	L/M	C46
			M/H	C20	21	10	11	L/M/H	C47
			L/H	C21	23	17	6	L//H	C48
	10	4	L/M	C22	25	14	11	L/M/H	C49
			M/H	C23	27	17	10	L/M/H	C50
			L/H	C24	31	24	7	L/M	C51
	7	7	L/M	C25	32	12	20	L/M/H	C52
			M/H	C26				L/M/H	C53
			L/H	C27					

Finalement, des instances du 3D-BCSPP ont été produites en combinant le **Tableau III-2** et le **Tableau III-4** dans le **Tableau III-5**. Dans chaque instance « *InstBCSP* », une barge « *Barge* » doit être amarrée dans le port initial pour transporter l'ensemble « *InstC* » de conteneurs, afin de visiter la sélection « *Des* » des ports S_P , où $P \in \{5, 10, 15\}$. La destination de chaque conteneur dans chaque instance est générée aléatoirement dans l'intervalle de la

sélection choisie, la notation $rand(S_p)$ représente l'affectation aléatoire des ports appartenant à la sélection S_p aux conteneurs correspondants de chaque instance.

Tableau III-5. Génération des instances du 3D-BCSPP

<i>InstBCSP</i>	<i>Barge</i>	<i>InstC</i>	<i>Des</i>	<i>InstBCSP</i>	<i>Barge</i>	<i>InstC</i>	<i>Des</i>
BCSP1	B1	C1	rand(S5)	BCSP46	B4	C28	rand(S5)
BCSP2		C2		BCSP47		C29	
BCSP3		C3		BCSP48		C30	
BCSP4		C4		BCSP49		C31	
BCSP5		C5		BCSP50		C32	
BCSP6		C6		BCSP51		C33	
BCSP7		C7		BCSP52		C34	
BCSP8		C8		BCSP53		C35	
BCSP9		C9		BCSP54		C36	
BCSP10	B2	C10	rand(S5)	BCSP55	B5	C28	rand(S10)
BCSP11		C11		BCSP56		C29	
BCSP12		C12		BCSP57		C30	
BCSP13		C13		BCSP58		C31	
BCSP14		C14		BCSP59		C32	
BCSP15		C15		BCSP60		C33	
BCSP16		C16		BCSP61		C34	
BCSP17		C17		BCSP62		C35	
BCSP18		C18		BCSP63		C36	
BCSP19	B6	C10	rand(S5)	BCSP64	B6	C37	rand(S5)
BCSP20		C11		BCSP65		C38	
BCSP21		C12		BCSP66		C39	
BCSP22		C13		BCSP67		C40	
BCSP23		C14		BCSP68		C41	
BCSP24		C15		BCSP69		C42	
BCSP25		C16		BCSP70		C43	
BCSP26		C17		BCSP71		C44	
BCSP27		C18		BCSP72		C45	
BCSP28	B3	C19	rand(S5)	BCSP73	B7	C37	rand(S10)
BCSP29		C20		BCSP74		C38	
BCSP30		C21		BCSP75		C39	
BCSP31		C22		BCSP76		C40	
BCSP32		C23		BCSP77		C41	
BCSP33		C24		BCSP78		C42	
BCSP34		C25		BCSP79		C43	
BCSP35		C26		BCSP80		C44	
BCSP36		C27		BCSP81		C45	
BCSP37	B4	C19	rand(S10)	BCSP82	B8	C46	rand(S15)
BCSP38		C20		BCSP83		C46	
BCSP39		C21		BCSP84		C47	
BCSP40		C22		BCSP85		C47	
BCSP41		C23		BCSP86		C48	
BCSP42		C24		BCSP87		C48	
BCSP43		C25		BCSP88		C49	
BCSP44		C26		BCSP89		C50	
BCSP45		C27		BCSP90		C50	
				BCSP91	B9	C51	rand(S15)
				BCSP92		C52	rand(S10)
				BCSP93		C52	rand(S15)
				BCSP94	B10	C53	rand(S5)

5.3 Résultats numériques et discussion

Nous avons modélisé le 3D-BCSPP avec trois objectifs, le premier correspond à la minimisation des frais de *shiftings* de toute la rotation, tandis que le deuxième et le troisième sont liés, respectivement, aux mesures de stabilité transversale et longitudinale de la barge. Cependant, le solveur CPLEX optimise un seul objectif à la fois en utilisant différents paramètres ($\alpha = 1$ pour la première fonction objectif et $\beta = \gamma = 0$ pour les deux objectifs restants et vice-versa).

Tableau III-6. Résultats numériques d'un ensemble d'instances du **Tableau III-5**, en minimisant les frais de *shiftings* où $\alpha = 1$ et $\beta = \gamma = 0$ et leur statut correspondant

<i>InstBCSP</i>	<i>Meilleure val. obj. CPLEX</i>	<i>Obtenue après (s)</i>	<i>Statut CPLEX</i>	<i>Statut temps (s)</i>
BCSP1	0	0.08	Prouvée optimale (*)	0.11
BCSP2	0	0.06	Prouvée optimale (*)	0.07
BCSP3	0	0.05	Prouvée optimale (*)	0.09
BCSP4	0	0.05	Prouvée optimale (*)	0.08
BCSP5	0	0.09	Prouvée optimale (*)	0.09
BCSP6	0	0.06	Prouvée optimale (*)	0.07
BCSP7	0	0.13	Prouvée optimale (*)	0.13
BCSP8	0	0.13	Prouvée optimale (*)	0.15
BCSP9	0	0.05	Prouvée optimale (*)	0.09
BCSP10	0	0.30	Prouvée optimale (*)	0.36
BCSP11	0	0.31	Prouvée optimale (*)	0.41
BCSP12	0	0.13	Prouvée optimale (*)	0.34
BCSP13	0	0.50	Prouvée optimale (*)	0.58
BCSP14	0	0.78	Prouvée optimale (*)	0.78
BCSP15	0	0.33	Prouvée optimale (*)	0.45
BCSP16	0	0.33	Prouvée optimale (*)	0.39
BCSP17	0	0.27	Prouvée optimale (*)	0.39
BCSP18	0	0.18	Prouvée optimale (*)	0.44
BCSP19	0	1.04	Prouvée optimale (*)	1.54
BCSP20	0	1.13	Prouvée optimale (*)	1.53
BCSP21	0	1.40	Prouvée optimale (*)	1.48
BCSP22	0	1.49	Prouvée optimale (*)	2.11
BCSP23	0	1.15	Prouvée optimale (*)	1.75
BCSP24	0	1.68	Prouvée optimale (*)	1.75
BCSP25	0	2.00	Prouvée optimale (*)	2.06
BCSP26	0	5.63	Prouvée optimale (*)	5.63
BCSP28	15	54.13	Prouvée optimale	58.13
BCSP37	0	8.56	Prouvée optimale (*)	19.56
BCSP38	0	20.14	Prouvée optimale (*)	20.17
BCSP39	0	19.97	Prouvée optimale (*)	19.97
BCSP40	25	7.88	Prouvée optimale	15.64
BCSP41	30	195.84	Actuelle	21600.00
BCSP42	0	12.48	Prouvée optimale (*)	12.48
BCSP43	40	819.20	Actuelle	21600.00
BCSP44	0	31.94	Prouvée optimale (*)	31.31
BCSP45	115	6.55	Actuelle	21600.00
BCSP73	0	740.30	Prouvée optimale (*)	780.30
BCSP82	0	1800	Prouvée optimale (*)	1985
BCSP84	0	11348.47	Prouvée optimale (*)	11398.47
BCSP86	340	2206.47	Actuelle	21600.00
BCSP88	30	6735.22	Prouvée optimale	6735.21
BCSP89	790	5086.22	Mém. Insuff.	21100.21
BCSP90	30	13892.50	Mém. Insuff.	20850.80
BCSP91	0	694.08	Prouvée optimale (*)	1002.21
BCSP92	690	881.22	Actuelle	21600.00
BCSP93	15	77369.32	Actuelle	21600.00
BCSP94	---	---	Mem. Insuff.	21600.00

Actuelle : la meilleure valeur objective atteinte à la fin de 6 heures de calcul

Prouvée optimale : la valeur objective de la solution est optimale

Mém. Insuff. : ressources mémoire insuffisantes

(*) : atteindre la borne inférieure

Dans le **Tableau III-6**, les résultats obtenus pour un ensemble d'instances mentionnées dans le **Tableau III-5** sont présentés, tels qu'obtenus par le solveur CPLEX, en minimisant uniquement les frais des opérations *shiftings* (où $\alpha = 1$ et $\beta = \gamma = 0$). Une limite de temps de 6 heures a été imposée au solveur. La première colonne de ce tableau « *InstBCSP* » indique l'identifiant de l'instance du 3D-BCSPP, la deuxième colonne « *Meilleure val. obj. CPLEX* » montre la meilleure valeur objective obtenue par le solveur CPLEX, la troisième colonne « *Obtenue après (s)* » affiche le temps de calcul nécessaire pour trouver la meilleure solution obtenue, la colonne quatre « *Statut CPLEX* » exprime le statut du CPLEX à la fin du calcul. La durée totale d'exécution du solveur nécessaire pour atteindre les statuts correspondants est indiquée dans la dernière colonne « *Statut temps (s)* ». Les valeurs objectives de la deuxième colonne qui ont été prouvées optimales sont en gras.

Evidemment, la valeur de la borne inférieure théorique des frais de *shiftings* est nulle. Selon les résultats obtenus dans le **Tableau III-6**, nous pouvons confirmer que le modèle proposé a permis d'optimiser tous les cas pour les petites instances, plus quelques moyennes instances, sans aucun mouvement *shifting* (indiquées par étoile (*) dans la 4^{ème} colonne), ce qui signifie qu'il a pu atteindre la borne inférieure, alors que dans certains cas, le délai de calcul était atteint sans que le solveur puisse reconnaître la solution actuelle comme étant optimale (indiquée par « *Actuelle* » dans la colonne 4), ou bien il n'a pas réussi à prouver l'optimalité de la solution, en raison des ressources mémoire insuffisantes de la machine (indiquée par « *Mém. Insuff.* » dans la colonne 4).

Comme mentionné auparavant, afin de maximiser la stabilité de la barge, il faut minimiser les mesures de la gîte et de l'assiette. Le **Tableau III-7** affiche les résultats obtenus pour un ensemble d'instances, tels qu'obtenus par le solveur CPLEX, en minimisant, respectivement, le facteur de stabilité transversale (la gîte) avec $\beta = 1$ et $\alpha = \gamma = 0$ et le facteur de stabilité longitudinale (l'assiette) avec $\gamma = 1$ et $\alpha = \beta = 0$. Une limite de temps de 6 heures a été imposée au solveur. Les colonnes de 2 à 5 de ce tableau montrent les résultats obtenus lorsque seul le facteur de stabilité transversale est minimisé, telles que, les valeurs sous l'en-tête « *Meill. val. obj. gîte* » représentent la meilleure valeur objective correspondante obtenue par CPLEX, les valeurs sous l'en-tête « *Obtenue après (s)* » affichent le temps de calcul nécessaire pour obtenir cette solution, les valeurs sous l'en-tête « *Statut CPLEX* » indiquent le statut du solveur à la fin du calcul, tandis que les valeurs sous l'en-tête « *Statut temps (s)* » représentent la durée totale d'exécution nécessaire pour atteindre les statuts correspondants. Suivant la même structure, les colonnes de 6 à 9 représentent les résultats obtenus en minimisant uniquement le facteur de stabilité longitudinale de la barge. Nous notons que les valeurs objectives de la deuxième et la sixième colonne qui ont été prouvées optimales sont indiquées en gras.

Tableau III-7. Résultats numériques d'un ensemble d'instances du **Tableau III-5**, en optimisant la gîte et l'assiette, où « $\beta = 1$ et $\alpha = \gamma = 0$ » et « $\gamma = 1$ et $\alpha = \beta = 0$ » respectivement, et leur statut correspondant

<i>InstBCSP</i>	<i>Stabilité transversale</i>				<i>Stabilité longitudinale</i>			
	<i>Meill. val. obj. gîte</i>	<i>Obtenue après (s)</i>	<i>Statut CPLEX</i>	<i>Statut temps (s)</i>	<i>Meill. val. obj. assiette</i>	<i>Obtenue après (s)</i>	<i>Statut CPLEX</i>	<i>Statut temps (s)</i>
BCSP1	3.99	61.20	P.O.	61.20	9.99	579.34	P.O.	579.34
BCSP2	4	1.47	P.O.	286.40	10	540.34	P.O.	540.34
BCSP3	23.99	123.03	P.O.	144.93	9.99	132.52	P.O.	152.42
BCSP4	8	0.36	P.O.	3.26	0	0.34	P.O. (*)	0.34
BCSP5	0	0.67	P.O. (*)	0.88	0	0.38	P.O. (*)	0.52
BCSP6	16	0.41	P.O.	1.11	9.99	383.51	P.O.	392.12
BCSP7	63.99	0.22	P.O.	0.29	9.99	45.11	P.O.	55.21
BCSP8	31.99	0.83	P.O.	0.89	0	0.29	P.O. (*)	0.34
BCSP9	32.99	1.03	P.O.	1.20	10	13.36	P.O.	17.36
BCSP10	0	38.47	P.O. (*)	48.27	0	1.35	P.O. (*)	1.41
BCSP11	4	0.26	P.O.	0.33	0	1.48	P.O. (*)	1.98
BCSP12	7.99	24.51	P.O.	28.01	0	4.02	P.O. (*)	4.18
BCSP13	4	281.24	Actuelle	21600.00	10	12511.68	Actuelle	21600.00
BCSP14	4	992.47	Actuelle	21600.00	10	2165.47	Actuelle	21600.00
BCSP15	460	8.66	P.O.	226.09	10	1778.94	Actuelle	21600.00
BCSP16	8	12.23	P.O.	370.60	14	1003.42	P.O.	1184.25
BCSP17	3.99	2554.17	Actuelle	21600.00	0	1.09	P.O. (*)	1.09
BCSP18	11.99	1420.90	Actuelle	21600.00	0	0.75	P.O. (*)	0.93
BCSP37	0	2252.58	P.O. (*)	2252.58	10	1426.96	Mém. Insuff.	7033.06
BCSP38	0	101.26	P.O. (*)	101.26	0	115.21	P.O. (*)	115.36
BCSP39	12	329.90	P.O.	1329.63	20	83.74	Actuelle	21600.00
BCSP40	0	7.00	P.O. (*)	7.69	0	36.54	P.O. (*)	45.38
BCSP41	0	2145.21	P.O. (*)	6200.00	10	869.11	Actuelle	21600.00
BCSP42	4	113.36	Mém. Insuff.	1988.31	0	32.43	P.O. (*)	36.64
BCSP43	180	87.72	P.O.	97.56	10	80.67	Mém. Insuff.	973.58
BCSP44	0	932.42	P.O. (*)	1156.42	10	1692.00	Mém. Insuff.	2981.23
BCSP45	8	11656.58	Actuelle	21600.00	110	16169.41	Actuelle	21600.00
BCSP73	0	3591.14	P.O. (*)	3600.00	30	1426.90	Actuelle	21600.00
BCSP74	24	3111.74	Actuelle	21600.00	30	12328.90	Actuelle	21600.00
BCSP75	152	933.87	Actuelle	21600.00	90	555.53	Actuelle	21600.00
BCSP76	0	34.89	P.O. (*)	36.43	0	46.52	P.O. (*)	72.14
BCSP77	8	239.32	P.O.	440.12	0	234.32	P.O. (*)	288.83
BCSP78	16	161.41	P.O.	4563.34	10	148.21	P.O.	148.33
BCSP79	32	4829.21	P.O.	5237.75	50	57.74	Mém. Insuff.	928.23
BCSP80	34	9316.34	Actuelle	21600.00	10	123.32	Mém. Insuff.	9381.98
BCSP81	120	7638.21	Actuelle	21600.00	50	873.23	Mém. Insuff.	6400.00
BCSP82	6	1722.56	Mém. Insuff.	10436.23	17	19743.09	Actuelle	21600.00
BCSP86	38	886.25	P.O.	1952.75	0	783.45	P.O. (*)	1669.03
BCSP91	9	15431.54	Actuelle	21600.00	270	3918.34	Mém. Insuff.	4401.91
BCSP93	240	15924.37	Actuelle	21600.00	570	18714.02	Actuelle	21600.00
BCSP94	720	11343.13	Actuelle	21600.00	990	13311.59	Mém. Insuff.	19063.83

Actuelle : la meilleure valeur objective atteinte à la fin de 6 heures de calcul

P.O. : la valeur objective de la solution est optimale (Prouvée Optimale)

Mém. Insuff. : ressources mémoire insuffisantes

(*) : atteindre la borne inférieure

Les résultats du **Tableau III-7** montrent que le modèle proposé peut être résolu avec une gîte et une assiette très faibles, ce qui signifie une très bonne stabilité transversale et longitudinale, et ce, dans une durée inférieure à 6 heures, et dans certains cas, en quelques secondes. Le solveur a pu prouver l'optimalité de la solution trouvée pour les instances marquées en gras pour les deux fonctions objectifs, de plus, les solutions qui ont atteint la borne inférieure sont indiquées par (*) dans la quatrième et la huitième colonne. Cependant, il existe un certain nombre d'instances pour lesquelles CPLEX n'a pas réussi à trouver l'optimum, mais a réussi à obtenir la meilleure solution réalisable trouvée pendant 6 heures de calcul, ces instances sont indiquées par « Actuelle » dans les colonnes 4 et 8. Nous pouvons constater que, parfois, pour une même instance, le solveur prouve l'optimalité pour l'une des deux fonctions objectifs représentées dans ce tableau sans qu'il puisse la prouver pour l'autre (BCSP15, BCSP17, BCSP18, BCSP21-BCSP25, BCSP37, BCSP39, BCSP41-BCSP44, BCSP73, BCSP79). De plus, nous remarquons que, pour certaines instances, CPLEX n'a pas réussi à prouver l'optimalité de la solution, en raison des ressources mémoire insuffisantes de la machine.

D'après les résultats présentés dans les deux tableaux, (**Tableau III-6** et **Tableau III-7**), nous constatons que le solveur CPLEX prend quelques secondes pour résoudre des problèmes de petite taille et des dizaines de minutes pour résoudre des problèmes de taille moyenne en prouvant l'optimalité. Ces résultats expérimentaux confirment la validité de notre modèle mathématique proposé pour l'ensemble des instances étudiées.

5.4 Comparaison numérique

Dans cette section, nous évaluons la performance du modèle proposé en termes de l'influence de la nature de l'instance sur la qualité de la solution. Nous illustrons quelques résultats numériques des **Tableau III-6** et **Tableau III-7** pour montrer la façon dont le plan de chargement pourrait être affecté en optimisant ou bien les mesures de stabilité, ou bien les frais de *shiftings* dans le cas uniforme et non-uniforme.

5.4.1 La performance des mesures de stabilité

Dans cette section, nous analysons l'influence de la taille des conteneurs sur la stabilité transversale et longitudinale de la barge. Pour ce faire, nous reprenons quelques instances traitées dans le **Tableau III-7**. Nous présentons, dans la **Figure III-33**, l'influence de la répartition entre le nombre de conteneurs de 20' et celui de 40' sur la stabilité transversale et dans la **Figure III-34**, l'influence sur la stabilité longitudinale.

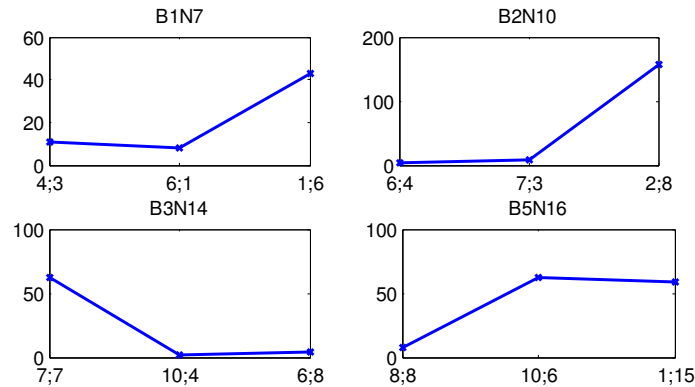


Figure III-33. L'influence de la nature de tailles de conteneurs sur la stabilité transversale de la barge

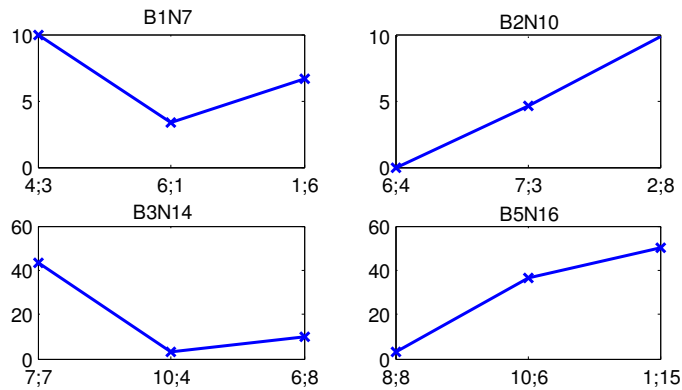


Figure III-34. L'influence de la nature de tailles de conteneurs sur la stabilité longitudinale de la barge

L'axe des abscisses représente le (nombre de conteneurs 20' ; nombre de conteneurs 40') et sur l'axe des ordonnées, nous illustrons la stabilité transversale dans la **Figure III-33** et celle longitudinale dans la **Figure III-34**.

Par exemple, le premier sous-graphe de la **Figure III-33** est généré en considérant l'arrimage de 7 conteneurs dans la barge B1, d'où le sous titre B1N7. L'axe horizontal correspond au nombre de conteneurs des deux tailles 20' et 40' ($|N2|; |N1|$), le premier point (4;3) représente la taille $|N2|=4$ et $|N1|=3$, le deuxième représente $|N2|=6$ et $|N1|=1$, tandis que le troisième correspond à $|N2|=1$ et $|N1|=6$ et l'axe vertical indique la stabilité transversale moyenne des instances « BCSP1, BCSP2, BCSP3 », « BCSP4, BCSP5, BCSP6 » et « BCSP7, BCSP8, BCSP9 » respectivement.

En analysant les deux figures (Figure III-33 et Figure III-34), nous remarquons qu'il n'y a pas de corrélation entre la taille des conteneurs et la stabilité de la barge. En effet, le changement de la taille des conteneurs n'influence pas sur la stabilité de la barge.

Ensuite, nous analysons la **stabilité transversale** de la barge en fonction de la nature de distribution des intervalles de poids (L/M, M/H et L/H), sachant que la capacité de la barge, le nombre de conteneurs $|N|$ et la répartition entre $|N1|$ et $|N2|$ sont fixés. En effet, nous illustrons dans la **Figure III-35**, l'influence de ces intervalles de poids sur la gîte, dans les trois cas L/M, M/H et L/H indiqués dans le **Tableau III-4**.

Par exemple, dans le premier sous-graphe de la **Figure III-35**, nous considérons l'arrimage de $|N|=7$ conteneurs, où $|N_2|=4$ et $|N_1|=3$, dans la barge B1, d'où le sous titre « B1N7N2₄N1₃ ». L'axe horizontal correspond à la nature des intervalles de poids, en considérant trois points : L/M, M/H et L/H et l'axe vertical indique la stabilité transversale des instances correspondantes « BCSP1 », « BCSP2 » et « BCSP3 » respectivement.

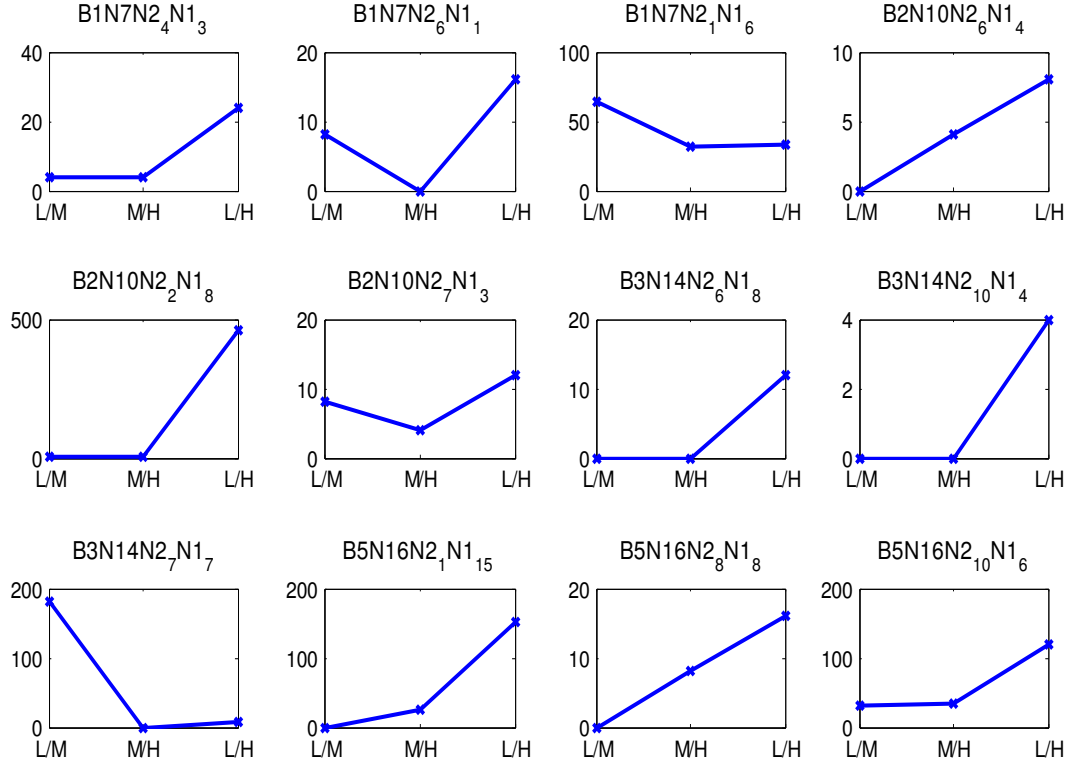


Figure III-35. L'influence de la distribution des intervalles de poids sur la stabilité transversale

Nous pouvons constater que pour la majorité des sous-graphes de la **Figure III-35**, lorsque les poids appartiennent à l'intervalle L/M ou M/H, le facteur de la gîte est petit par rapport à celui de L/H, et parfois il atteint la borne inférieure, ce qui signifie une meilleure stabilité transversale. Cependant, lorsque les poids sont plus dispersés, la gîte prend des valeurs plus grandes par rapport à celles de L/M et M/H, ce qui signifie une stabilité moins bonne. Cela se produit avec tous les sous graphes à part le « B3N14N2₇N1₇ » et le « B1N7N2₁N1₆ », où la stabilité est dans son état parfait lorsque les poids appartiennent aux intervalles M/H et L/H. De plus, les sous graphes « B2N10N2₆N1₄ » et « B5N16N2₈N1₈ » montrent une relation linéaire entre la distribution des poids et la stabilité transversale de la barge.

Suivant la même description de la **Figure III-35**, nous étudions la **stabilité longitudinale** de la barge en fonction de la nature de distribution des intervalles de poids (L/M, M/H et L/H), sachant que la capacité de la barge et le nombre de conteneurs $|N|$, $|N_1|$ et $|N_2|$ sont fixés. Nous illustrons dans la **Figure III-36**, l'influence de ces intervalles de poids sur l'assiette, dans les trois cas L/M, M/H et L/H indiqués dans le **Tableau III-4**.

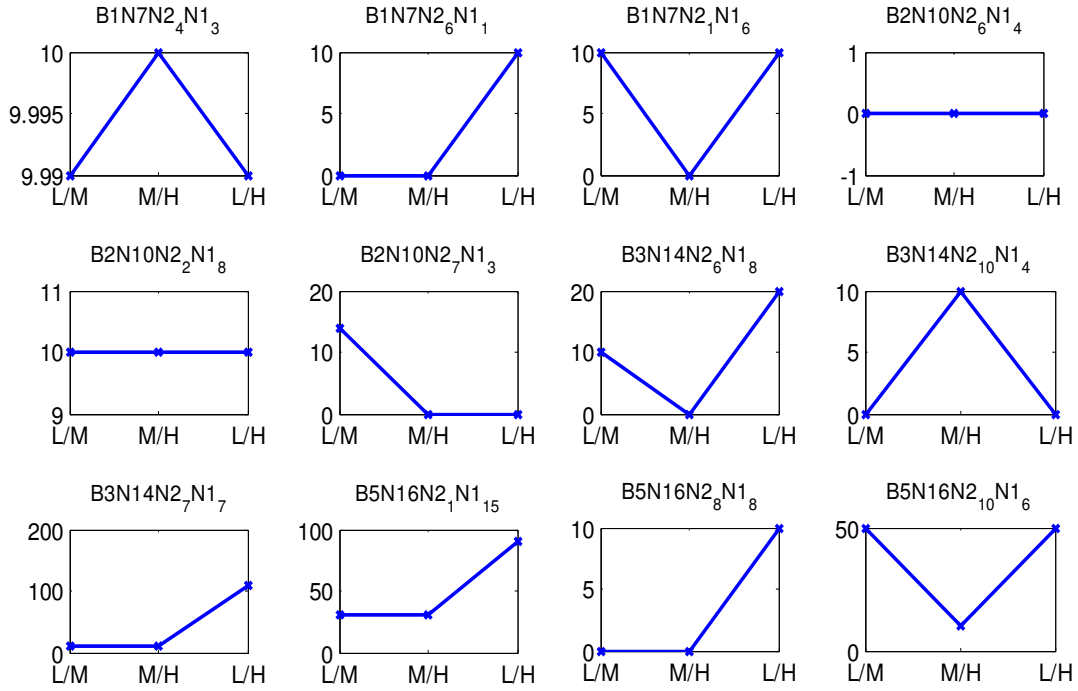


Figure III-36. L'influence de la distribution des intervalles de poids sur la stabilité longitudinale

Selon la **Figure III-36**, nous pouvons affirmer que la stabilité longitudinale (l'assiette) diffère d'une instance à une autre et diffère, aussi, selon la répartition des poids des conteneurs. Les sous-graphes « B1N7N2₆N1₁ », « B1N7N2₁N1₆ », « B2N10N2₇N1₃ », « B3N14N2₆N1₈ », « B3N14N2₁₀N1₄ », « B3N14N2₇N1₇ », « B5N16N2₁N1₁₅ », « B5N16N2₈N1₈ » et « B5N16N2₁₀N1₆ » montrent une influence forte des distributions des poids sur la stabilité longitudinale de la barge, par exemple, dans « B3N14N2₇N1₇ », la stabilité varie entre 10 et 100. Tandis que, « B1N7N2₄N1₃ » représente une influence presque négligeable, où la stabilité varie entre 9.99 et 10. De plus, les deux sous-graphes « B2N10N2₆N1₄ » et « B2N10N2₂N1₈ » ne montrent aucune influence des distributions des poids sur la stabilité longitudinale de la barge. Nous remarquons, aussi, que la répartition des poids des conteneurs n'influence pas, de la même manière, sur la stabilité longitudinale que sur la stabilité transversale illustrée dans la **Figure III-35**.

5.4.2 Comparaison des frais de *shiftings* dans le cas uniforme et non-uniforme

Afin de montrer l'influence des frais de *shiftings* sur la planification de l'arrimage des conteneurs, nous concevons des expériences numériques sur deux cas, le cas où les frais de *shiftings* sont uniformes et le cas où ils sont non-uniformes. Pour ce faire, nous considérons un ensemble d'instances présentées dans le **Tableau III-5**. Dans le cas des frais de *shiftings* uniformes, sans perte de généralité, nous définissons les frais unitaire comme frais moyens sur les 10 premiers ports du **Tableau III-1**, c'est-à-dire que les frais de *shiftings* unitaires dans chaque port sont de 41.5 (unité monétaire). Lorsque le modèle est réglé pour minimiser uniquement la fonction de frais de *shiftings* ($\alpha = 1$ et $\beta = \gamma = 0$), les solutions optimales

trouvées par CPLEX pour les deux cas sont présentées dans le **Tableau III-8**. La première colonne représente les instances étudiées, la deuxième et la troisième colonne affichent les résultats obtenus pour le cas des frais de *shiftings* non-uniformes, tandis que la quatrième et la cinquième colonne indiquent les résultats dans le cas des frais uniformes. L'entête « *Nbre* » et « *Frais* » désignent, respectivement, le nombre total de *shiftings* et les frais de *shiftings*.

Tableau III-8. Comparaison des frais de *shiftings* dans le cas uniforme et non-uniforme

<i>Instance</i>	<i>Frais de shiftings non-uniformes</i>		<i>Frais de shiftings uniformes</i>	
	<i>Nbre</i>	<i>Frais</i>	<i>Nbre</i>	<i>Frais</i>
BCSP1	0	0	0	0
BCSP10	0	0	0	0
BCSP20	0	0	0	0
BCSP28	1	15	1	41.5
BCSP45	4	115	3	124.5
BCSP88	2	30	1	41.5

Dans le **Tableau III-8**, pour les deux cas où les frais de *shiftings* sont uniformes et non-uniformes, le solveur résout de manière optimale les trois premières instances en atteignant la borne inférieure. Pour toutes les trois dernières instances, nous observons que les solutions dans le cas de frais de *shiftings* non-uniformes correspondent à des valeurs objectives plus petites que dans l'autre cas où les frais de *shiftings* sont uniformes. Cela implique que, dans le cas des frais non-uniformes, la plupart des mouvements *shiftings* sont affectés aux ports ayant les tarifs les moins chers, bien que cela puisse entraîner quelques mouvements supplémentaires des conteneurs. Par exemple, pour l'instance « BCSP45 », il y a $(4 - 3)$ *shiftings* de plus, mais $(124.5 - 115 = 9.5)$ unités monétaires) moins de frais dans le cas des frais de *shiftings* non-uniformes. Par conséquent, nous concluons que lorsque les frais de *shiftings* ne sont pas uniformes, pour éviter de générer des frais de déchargement élevés dans les ports où les frais de *shiftings* sont élevés, des mouvements *shiftings* peuvent se produire dans certains ports précédents moins chers. Bien qu'un tel ajustement puisse entraîner un plus grand nombre de *shiftings*, le coût total de *shiftings* des conteneurs peut être réduit.

6 Conclusion

Dans ce chapitre, nous avons proposé un modèle mathématique du problème d'optimisation des plans de chargement et d'arrimage des conteneurs dans une barge dans le cas des frais de *shiftings* non-uniformes capable de gérer des contraintes structurelles et opérationnelles réelles. Compte tenu d'une seule barge avec ses caractéristiques structurelles et son itinéraire décrit par une séquence de ports à visiter, où chaque port a des frais de mouvement de conteneurs différents, le problème consiste à définir le plan d'arrimage pour un ensemble donné de conteneurs de différentes tailles, poids, fragilités et ports de destination, tous ces conteneurs doivent être chargés à bord en optimisant la stabilité de la barge et les frais de *shiftings* non-uniformes, ce qui apporte une contribution originale à la littérature. Une série de tests expérimentaux est ensuite effectuée à l'aide du solveur CPLEX, et les résultats de calcul rapportés vérifient la faisabilité du modèle proposé et

démontrent son efficacité. En effet, le modèle proposé peut souvent trouver un plan d'arrimage sans *shiftings* et un plan avec une stabilité maximale, ce qui signifie qu'il atteint la borne limite inférieure théorique pour toutes les fonctions objectifs considérées. Les futurs plans de recherche portent sur le développement d'un algorithme heuristique dans le cas multi-objectif pour les problèmes de grande taille en prenant en compte l'influence du temps de calcul. Nous proposons dans le chapitre suivant, de présenter le concept multi-objectif pour résoudre le 3D-BCSPP dans le cas d'un système multi-barges.



Chapitre IV

Optimisation multi-objectif du problème de chargement et d'arrimage des conteneurs en 3D dans un système de convoi de barges: modélisation et résolution par le NSGA-II

1 Introduction

Ce chapitre est une continuation des études présentées au chapitre III. Il s'intéresse à un cas inspiré par des problèmes réels, où le plan de chargement et d'arrimage des conteneurs est généré pour un système de convoi constitué d'un bateau poussant plusieurs barges amarrées entre elles. Dans ce cas, le problème est nommé problème d'optimisation des plans de chargement et d'arrimage des conteneurs, en trois dimensions, dans un système de convoi de barges homogènes (en anglais, *3D – barge Convoy Container Stowage Planning Problem*, (3D-CCSPP)).

Les exigences relatives à l'arrimage de tous les conteneurs dans un système de convoi de barges impliquent des contraintes supplémentaires liées à la fois aux conteneurs et au convoi. Cette variante du problème de planification des plans de chargement et d'arrimage des conteneurs (CSPP) est traitée sur la base de sa relation avec le problème de *bin-packing* en trois dimensions (3D-BPP), dans lequel les objets remplis représentent les conteneurs, et les *bins* indiquent les barges du convoi. Dans notre étude, nous considérons l'aspect multi-objectif du problème avec trois objectifs réalistes à minimiser. C'est un aspect nouveau et efficace compte tenu de l'état de l'art des problèmes de chargement des barges. Nous avons développé une nouvelle formulation mathématique multi-objectif du 3D-CCSPP. Bien qu'elle obtienne des résultats optimaux pour presque toutes les instances de petite taille, d'autres résultats de calcul montrent qu'elle n'est pas adaptée à la résolution d'instances réelles de grandes tailles dans un très court temps de calcul tel requis par l'industrie. Le 3D-CCSPP est plus complexe, mais aussi plus réaliste, et pourrait être traité en utilisant une méta-heuristique. Par conséquent, nous avons adapté la méta-heuristique multi-objectif NSGA-II (*Non-dominated Sorting Genetic Algorithm II*), en la combinant avec les heuristiques du problème de *bin-packing*.

À notre connaissance, aucune étude dans la littérature ne considère le cas d'arrimage des conteneurs dans les systèmes de convoi de barges et vise simultanément à optimiser le nombre de *shiftings*, la stabilité totale du convoi et le nombre de barges utilisées dans une formulation à objectifs multiples.

Le reste de ce chapitre est organisé comme suit. La deuxième section met en œuvre le principe de fonctionnement d'un système de convoi de barges. La troisième section décrit formellement le problème de chargement et d'arrimage des conteneurs dans un convoi de barges (3D-CCSPP) et identifie ses différentes caractéristiques, en s'appuyant sur l'aspect multi-objectif du problème et sa relation avec le problème de *bin-packing* en 3D (3D-BPP). Ensuite, un modèle mathématique approprié est présenté dans la quatrième section décrivant ainsi les différentes contraintes structurelles et opérationnelles. La cinquième section présente une série d'exemples pratiques résolus par le solveur CPLEX pour valider le modèle proposé. De plus, l'approche multi-objectif NSGA-II est développée, en la combinant avec deux heuristiques du problème de *bin-packing* pour résoudre le 3D-CCSPP sur des instances de grandes tailles. Nous présentons ensuite un ensemble de résultats numériques que nous évaluons en proposant un ensemble de mesures de performances adaptées au problème. Enfin, nous terminons par une conclusion.

2 Transport fluvial par un système de convoi de barges

Comme déjà mentionné dans le chapitre I, les barges sont des unités de transport fluvial à fond plat, avec lesquelles, différents types de marchandises peuvent être transportés. Cependant, l'aspect le plus important des barges réside dans le fait qu'elles ne sont pas, toutes, généralement des bateaux indépendants. Elles doivent être remorquées par un remorqueur (ou pousseur, « *towboat* » ou « *tugboat* ») pour assurer leur déplacement. Ce dernier est un type de navire propulseur très puissant qui manœuvre d'autres navires en les poussant ou en les tirant, soit par un contact direct, soit au moyen d'un câble solide de remorquage. Il possède généralement de grandes plaques montées sur la proue permettant de s'aligner sur la poupe rectangulaire de la barge. Sa cabine de pilotage peut être montée pour offrir au pilote une bonne vision de navigation et aussi abaissée pour passer sous les ponts.

De ce fait, plusieurs barges peuvent être accouplées et fixées de bout en bout, formant ainsi un système appelé convoi ou convoi poussé, dont le déplacement est assuré par un pousseur sur les rivières et les canaux à grand gabarit. Il n'est pas rare d'associer plusieurs barges, en largeur comme en longueur, afin de constituer un convoi « séparable » (**Figure IV-1**).

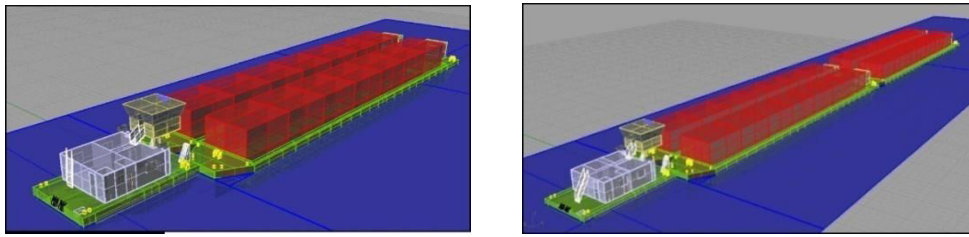


Figure IV-1. Deux types de système de barges poussées dans un convoi. Source : (Van Hassel, 2011)

Il est à noter que les convois fluviaux sont constitués d'un bateau pousseur et un ensemble de barges dont les dimensions peuvent varier. Cependant, la tendance actuelle est de concevoir des unités aux dimensions standardisées afin d'optimiser les capacités de transport lourd qu'offre la voie navigable. De ce fait, nous nous intéressons dans ce chapitre au cas où le convoi est constitué d'un ensemble de barges homogènes.

3 Description du problème

Principalement motivé par la massification des flux sur la partie fluviale, où plusieurs barges peuvent être rassemblées en convois et dont le déplacement est assuré par un pousseur pour transporter un grand nombre de conteneurs vers et depuis l'*hinterland*, le problème d'optimisation des plans de chargement et d'arrimage des conteneurs dans un système de convoi de barges homogènes en 3D (3D-CCSPP) est une extension originale du problème de planification de plans de chargement et d'arrimage des conteneurs en 3D dans une seule barge (3D-BCSPP) traité dans le chapitre III.

Comme bien mentionné auparavant, l'optimisation du nombre de mouvements improductifs « *shiftings* » des conteneurs lors des opérations de déchargement est devenue une préoccupation majeure dans le but d'éviter l'augmentation des coûts et les retards dans les ports. Dans ce chapitre, nous supposons que chaque mouvement de conteneur consomme un coût uniforme dans tous les ports. De ce fait, la minimisation des frais aux ports exige à son tour la minimisation des mouvements improductifs des conteneurs effectués dans chaque port.

En se basant sur le 3D-BCSPP, le 3D-CCSPP peut être formellement défini comme suit : étant donné un convoi fluvial de barges homogènes avec un trajet bien défini à l'avance, décrit par une séquence de ports à visiter ayant un coût de *shifting* uniforme, le problème consiste à trouver le plan de chargement et d'arrimage d'un ensemble de conteneurs dans le convoi tout en respectant un certain nombre de contraintes structurelles, opérationnelles et physiques relatives à la fois aux conteneurs et au convoi ; tous les conteneurs doivent être chargés à bord et chaque conteneur est identifié par sa taille, son poids, sa fragilité et par son port de destination. Le plan est généré dans le but d'optimiser trois objectifs. Le premier correspond aux mouvements improductifs des conteneurs lors des opérations de déchargement à chaque port du trajet, le deuxième représente la notion de stabilité du convoi définie par la distance métacentrique de chaque barge, que nous détaillons par la

suite, tandis que le troisième est lié à l'aspect du problème de *bin-packing* en trois dimensions afin de diminuer le nombre de barges dans le convoi.

L'arrangement d'une dizaine ou d'une centaine de conteneurs ayant une dizaine de destinations dans un système multi-barges soulève de nombreuses possibilités difficiles à résoudre. Nous rappelons, brièvement, les principales règles à respecter lors des opérations de chargement des conteneurs mentionnées dans le chapitre III, et cela est appliqué à toutes les barges du convoi :

- ▶ Eviter les chevauchements des conteneurs dans le sens où deux conteneurs ne peuvent pas être placés dans le même emplacement ;
- ▶ Les emplacements assignés doivent former des piles (les conteneurs sont superposés dans les piles et ne peuvent pas rester suspendus dans l'air) ;
- ▶ Ne pas dépasser la limite de nombre de conteneurs alloué en termes de chaque dimension ;
- ▶ Un conteneur de 40 pieds peut être empilé au-dessus de deux conteneurs de 20 pieds, ce qui est strictement interdit dans le cas inverse ;
- ▶ Ne pas dépasser le poids limite de la pile qui est défini par la fragilité des conteneurs qui y sont chargés.

Comme le montre la **Figure IV-2**, la solution du 3D-CCSPP devrait produire un plan d'arrimage étroitement lié à la structure cellulaire-rectangulaire de l'espace de chargement de chaque barge du convoi qui est discrétisé en EVP selon les 3D.

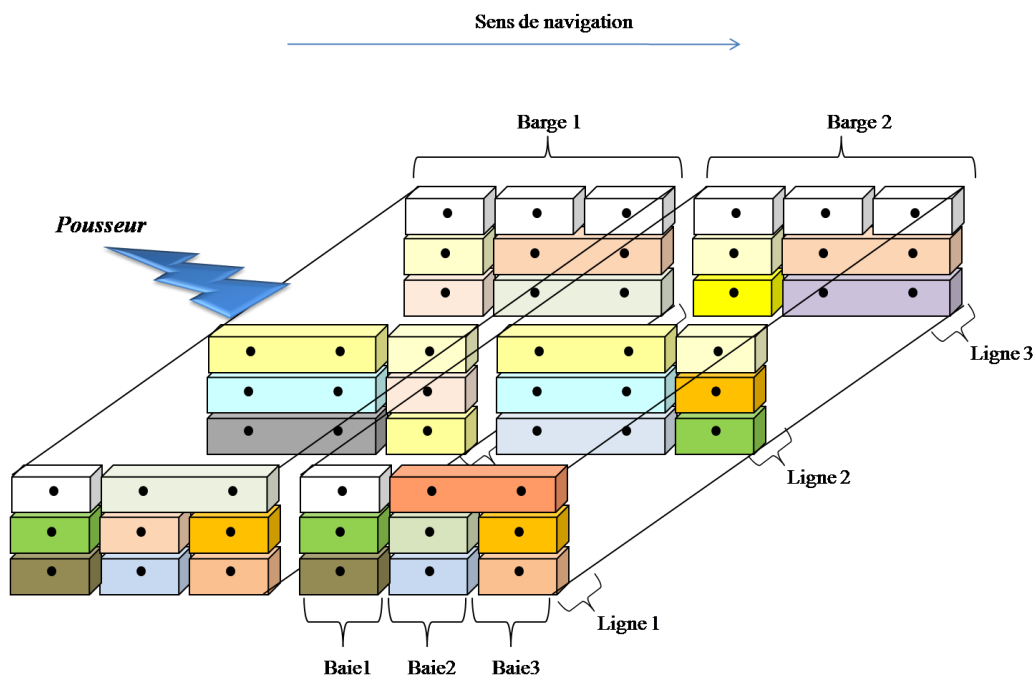


Figure IV-2. Exemple d'un convoi poussé de deux barges homogènes de capacité 27 EVP totalement remplies

3.1 Relation du 3D-CCSPP avec le problème de *bin-packing* en 3D

La structure spéciale du convoi de barges permet de résoudre le 3D-CCSPP en tant que cas particulier des problèmes de découpe et d'emballage (*Cutting and Packing Problem*, (CPP)), en particulier en tant que problème de *bin-packing* en 3D dans le cas orthogonal (3D-OBPP) à orientation fixe (Dyckhoff, 1990 ; Wäscher et al., 2007). Ce dernier est un problème combinatoire qui consiste à placer un ensemble S de n objets de dimensions (w_i, h_i, d_i) , $i = 1, \dots, n$, dans le cas orienté, dans un nombre minimum de *bins* identiques R_j , $j = 1, \dots, m$, de dimensions (W, H, D) où W (resp. w_i) est la largeur, H (resp. h_i) la hauteur et D (resp. d_i) la profondeur des *bins* (resp. des objets) de telle façon que les arêtes des objets soient parallèles à celles des *bins* qui les contiennent sans déborder les *bins* et sans aucun chevauchement. À notre connaissance, il n'existe pas de travaux dans la littérature qui traitent le problème d'arrimage de conteneurs dans un convoi de barges sous forme de problème de *bin-packing*.

3.2 La stabilité de la barge selon la distance métacentrique GM

Nous rappelons que la stabilité de la barge dépend généralement de trois facteurs, à savoir, la gîte, l'assiette et la distance métacentrique GM. Plus de discussions sur ces trois facteurs peuvent être vues dans (Imai et al., 2006), (Derrett et Barrass, 1999), (Azevedo et al., 2018) et (Barrass, 2000). Pour simplifier, nous considérons dans ce chapitre la distance métacentrique comme mesure de stabilité du convoi et nous supposons que les problèmes de stabilité soulevés par les deux autres facteurs peuvent être résolus grâce à l'utilisation de citernes à ballast.

Le facteur GM indique la distance entre le métacentre (M) et le centre de gravité (G) de la barge, ce qui montre, entre autres choses, à quel point il est facile ou difficile pour la barge de chavirer. Plus cette distance est grande, plus il devient difficile pour la barge de se renverser. En général, un GM excessif peut être causé par : une mauvaise conception du navire, une distribution inadéquate de la cargaison ou une distribution inadéquate du bunker¹¹ et du ballast. Dans cette étude, nous mettons l'accent sur le deuxième point.

GM traite principalement la stabilité transversale (**Figure IV-3**), qui concerne l'équilibre vertical et l'équilibre croisé (Imai et al., 2006 ; Zhang et Lee, 2016). Il peut être calculé à l'aide de la formule suivante :

$$GM = G_0M + \frac{\sum_n w_n * m_n}{\Delta_T} \quad (IV-1)$$

Dans (IV-1), G_0M est la distance métacentrique initiale, elle est fixe pour toutes les barges. w_n est le poids du conteneur n . m_n est la distance verticale entre le métacentre (M) et le conteneur n . Δ_T est le poids de la barge avec cargaison, défini par $\Delta_T = \Delta_S + \sum_n w_n$, où Δ_S est le poids de la barge sans cargaison.

¹¹ Bunker : le soutage est l'opération consistant en la prise d'hydrocarbures de soute ou « bunkers » à bord d'un navire. Ces hydrocarbures de soute servent à la propulsion du navire.

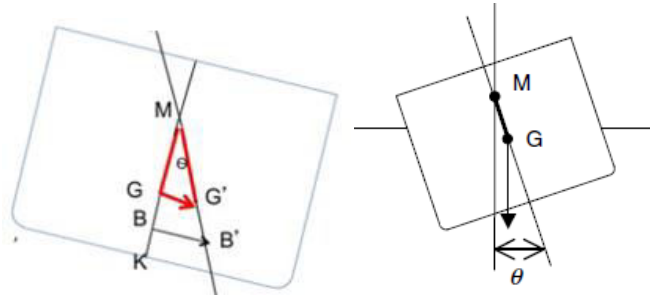


Figure IV-3. La distance métacentrique GM de la barge

En pratique, pour optimiser GM, les planificateurs d'arrimage cherchent à placer les conteneurs les plus légers sur les conteneurs les plus lourds.

3.3 Aspect multi-objectif du 3D-CCSPP

Notre contribution principale consiste à traiter de manière appropriée trois objectifs du 3D-CCSPP: la minimisation des mouvements *shiftings* et du nombre de barges réellement utilisées et la maximisation de la stabilité totale du convoi. Ces trois objectifs du problème constituent un ensemble de solutions pour les cas multiples qui ne peuvent pas simplement se comparer (Vrugt et Robinson, 2007). Cela donne lieu à un ensemble de solutions optimales de Pareto ou de solutions non dominées. Ces solutions ne peuvent être améliorées sans entraîner une dégradation simultanée d'au moins un autre objectif. Par ailleurs, l'optimisation multi-objectif appliquée au 3D-CCSPP offre certains avantages au problème par rapport à la méthode mono-objectif (Araújo et al., 2016). L'un de ces avantages est de permettre au planificateur de choisir la solution qui correspond le mieux à ses intérêts dans une situation donnée. Par exemple, si le fleuve/la mer est agité(e), il est judicieux de choisir une solution offrant un bon niveau de stabilité. Si le chemin entre deux ports est calme, nous pouvons choisir un plan de chargement qui optimise le nombre de *shiftings* pour éviter les pénalités et les retards dans les ports.

Afin d'illustrer la nature de l'optimisation multi-objectif du 3D-CCSPP, la **Figure IV-4**, la **Figure IV-5** et la **Figure IV-6** montrent le plan de chargement obtenu par le solveur CPLEX pour une instance donnée, en minimisant uniquement le nombre de *shiftings* total, en maximisant uniquement la stabilité totale et ensuite en minimisant uniquement le nombre de barges réellement utilisées dans le convoi, respectivement. L'instance proposée consiste en un convoi constitué de trois barges de dimensions $l = 2$, $L = 4$ et $Niv = 3$. Le nombre de ports à visiter est $P = 5$, et le nombre de conteneurs à charger est 11, où 4 conteneurs sont de taille 20' et les 7 restants de tailles 40'.

Chaque barge est illustrée par ses deux lignes, où chaque ligne est composée de quatre piles. Chaque carré de chaque pile contient le triplet (n, d, N) représentant ainsi un emplacement occupé par un conteneur, ou un caractère « - » représentant un espace vide, « n » indique l'identifiant du conteneur; « d » représente le port de destination du conteneur et « N » donne la taille du conteneur ($N1$ pour 40' et $N2$ pour 20').

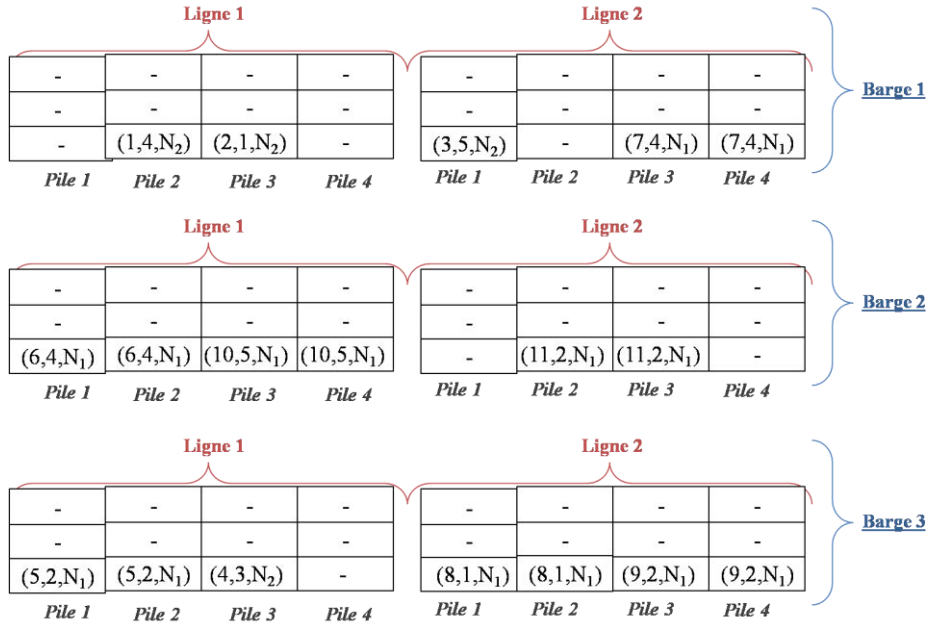


Figure IV-4. Le plan de chargement obtenu lorsque seule la stabilité est maximisée

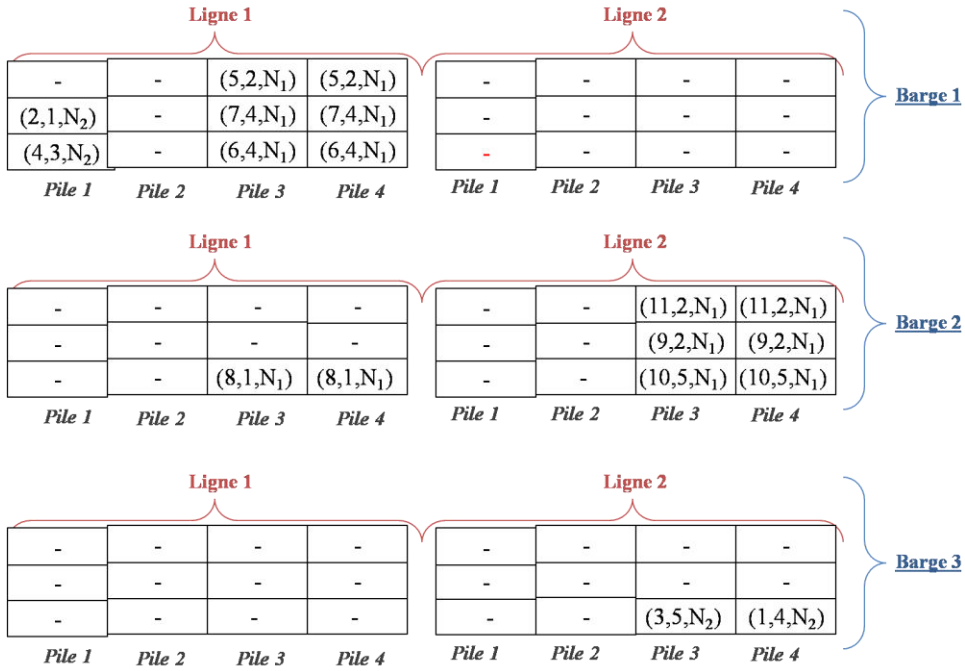


Figure IV-5. Le plan de chargement obtenu lorsque seul le nombre de *shiftings* est minimisé

Ligne 1				Ligne 2				Barge 1
-	(9,2,N ₁)	(9,2,N ₁)	(1,4,N ₂)	(6,4,N ₁)	(6,4,N ₁)	-	-	
-	(10,5,N ₁)	(10,5,N ₁)	(4,3,N ₂)	(5,2,N ₁)	(5,2,N ₁)	-	-	
(2,1,N ₂)	(8,1,N ₁)	(8,1,N ₁)	(3,5,N ₂)	(7,4,N ₁)	(7,4,N ₁)	(11,2,N ₁)	(11,2,N ₁)	
Pile 1	Pile 2	Pile 3	Pile 4	Pile 1	Pile 2	Pile 3	Pile 4	

Ligne 1				Ligne 2				Barge 2
-	-	-	-	-	-	-	-	
-	-	-	-	-	-	-	-	
-	-	-	-	-	-	-	-	
Pile 1	Pile 2	Pile 3	Pile 4	Pile 1	Pile 2	Pile 3	Pile 4	

Ligne 1				Ligne 2				Barge 3
-	-	-	-	-	-	-	-	
-	-	-	-	-	-	-	-	
-	-	-	-	-	-	-	-	
Pile 1	Pile 2	Pile 3	Pile 4	Pile 1	Pile 2	Pile 3	Pile 4	

Figure IV-6. Le plan de chargement obtenu lorsque seul le nombre de barges réellement utilisées est minimisé

À partir des **Figure IV-4**, **Figure IV-5** et **Figure IV-6**, nous pouvons confirmer que la priorité accordée à chaque fonction objectif influence fortement le plan d'arrimage obtenu dans chaque cas. Par exemple, dans les figures, **Figure IV-4** et **Figure IV-5**, la priorité donnée, respectivement, à la stabilité du convoi et au nombre de *shiftings* dans l'objectif à optimiser, pourrait avoir un impact considérable sur le plan de chargement des conteneurs en augmentant le nombre de barges réellement utilisées, nous remarquons que dans ces deux figures, toutes les trois barges sont utilisées contrairement à la **Figure IV-6** où seulement une seule barge est réellement utilisée.

4 Formulation mathématique du 3D-CCSPP

Dans cette section, en se basant sur le modèle mathématique du 3D-BCSPP proposé dans le chapitre III, nous proposons une modélisation mathématique multi-objectif non linéaire en nombres entiers pour résoudre le problème de planification des plans de chargement et d'arrimage des conteneurs dans un convoi de barges homogènes. Cette modélisation intègre des contraintes nouvelles et réalistes qui reflètent le fonctionnement réel lors de l'arrimage. D'abord, nous présentons quelques hypothèses et terminologie qui seront utilisées dans le modèle mathématique ultérieurement.

4.1 Hypothèses du problème

Les hypothèses de base utilisées dans ce chapitre sont listées comme suit:

- ▶ Chaque barge du système de convoi a un format rectangulaire et peut être représentée par une matrice avec des lignes $i = 1, \dots, l$, des baies $j = 1, \dots, L$ et des niveaux $h = 1, \dots, Niv$ avec une capacité maximale de $l \times L \times Niv$ EVP,
- ▶ Le nombre de barges disponibles dans le port est limité mais suffisant pour arrimer tous les conteneurs à transporter,
- ▶ Toutes les barges disponibles sont vides,
- ▶ Toutes les barges disponibles sont homogènes,
- ▶ Le nombre de barges constituant le convoi ne dépasse pas le nombre de barges disponibles dans le port,
- ▶ Le convoi de barges commence sa tournée à partir du port initial, pour lequel nous générons le plan d'arrimage, puis visite successivement un nombre fini de ports connu à l'avance où seules les opérations de déchargement sont autorisées,
- ▶ Chaque barge est divisée en un ensemble fini de piles. Chaque pile est définie par la combinaison de chaque ligne avec chaque baie, constituant ainsi un ensemble d'emplacements. Elle peut contenir jusqu'à un nombre maximal de conteneurs empilés verticalement,
- ▶ Chaque emplacement peut accueillir un conteneur de 20 pieds. Un EVP est considéré comme l'unité de discrétisation des emplacements à l'intérieur de la barge,
- ▶ Les coordonnées de chaque emplacement dans la barge sont données en unité « pieds »,
- ▶ Les conteneurs à arrimer sont de deux tailles différentes (40' et 20'),
- ▶ La fragilité de chaque contenant est supérieure à son poids.

4.2 Modèle mathématique

4.2.1 Paramètres d'entrée et notations

Afin de modéliser le 3D-CCSPP, nous proposons les notations suivantes :

4.2.1.1 Les indices

- ▶ $t := 1..B$ l'indice des barges
- ▶ $n := 1..|N|$ l'indice des conteneurs
- ▶ $i := 1..l$ l'indice des lignes de la barge ($i=1$ représente le côté bâbord de la barge)
- ▶ $j := 1..L$ l'indice des baies de la barge ($j=1$ représente la proue de la barge)
- ▶ $h := 1..Niv$ l'indice des niveaux de la barge ($h=1$ représente le niveau le plus bas de la barge)

4.2.1.2 Les paramètres d'entrée

N_1 : l'ensemble des conteneurs de taille 40'

N_2 : l'ensemble des conteneurs de taille 20'

B : le nombre de barges homogènes disponibles dans le port

N : l'ensemble de tous les conteneurs $N = N_1 \cup N_2$

L : longueur de la barge discrétisée en 20 pieds

l : largeur de la barge discrétisée en 8 pieds

Niv : le nombre maximal de niveaux dans la barge discrétisée en 8 pieds

m_h : la distance verticale entre le métacentre de la barge et le niveau h

w_n : le poids du conteneur n donné en tonne métrique

$f_{max} = \max_{j \in N} \{f_j\}$ la fragilité maximale parmi celles des conteneurs de l'instance considérée

f_n : la fragilité du conteneur n donnée en tonne métrique

d_n : le port de destination du conteneur n

$$des_{n,n'} = \begin{cases} 1 & \text{si } d_n \text{ est strictement avant } d_{n'}, \\ 0 & \text{sinon} \end{cases}$$

$Des = (des_{n,n'})_{1 \leq n, n' \leq |N|}$ la matrice qui définit la relation de succession entre n et n'

$$y_n = \begin{cases} 1 & \text{si le conteneur } n \text{ est de taille } N_1 \\ 0 & \text{si le conteneur } n \text{ est de taille } N_2 \end{cases}$$

4.2.2 Variables de décision

Avec ces notations et paramètres, une formulation mathématique multi-objectif non linéaire en nombres entiers du 3D-CCSP est construite en introduisant les trois variables ci-dessous :

$$V_{ijhn}^t = \begin{cases} 1 & \text{si le conteneur } n \text{ est chargé dans la position } (i, j, h) \text{ de la barge } t, \text{ et } n \in N_2 \\ 0 & \text{sinon} \end{cases}$$

$$U_{ijhn}^t = \begin{cases} 1 & \text{si le conteneur } n \text{ est chargé dans la position } (i, j, h) \text{ de la barge } t, \text{ et } n \in N_1 \\ 0 & \text{sinon} \end{cases}$$

$$b_t = \begin{cases} 1 & \text{si la barge } t \text{ est utilisée} \\ 0 & \text{sinon} \end{cases}$$

Nous notons que : $V : l \times L \times Niv \times N \times B \rightarrow \{0,1\}$ et $U : l \times L \times Niv \times N \times B \rightarrow \{0,1\}$

$$(i, j, h, n, t) \rightarrow V_{ijhn}^t \quad (i, j, h, n, t) \rightarrow U_{ijhn}^t$$

où :

$$\sum_{t=1}^B \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} V_{ijhn}^t = 0, \quad \forall n \in N_1 \quad (IV-2)$$

$$\sum_{t=1}^B \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} U_{ijhn}^t = 0, \quad \forall n \in N_2 \quad (IV-3)$$

4.2.3 Fonctions objectifs

Le 3D-CCSPP cherche à optimiser le plan de chargement d'un ensemble de conteneurs dans un convoi de barges selon trois fonctions objectifs :

La première fonction objectif (f_1) permet de minimiser le nombre total de *shiftings* effectués tout au long du trajet du convoi, elle peut être formulée selon l'équation (IV-4) :

$$f_1 = \text{Min} \sum_{t=1}^B \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} \left(1 - \frac{1}{2} y_n \right) \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Niv} des_{n,n'} (y_{n'} U_{ijh'n'}^t + (1 - y_{n'}) V_{ijh'n'}^t) \right) (y_n U_{ijhn}^t + (1 - y_n) V_{ijhn}^t) \quad (IV-4)$$

La deuxième fonction objectif (f_2) consiste à maximiser la stabilité totale du convoi en maximisant la somme des distances métacentriques de toutes les barges utilisées dans le convoi. Pour ce faire, nous adoptons la formulation du facteur de stabilité GM proposée par Imai et al. (2002), en définissant le ratio de contribution de GM (donné en pieds) comme suit:

$$G_{nh} = \frac{w_n m_h}{\sum_{n \in N} w_n} \quad \forall n \in N, \forall h \in \{1, \dots, Niv\} \quad (IV-5)$$

Alors, la fonction objectif f_2 peut être formulée selon l'équation (IV-6) :

$$f_2 = \text{Max} \sum_{t=1}^B \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} G_{nh} \left((1 - y_n) V_{ijhn}^t + \frac{1}{2} y_n U_{ijhn}^t \right) \quad (IV-6)$$

De plus, puisque dans le 3D-CCSPP tous les conteneurs doivent être chargés, la somme des poids des conteneurs est donc constante. Du coup, f_2 peut être reformulée par (IV-7):

$$f_2 = \text{Max} \left(\frac{1}{\sum_{n \in N} w_n} \right) \sum_{t=1}^B \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} \left(w_n m_h \left((1 - y_n) V_{ijhn}^t + \frac{1}{2} y_n U_{ijhn}^t \right) \right) \quad (IV-7)$$

4.2.3.1 Calculer la valeur de m_h

Compte tenu de la description de la structure physique de chaque emplacement dans la barge expliquée au niveau de la section 3 du chapitre III, et en se basant sur la définition de la distance métacentrique GM de la barge, nous formulons le paramètre m_h utilisé dans la deuxième fonction objectif.

D'abord, nous rappelons que, dans l'état d'équilibre de la barge, le métacentre M se trouve au dessus du centre de gravité G , les deux points sont situés sur le même axe vertical de la barge. En se basant sur les travaux de Imai et al., (2002, 2006), nous supposons que M se situe au centre géométrique du niveau le plus haut dont la position est $M(i, j, h) = \left(\frac{l}{2}, \frac{L}{2}, Niv\right)$ comme le montre la **Figure IV-7**. Par ailleurs, les coordonnées de M sont calculées selon la formule proposée dans le chapitre III par :

$$M(a_i, b_j, c_h) = (8i - 4, 20j - 10, 8h + 4)$$

$$M(a_i, b_j, c_h) = \left(\frac{8l}{2} - 4, \frac{20L}{2} - 10, 8Niv + 4\right) = (4l - 4, 10L - 10, 8Niv + 4)$$

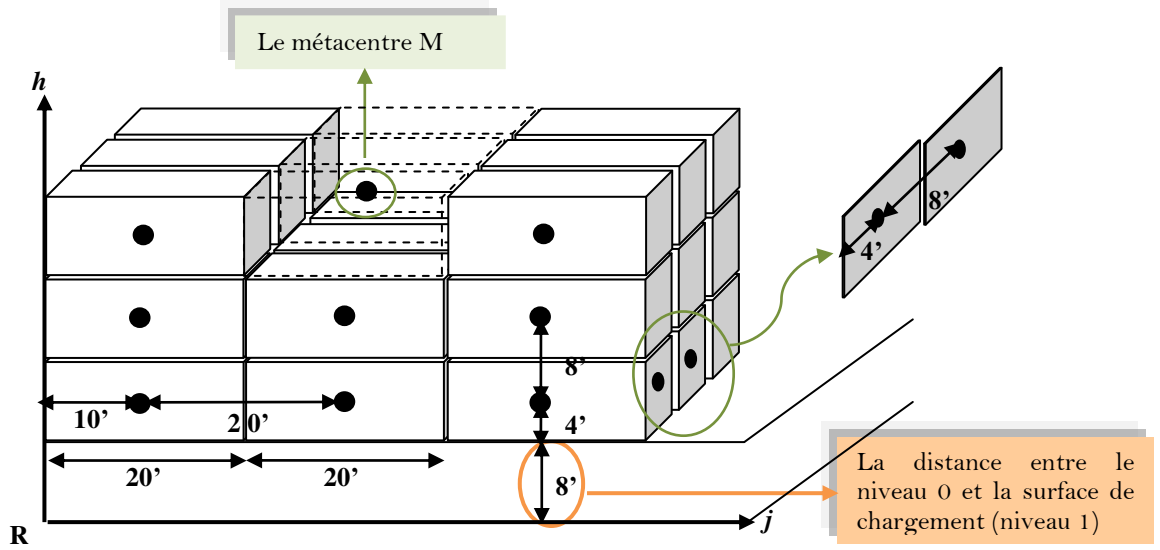


Figure IV-7. La position du métacentre M dans la barge

Il est à noter que comme GM est une distance verticale, tous les conteneurs d'un niveau particulier ont la même distance du métacentre. C'est pourquoi m_h ne dépendra que de l'axe hauteur, d'où la notation m_h . Ainsi, nous calculons la distance verticale entre le métacentre M de la barge et le niveau $h, \forall h \in \{1, \dots, Niv\}$ selon l'équation (IV-8) :

$$m_h = \sqrt{(8Niv + 4 - 8h - 4)^2} = 8|Niv - h| \quad \forall h \in \{1, \dots, Niv\} \quad (IV-8)$$

Par conséquent, la fonction objectif (IV-7) peut être remplacée par (IV-7)' comme suit :

$$f_2 = \text{Max} \left(\frac{8}{\sum_{n \in N} w_n} \right) \sum_{t=1}^B \sum_{i=1}^L \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} w_n |Niv - h| \left((1 - y_n) V'_{ijhn} + \frac{1}{2} y_n U'_{ijhn} \right) \quad (IV-7)'$$

Tandis que la troisième fonction objectif (f_3) est liée à l'aspect de *bin-packing* qui consiste à minimiser le nombre de barges réellement utilisées pour transporter l'ensemble de conteneurs (IV-9).

$$f_3 = \text{Min} \sum_{t=1}^B b_t \quad (\text{IV-9})$$

Les trois fonctions objectifs du modèle mathématique sont optimisées sous les contraintes décrites par la suite.

4.2.4 Contraintes

$$\sum_{t=1}^B \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} V_{ijhn}^t = 1 \quad \forall n \in N_2 \quad (\text{IV-10})$$

Les contraintes (IV-10) assurent que tous les conteneurs de taille 20'sont chargés dans le convoi et chaque conteneur est chargé dans un et un seul emplacement d'une seule et une seule barge.

$$\sum_{t=1}^B \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} U_{ijhn}^t = 2 \quad \forall n \in N_1 \quad (\text{IV-11})$$

$$U_{ij+1hn}^t - U_{ij-1hn}^t \leq U_{ijhn}^t (1 - U_{ij-1hn}^t) \quad \forall n \in N_1, \forall i \in \{1, \dots, l\}, \forall j \in \{2, \dots, L-1\}, \forall h \in \{1, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (\text{IV-12})$$

$$U_{ij+1hn}^t \geq U_{ijhn}^t (1 - U_{ij-1hn}^t) \quad \forall n \in N_1, \forall i \in \{1, \dots, l\}, \forall j \in \{2, \dots, L-1\}, \forall h \in \{1, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (\text{IV-13})$$

$$U_{i1hn}^t \leq U_{i2hn}^t \quad \forall n \in N_1, \forall i \in \{1, \dots, l\}, \forall h \in \{1, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (\text{IV-14})$$

$$U_{iLhn}^t \leq U_{iL-1hn}^t \quad \forall n \in N_1, \forall i \in \{1, \dots, l\}, \forall h \in \{1, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (\text{IV-15})$$

Les contraintes ((IV-11)-(IV-15)) imposent le fait que tous les conteneurs de taille 40'sont chargés et que chaque conteneur est chargé dans une et une seule barge occupant ainsi deux emplacements consécutifs le long de l'axe des baies de la barge correspondante.

$$b_t \geq y_n U_{ijhn}^t + (1 - y_n) V_{ijhn}^t \quad \forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (\text{IV-16})$$

Les contraintes (IV-16) dressent la liaison entre les variables de décisions, elles assurent que la barge t est utilisée si au moins l'une de ses positions est occupée.

$$(h-1) \left(\sum_{n \in N} (y_n U_{ijhn}^t + (1 - y_n) V_{ijhn}^t) \right) \leq \sum_{h'=1}^{h-1} \sum_{n' \in N} (y_{n'} U_{ijh'n'}^t + (1 - y_{n'}) V_{ijh'n'}^t) \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{2, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (\text{IV-17})$$

Les contraintes (IV-17) assurent que si un emplacement (i, j, h) de la barge t est occupé, tous les emplacements en-dessous doivent également être occupés.

$$1 - \left(y_n U_{ijhn}^t + (1 - y_n) V_{ijhn}^t \right) \geq \sum_{\substack{n' \in N \\ n' \neq n}} \left(y_{n'} U_{ijhn'}^t + (1 - y_{n'}) V_{ijhn'}^t \right) \quad (IV-18)$$

$$\forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}, \forall t \in \{1, \dots, B\}$$

Les contraintes (IV-18) indiquent que si la position (i, j, h) de la barge t est occupée, alors elle doit être occupée par un et un seul conteneur.

$$\sum_{h=1}^{Niv} \sum_{n \in N} \left(y_n U_{ijhn}^t + (1 - y_n) V_{ijhn}^t \right) \leq Niv \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall t \in \{1, \dots, B\} \quad (IV-19)$$

Les contraintes (IV-19) garantissent que le nombre de conteneurs chargés dans chaque pile de chaque barge du convoi ne dépasse pas le nombre maximal de niveaux dans la barge. Nous rappelons que les barges constituant le convoi sont homogènes.

$$\sum_{j=1}^L \sum_{n \in N} \left(y_n U_{ijhn}^t + (1 - y_n) V_{ijhn}^t \right) \leq L \quad \forall i \in \{1, \dots, l\}, \forall h \in \{1, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (IV-20)$$

Les contraintes (IV-20) indiquent que le nombre de conteneurs arrimés dans chaque ligne de chaque barge ne dépasse pas la longueur L .

$$\sum_{i=1}^l \sum_{n \in N} \left(y_n U_{ijhn}^t + (1 - y_n) V_{ijhn}^t \right) \leq l \quad \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (IV-21)$$

Les contraintes (IV-21) indiquent que le nombre de conteneurs arrimés dans chaque baie de chaque barge ne dépasse pas la largeur l .

$$\sum_{n \in N_2} V_{ijhn}^t \leq \sum_{n \in N_2} V_{ijh-1n}^t \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{2, \dots, Niv\}, \forall t \in \{1, \dots, B\} \quad (IV-22)$$

Comme expliqué dans le chapitre III, tous les conteneurs sont équipés de pièces de coins imposant ainsi les contraintes (IV-22) qui assurent que les conteneurs de 20' ne peuvent pas être empilés sur les conteneurs de 40'.

$$\begin{aligned} & \left(1 - \sum_{n \in N_2} V_{ijhn}^t \right) \left(\sum_{n' \in N_1} U_{ijh+1n'}^t + \sum_{n \in N_2} V_{ijhn}^t \right) + \sum_{n \in N_2} V_{ijhn}^t \left(\sum_{n' \in N_1} U_{ijh+1n'}^t + \sum_{n \in N_2} V_{ijhn}^t - 1 \right) \leq \sum_{n'' \in N_2} V_{ij+1hn''}^t \left(\sum_{n \in N_2} V_{ijhn}^t \right) \\ & + \sum_{n'' \in N_1} U_{ijhn''}^t \left(1 - \sum_{n \in N_2} V_{ijhn}^t \right) \quad \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L-1\}, \forall h \in \{1, \dots, Niv-1\}, \forall t \in \{1, \dots, B\} \end{aligned} \quad (IV-23)$$

Les contraintes (IV-23) montrent que si l'emplacement (i, j, h) d'une barge donnée t est occupé par un conteneur de taille 20' et que l'emplacement $(i, j, h + 1)$ de la même barge est occupé par un conteneur de 40' alors l'emplacement $(i, j + 1, h)$ de t doit être occupé par un conteneur 20'.

$$\begin{aligned} & \sum_{h'=h+1}^{Niv} \sum_{\substack{n' \in N \\ n' \neq n}} w_{n'} \left(\frac{1}{2} y_{n'} U_{ijh'n'}^t + (1 - y_{n'}) V_{ijh'n'}^t \right) \leq f_{\max} + \frac{1}{2} y_n U_{ijhn}^t (f_n - 2f_{\max}) + (1 - y_n) V_{ijhn}^t (f_n - f_{\max}) \\ & \forall n \in N, \forall i \in \{1, \dots, l\}, \forall j \in \{1, \dots, L\}, \forall h \in \{1, \dots, Niv-1\}, \forall t \in \{1, \dots, B\} \end{aligned} \quad (IV-24)$$

Les contraintes (IV-24) sont les contraintes de fragilité qui assurent que la fragilité de chaque conteneur placé dans une pile d'une barge donnée doit être supérieure ou égale à la somme des poids de tous les conteneurs placés au-dessus.

$$U_{ijhn}^t \in \{0,1\} \quad \forall n \in N_1, \forall i \in \{1,...,l\}, \forall j \in \{1,...,L\}, \forall h \in \{1,...,Niv\}, \forall t \in \{1,...,B\} \quad (IV-25)$$

$$V_{ijhn}^t \in \{0,1\} \quad \forall n \in N_2, \forall i \in \{1,...,l\}, \forall j \in \{1,...,L\}, \forall h \in \{1,...,Niv\}, \forall t \in \{1,...,B\} \quad (IV-26)$$

$$b_t \in \{0,1\} \quad \forall t \in \{1,...,B\} \quad (IV-27)$$

Les contraintes ((IV-25) – (IV-27)) sont les contraintes d'intégrité.

- **Remarque :** Comme dans le chapitre III, nous notons $Z_{ijhn}^t = y_n U_{ijhn}^t + (1 - y_n) V_{ijhn}^t$, nous pouvons donner, alors, un nouveau modèle mathématique en remplaçant la fonction objectif (IV-4) par (IV-28), et cela en introduisant une troisième variable $W_{ijhnh'n'}^t$ telle que $W_{ijhnh'n'}^t = Z_{ijhn}^t * Z_{ijh'n'}^t$ avec les contraintes (IV-29) et (IV-30) :

$$f_1 = \text{Min} \sum_{t=1}^B \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv} \sum_{n \in N} \left(1 - \frac{1}{2} y_n \right) \left(\sum_{\substack{n' \in N \\ n' \neq n}} \sum_{h'=h+1}^{Niv} des_{n,n'} W_{ijhnh'n'}^t \right) \quad (IV-28)$$

$$W_{ijhnh'n'}^t \geq Z_{ijhn}^t + Z_{ijh'n'}^t - 1 \quad \forall t \in \{1,...,B\}, \forall i \in \{1,...,l\}, \forall j \in \{1,...,L\}, \forall h \in \{1,...,Niv\}, \forall n \in N, \forall h' \in \{1,...,Niv\}, \forall n' \in N \quad (IV-29)$$

$$W_{ijhnh'n'}^t \leq \frac{1}{2} (Z_{ijhn}^t + Z_{ijh'n'}^t) \quad \forall t \in \{1,...,B\}, \forall i \in \{1,...,l\}, \forall j \in \{1,...,L\}, \forall h \in \{1,...,Niv\}, \forall n \in N, \forall h' \in \{1,...,Niv\}, \forall n' \in N \quad (IV-30)$$

5 Méthodes de résolution

Il est bien connu que le problème de chargement et d'arrimage des conteneurs est un problème d'optimisation NP-difficile (Avriel et al., 2000). Un bon plan d'arrimage doit placer chaque conteneur dans l'emplacement le plus convenable possible. Cependant, l'optimisation d'un tel plan est difficile et complexe car les possibilités sont très nombreuses. Par exemple, pour un petit porte-conteneurs d'une capacité de 20 EVP seulement et de 20 conteneurs en attente de chargement, il existe $2,4 \times 10^{18}$ possibilités. Par ailleurs, un convoi, par exemple, de 5 barges et de 100 conteneurs à charger offre encore plus de possibilités. Il peut donc être difficile d'utiliser des méthodes exactes pour résoudre ces problèmes dans des délais raisonnables. Par conséquent, nous avons utilisé les approches méta-heuristiques pour le résoudre. En règle générale, les méta-heuristiques fournissent une solution satisfaisante mais

pas nécessairement optimale. Les algorithmes développés dans ce chapitre sont basés sur le concept multi-objectif en intégrant un ensemble d'heuristiques de *bin-packing* en 1D. Une description détaillée de ces algorithmes est donnée dans les sections suivantes.

5.1 Aspect multi-objectif de l'approche de résolution

Comme bien expliqué, le 3D-CCSPP est un problème d'optimisation multi-objectif visant à optimiser les trois fonctions objectifs suivantes:

- ▶ Minimiser le nombre total de *shiftings* durant toute la tournée,
- ▶ Maximiser la stabilité totale du convoi selon le facteur de la distance métacentrique,
- ▶ Minimiser le nombre de barges réellement utilisées dans le convoi.

Afin d'obtenir la meilleure solution, nous devons être en mesure d'évaluer chacune de ces fonctions et de déterminer laquelle des différentes alternatives trouvées répond mieux à tous les objectifs. Pour résoudre ce problème, nous utiliserons des algorithmes évolutionnaires qui étudient différentes alternatives et les font «évoluer» jusqu'à ce qu'un critère d'arrêt soit atteint. De ce fait, ce chapitre présente une nouvelle alternative pour résoudre le 3D-CCSPP en cherchant la frontière optimale au sens de Pareto. Ce cadre d'optimisation est basé sur des techniques de calcul évolutif, en particulier, pour les algorithmes évolutionnaires multi-objectif NSGA-II, qui est au cœur des méthodes proposées. Outre sa simplicité, l'utilisation du NSGA-II dans ce chapitre s'explique par le fait qu'il n'a pas été encore appliqué au problème de chargement et d'arrimage des conteneurs en tenant compte des cinq facteurs: taille des conteneurs, limite dynamique du poids de la pile, nombre de *shiftings*, nombre de barges dans le convoi et la stabilité totale. Nous concevons trois nouvelles adaptations du NSGA-II en se basant sur deux heuristiques de *bin-packing* pour résoudre le 3D-CCSPP. Pour ce faire, nous rappelons tout d'abord le concept de base de l'algorithme NSGA-II.

5.2 Principe général de la méthode NSGA-II

Introduit par Deb et al. (2002), le NSGA-II est un algorithme de recherche robuste basé sur les principes de l'évolution biologique. En essence, il est basé sur l'algorithme génétique qui est une simulation du processus évolutif utilisant les puissants opérateurs de sélection, de mutation et de croisement « naturels » pour apporter une solution à tout problème de conception complexe.

Au fil des années, l'approche évolutionnaire NSGA-II a montré de très bonnes performances en termes d'efficacité et d'efficacité pour traiter les problèmes à plusieurs objectifs. En effet, à chaque itération (également appelée génération), une nouvelle population de solutions possibles nommée population enfant « *offspring population* » est créée à partir de la population parent. Les individus appartenant à la population enfant sont évalués à l'aide des fonctions objectifs définies dans le problème, puis comparés à la population parent afin de choisir ceux qui correspondent le mieux aux exigences du problème. Un ensemble d'individus remplace

les anciens suivant certains critères afin de poursuivre le processus d'évolution jusqu'à ce qu'une condition d'arrêt soit satisfaite.

Dans le contexte multi-objectif, l'approche évolutionnaire NSGA-II offre, par le biais de la notion de population, des mécanismes pertinents pour approcher la solution optimale au sens de Pareto, à savoir, la sélection Pareto, l'élitisme et la diversification, plus de détails sont donnés dans (Deb et al., 2002) et (Tan et al., 2002).

5.3 Adaptation du NSGA-II pour le 3D-CCSPP

Dans cette section, nous adaptons les différentes étapes de la méta-heuristique NSGA-II, illustrées dans la **Figure IV-8**, au problème d'optimisation des plans de chargement et d'arrimage dans un convoi de barges.

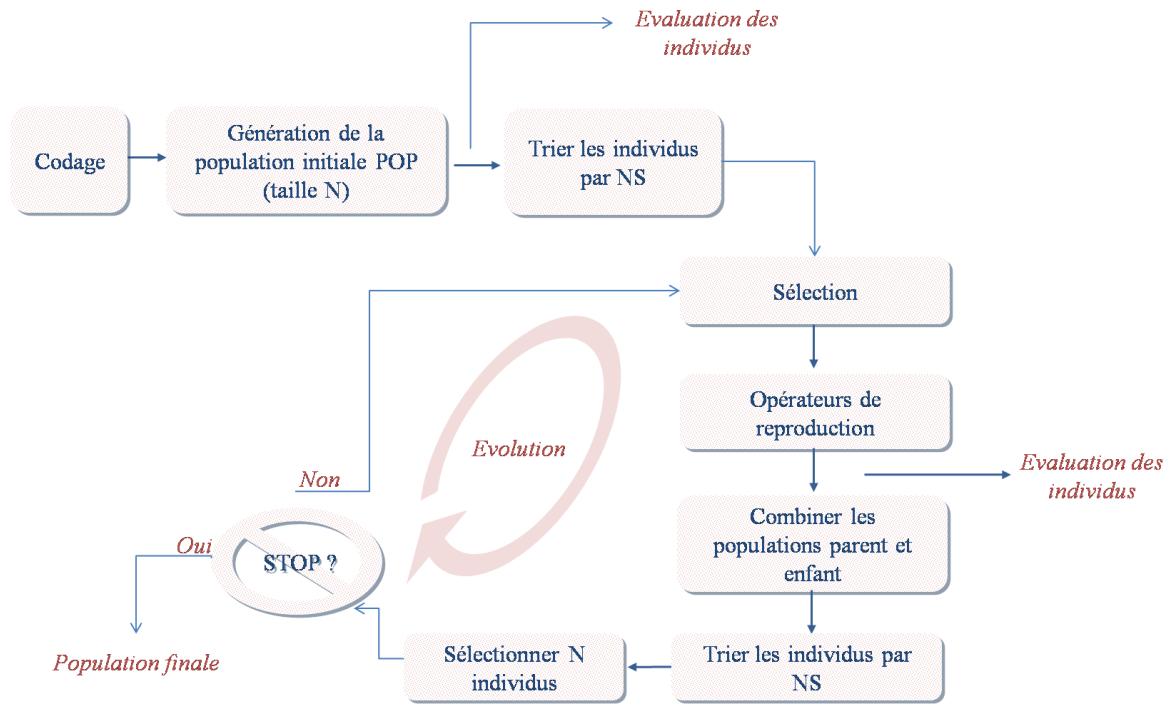


Figure IV-8. Schéma explicatif des étapes du NSGA-II

5.3.1 Codage d'un individu

La première étape du NSGA-II consiste à coder et définir convenablement chaque individu de la population. Cette étape associe à chaque point de l'espace de recherche une structure de données spécifique, appelée chromosome. Dans le 3D-CCSPP, chaque solution est représentée par un ensemble de chromosomes artificiels indiquant les lignes de chaque barge du convoi. Ils sont illustrés sous forme de plusieurs rangées, où chaque rangée représente une barge. Nous notons que le nombre de rangées est égale au nombre de barges disponibles dans le port, le nombre de chromosomes constituant une rangée égale au nombre de lignes de la barge correspondante et le nombre total de chromosomes constituant

l'individu est égal au nombre de lignes de tout le convoi. Chaque chromosome (ligne) est composé d'un ensemble de colonnes représentant les piles, où le nombre de colonnes de chaque ligne est défini par le nombre de baies de la barge correspondante. Chaque pile est constituée d'un ensemble de gènes indiquant ainsi les emplacements (ou positions) disponibles dans chaque barge, où le nombre de gènes dans chaque pile est égal au nombre de niveaux de la barge correspondante. Nous avons ainsi deux cas :

- Dans le cas où l'emplacement est occupé, le gène est illustré sous forme d'un carré contenant un triplet (n, d_n, N_l) avec « n » l'identifiant du conteneur, « d_n » le port de destination du conteneur et « $N_{l,l \in \{1,2\}}$ » la taille du conteneur où l'indice 1 indique la taille 40' et l'indice 2 représente 20',
- Dans le cas où l'emplacement est vide, le gène est illustré sous forme d'un carré contenant le caractère « - ».

Nous illustrons, dans la **Figure IV-9**, un exemple explicatif du codage proposé.

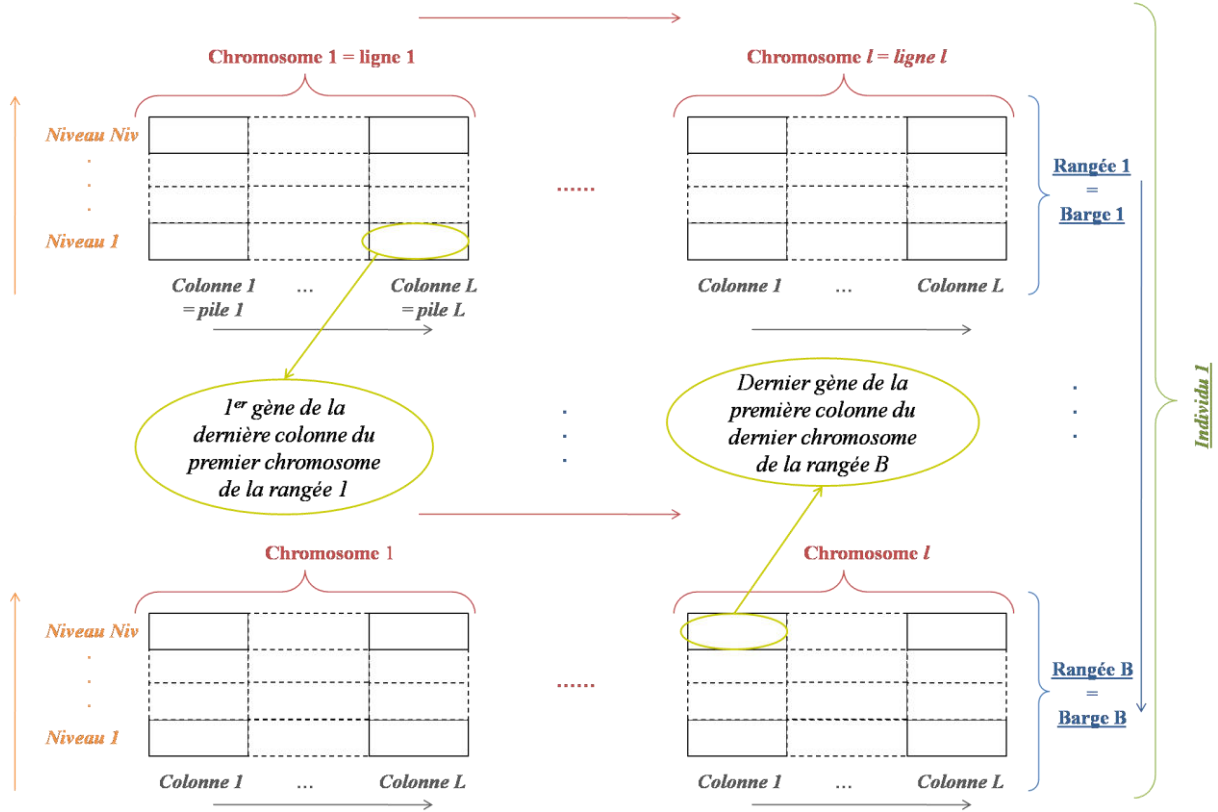


Figure IV-9. Illustration du codage proposé pour le 3D-CCSPP

Une telle représentation offre certaines facilités. En effet, le génotype devient structuré et peut être décomposé en différentes parties identifiables permettant ainsi son exploitation par les opérateurs de croisement et de mutation.

5.3.1.1 Exemple illustratif d'un individu

Etant donnée deux barges disponibles dans le port initial, chacune ayant une capacité maximale de 18 EVP avec une largeur $l = 2$, une longueur $L = 3$ et une hauteur $Niv = 3$ et un ensemble N de conteneurs à transporter, $N = N_1 \cup N_2$ où $N_1 = \{7,8,9,10,11\}$ et $N_2 = \{1,2,3,4,5,6\}$, nous illustrons un exemple de solution réalisable de cette instance dans la **Figure IV-10**. Par exemple, le deuxième gène, à partir du bas, de la première et la deuxième colonne (pile), à partir de la gauche, du chromosome 1 (ligne 1), à partir de la gauche, de la première rangé (barge), à partir du haut, possède le triplet $(7,5,N_1)$, ce qui signifie que ces deux positions ou gènes sont occupés par le conteneur 7 de taille 40 pieds et de destination 5. Le troisième gène, à partir du bas, de la troisième colonne, à partir de la gauche, du même chromosome contient le caractère « - », ce qui signifie que cette position est vide.

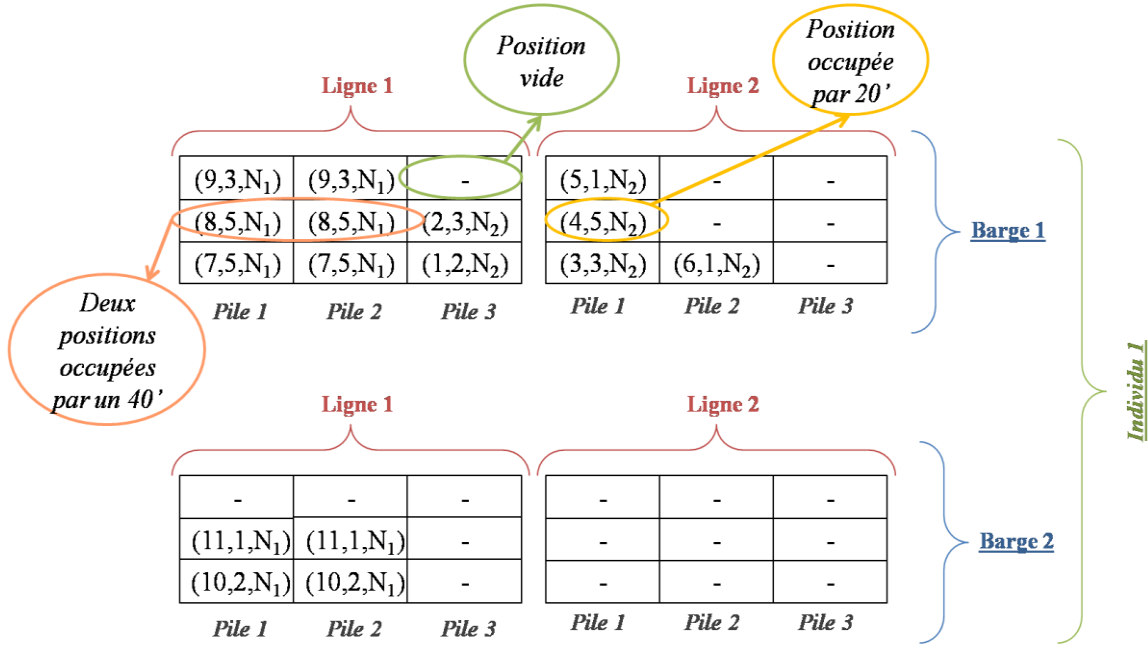


Figure IV-10. Illustration d'une solution réalisable du 3D-CCSPP (non optimale)

5.3.2 Génération de la population initiale

Nous notons que le NSGA-II dépend fortement de la population initiale qui est souvent générée d'une façon aléatoire, cependant, lorsque le problème à résoudre est soumis à plusieurs contraintes, il devient difficile de trouver une solution réalisable aléatoirement. C'est le cas du 3D-CCSPP. Pour cette raison, nous avons conçu deux heuristiques de chargement basées sur le problème de *bin-packing* en 1D pour générer la population initiale de notre problème.

5.3.2.1 Règles de chargement

Les heuristiques proposées consistent à ranger les conteneurs itérativement dans les barges. Pour ce faire, nous introduisons d'abord deux types de traitement:

- ▶ « RT » : l'ordre dans lequel les conteneurs sont examinés
- ▶ « RP » : la barge et la position dans laquelle nous cherchons à placer le conteneur

Pour chaque type, nous proposons, respectivement, deux règles de chargement « RT1 », « RT2 », « RP1 » et « RP2 » expliquées comme suit :

- ▶ **Règle de chargement « RT1 »** : L'ensemble de conteneurs N à charger est trié, d'abord par taille, en donnant la priorité aux conteneurs de taille 20', puis par ordre décroissant du poids. Autrement dit, lors du chargement, nous commençons par ranger les conteneurs 20' en premier, dans l'ordre décroissant de leur poids, puis, nous passons à ranger les 40' suivant le même ordre.
- ▶ **Règle de chargement « RT2 »** : L'ensemble de conteneurs N à charger est trié, d'abord par ordre décroissant du poids, puis par taille, c.-à-d., nous trions premièrement les conteneurs par ordre décroissant de leur poids, puis, pour l'ensemble de conteneurs avec le même poids, nous favorisons le chargement des 20' en premier.

Les deux règles de chargement (« RT1 » et « RT2 ») peuvent aider à éviter les mauvaises situations où les conteneurs de 20' sont arrimés sur les 40', ou bien, lorsque les conteneurs les plus lourds entreposent les conteneurs les plus légers. Cependant, pour une instance donnée, lors de la création de la population, elles ne peuvent générer qu'un seul individu. Ainsi, afin d'introduire la diversité dans les individus constituant la population initiale, nous ajoutons la notion du choix aléatoire des conteneurs à charger en premier, donnant ainsi deux nouvelles règles « RTN1 » et « RTN2 ».

- ▶ **Règle de chargement « RTN1 »** : Les n premiers conteneurs à charger de l'ensemble N ($n > 2$) sont choisis d'une façon aléatoire, puis le reste est chargé en respectant la règle RT1.
- ▶ **Règle de chargement « RTN2 »** : Les n premiers conteneurs à charger de l'ensemble N ($n > 2$) sont choisis d'une façon aléatoire, puis le reste est chargé en respectant la règle RT2.
- ▶ **Règle de chargement « RP1 »** : Suivant l'ordre indiqué dans la **Figure IV-11** , les conteneurs sont chargés ligne par ligne en commençant par le milieu de la barge, selon l'axe largeur, et continuant d'une façon alternative vers les deux cotés bâbord et tribord, pile par pile, en remplissant la pile qui se trouve au milieu de la ligne, selon l'axe longueur, et continuant vers les deux extrémités proue et poupe de la barge d'une manière alternative, et cela en partant du niveau inférieur de chaque pile vers le niveau le plus haut. Chaque gène contient un nombre entre deux parenthèses

indiquant l'ordre des positions de la barge. De plus, cette règle de chargement est basée sur l'heuristique First Fit du *bin-packing*, que nous expliquerons par la suite.

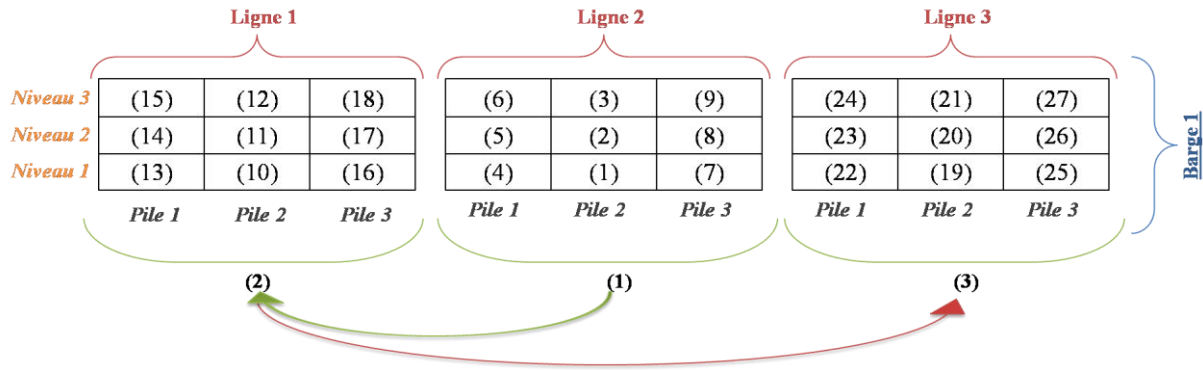


Figure IV-11. Illustration de la règle de chargement RP1

- **Règle de chargement « RP2 » :** En suivant l'ordre indiqué dans la **Figure IV-12**, les conteneurs sont chargés ligne par ligne en commençant par la première ligne du côté bâbord de la barge puis continuant d'une façon séquentielle vers le côté tribord, pile par pile, en remplissant la pile qui se trouve au coin bas-gauche de la proue et continuant vers la poupe de la barge d'une manière séquentielle, et cela en partant du niveau inférieur de chaque pile vers le niveau le plus haut. De même, cette règle suit le principe de l'heuristique First Fit du *bin-packing*.

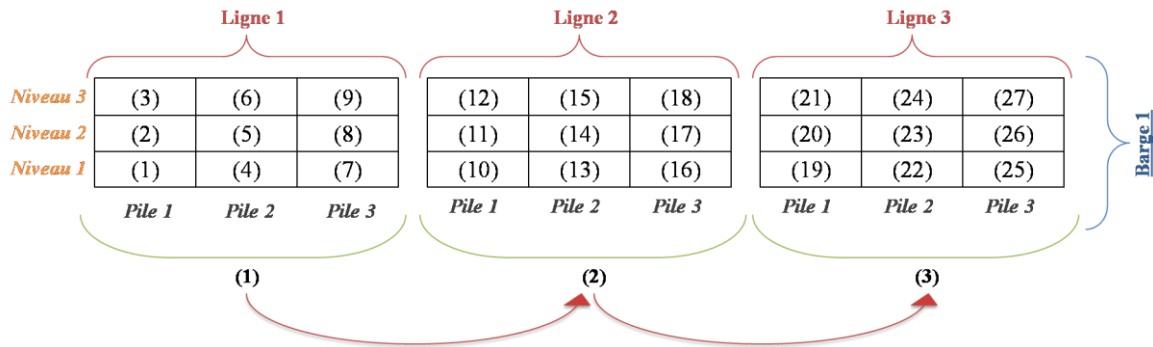


Figure IV-12. Illustration de la règle de chargement RP2

5.3.2.2 Heuristique de chargement des conteneurs dans une seule barge H_UniBarge

Pour arrimer un ensemble de conteneurs dans une barge donnée, nous proposons l'heuristique H_UniBarge qui se base sur les règles de chargements RTN1, RTN2, RP1 et RP2. L'algorithme 1 explique les différentes étapes de H_UniBarge d'une façon détaillée.

Algorithm 1: Structure générale de l'heuristique constructive H_UniBarge.

Entrée

Appliquer la règle de chargement de conteneurs selon le type de traitement RT (*)

pour chaque conteneur c **faire**

S = piles non vides de hauteur $< Niv$

$c(s_i)$ = le conteneur le plus haut dans la pile $s_i \in S$

si $c \in N_2$ **alors**

$S_{N2} = \{s_i / s_i \in S \text{ et } c(s_i) \in N_2 \text{ et } w(c) \leq Fpile(s_i)\}$

si $S_{N2} \neq \emptyset$ **alors**

$S_{N2}^{min} = \text{argmin}_{s \in S_{N2}} (des(c(s)))$ (pile qui a la destination la plus proche de celle de c)

Charger le conteneur c dans la pile S_{N2}^{min}

Mettre à jour la fragilité et la hauteur de la pile S_{N2}^{min}

sinon

Charger le conteneur c dans une pile vide selon le type de traitement RP (**)

Mettre à jour la fragilité et la hauteur de la pile

fin si

sinon

$Sc_{N1} = \{(s_i, s_{i+1}) / s_i, s_{i+1} \in S \text{ et } c(s_i) = c(s_{i+1}) \in N_1 \text{ et } \frac{1}{2}w(c) \leq \min(Fpile(s_i), Fpile(s_{i+1}))\}$

si $Sc_{N1} \neq \emptyset$ **alors**

$Sc_{N1}^{min} = \text{argmin}_{s \in Sc_{N1}} (des(c(s)))$

Charger le conteneur c dans la pile Sc_{N1}^{min}

Mettre à jour la fragilité et la hauteur de la pile Sc_{N1}^{min}

sinon

$Sc_{N2} = \{(s_i, s_{i+1}) / s_i, s_{i+1} \in S \text{ et } c(s_i), c(s_{i+1}) \in N_2 \text{ avec } c(s_i) \neq c(s_{i+1}) \text{ et niveau}(s_i) = \text{niveau}(s_{i+1}) \text{ et } \frac{1}{2}w(c) \leq \min(Fpile(s_i), Fpile(s_{i+1}))\}$

si $Sc_{N2} \neq \emptyset$ **alors**

$Sc_{N2}^{min} = \text{argmin}_{s \in Sc_{N2}} (des(c(s)))$

Charger le conteneur c dans la pile Sc_{N2}^{min}

Mettre à jour la fragilité et la hauteur de la pile Sc_{N2}^{min}

sinon

Charger le conteneur c dans une pile vide selon le type de traitement RP (**)

Mettre à jour la fragilité et la hauteur de la pile

fin si

fin si

fin si

fin pour

5.3.2.3 Randomiser l'heuristique constructive H_UniBarge

Afin d'introduire encore plus de diversité aux solutions générées par l'heuristique H_UniBarge, nous utilisons deux procédures de randomisation supplémentaires. D'une part, puisque nous avons développé deux règles de chargement constructives, RTN1 et RTN2, nous en choisissons une au hasard à chaque itération de l'algorithme 1 pour élaborer la solution indiquée par (*). D'autre part, au lieu de toujours commencer le chargement par le coin bas-gauche de la proue, nous choisissons au hasard, à chaque itération, une règle de chargement de conteneurs selon l'ordre des positions de la barge RP1 et RP2, ce qui est indiquée par (**) dans l'algorithme 1.

5.3.2.4 Heuristique de chargement des conteneurs dans un convoi de barges H_MultiBarge

Pour arrimer un ensemble de conteneurs dans un ensemble de barges constituant un convoi, nous proposons l'heuristique H_MultiBarge qui hybride l'heuristique H_UniBarge avec les heuristiques du problème de *bin-packing* en une dimension.

Pour ce faire, nous décrivons d'abord les deux heuristiques utilisées pour le cas unidimensionnel, qui ont inspiré la construction de l'algorithme 2. Auparavant, nous donnons quelques précisions :

- ▶ Chaque barge du convoi représente un *bin*
 - ▶ Chaque conteneur représente un objet
 - ▶ Tous les *bins* sont vides au début
 - ▶ Chaque *bin* qui n'est pas vide, est soit fermé, soit ouvert
 - ▶ Un *bin* est ouvert si l'on peut encore mettre des objets dedans
 - ▶ Un *bin* fermé n'est plus disponible pour recevoir des objets; une fois fermé, il n'est plus rouvert
-
- ▶ **Heuristique Next Fit (NF) :** Dans cette méthode nous ne considérons qu'une barge ouverte à la fois. Les conteneurs sont traités selon un ordre donné en se basant sur les règles RTN1 et RTN2. Les conteneurs sont arrimés successivement dans la barge ouverte tant qu'il y a de la place pour le conteneur **c** en cours, sinon la barge en cours est fermée et une nouvelle barge est ouverte. La complexité temporelle de cette heuristique peut être linéaire en fonction du nombre de conteneurs à placer. Par contre, le fait de ne considérer qu'une seule barge à la fois peut causer beaucoup de perte d'espaces exploitables.
 - ▶ **Heuristique First Fit (FF) :** Initialement une seule barge est considérée, et les conteneurs sont traités selon un ordre donné (RTN1 et RTN2). Quand il n'y a plus de la place dans la première barge pour ranger le conteneur en cours, une deuxième barge est alors ouverte mais sans fermer la première. Dans une étape intermédiaire où nous disposons d'un ensemble **B** de barges ouvertes numérotées de **1** à **|B|** selon l'ordre de leur première utilisation, un conteneur **c** en cours est rangé dans la barge du plus petit numéro qui peut le contenir. Dans le cas où aucune barge ne peut contenir **c**, une nouvelle barge (**|B| + 1**) est alors utilisée sans fermer les autres. L'ordre selon lequel les conteneurs sont traités est crucial pour la qualité de la solution.

En combinant l'heuristique H_UniBarge avec celles de *bin-packing* qui viennent d'être présentées, nous obtenons deux heuristiques de chargement de conteneurs dans un convoi de barges HNF_MultiBarge et HFF_MultiBarge. Plus de détails sont expliqués respectivement dans les algorithmes 2 et 3.

Algorithm 2: Structure générale de l'heuristique constructive HNF_MultiBarge.

Entrée

Ouvrir la première barge du convoi

Appliquer la règle de chargement de conteneurs selon le type de traitement RT (*)

pour chaque conteneur c **faire**

si il ya de la place dans la barge ouverte courante **faire**

S = piles non vides de hauteur $< Niv$

$c(s_i)$ = le conteneur qui se trouve dans le niveau le plus haut de la pile $s_i \in S$

si $c \in N_2$ **alors**

$S_{N2} = \{s_i / s_i \in S \text{ et } c(s_i) \in N_2 \text{ et } w(c) \leq Fpile(s_i)\}$

si $S_{N2} \neq \emptyset$ **alors**

$S_{N2}^{min} = \text{argmin}_{s \in S_{N2}} (des(c(s)))$ (pile qui a la destination la plus proche de celle de c)

 Charger le conteneur c dans la pile S_{N2}^{min}

 Mettre à jour la fragilité et la hauteur de la pile S_{N2}^{min}

sinon

 Charger le conteneur c dans une pile vide selon le type de traitement RP (**)

 Mettre à jour la fragilité et la hauteur de la pile

fin si

sinon

$Sc_{N1} = \{ (s_i, s_{i+1}) / s_i, s_{i+1} \in S \text{ et } c(s_i) = c(s_{i+1}) \in N_1 \text{ et } \frac{1}{2}w(c) \leq \min(Fpile(s_i), Fpile(s_{i+1})) \}$

si $Sc_{N1} \neq \emptyset$ **alors**

$Sc_{N1}^{min} = \text{argmin}_{s \in Sc_{N1}} (des(c(s)))$

 Charger le conteneur c dans la pile Sc_{N1}^{min}

 Mettre à jour la fragilité et la hauteur de la pile Sc_{N1}^{min}

sinon

$Sc_{N2} = \{ (s_i, s_{i+1}) / s_i, s_{i+1} \in S \text{ et } c(s_i), c(s_{i+1}) \in N_2 \text{ avec } c(s_i) \neq c(s_{i+1}) \text{ et niveau}(s_i) = \text{niveau}(s_{i+1}) \text{ et } \frac{1}{2}w(c) \leq \min(Fpile(s_i), Fpile(s_{i+1})) \}$

si $Sc_{N2} \neq \emptyset$ **alors**

$Sc_{N2}^{min} = \text{argmin}_{s \in Sc_{N2}} (des(c(s)))$

 Charger le conteneur c dans la pile Sc_{N2}^{min}

 Mettre à jour la fragilité et la hauteur de la pile Sc_{N2}^{min}

sinon

 Charger le conteneur c dans une pile vide selon le type de traitement RP (**)

 Mettre à jour la fragilité et la hauteur de la pile

fin si

fin si

fin si

sinon

 Ouvrir une nouvelle barge et fermer l'ancienne

$c \leftarrow$

fin si

fin pour

Algorithm 3: Structure générale de l'heuristique constructive HFF_MultiBarge.

Entrée

Ouvrir la première barge du convoi

Appliquer la règle de chargement de conteneurs selon le type de traitement RT (*)

$B = \{\text{barge ouvertes}\}$

pour chaque conteneur c **faire**

pour chaque barge b allant de 1 à $|B|$ **faire**

si il ya de la place dans la barge b **faire**

$S =$ piles non vides de hauteur $< Niv$

$c(s_i) =$ le conteneur qui se trouve dans le plus haut niveau de la pile $s_i \in S$

si $c \in N_2$ **alors**

$S_{N2} = \{s_i / s_i \in S \text{ et } c(s_i) \in N_2 \text{ et } w(c) \leq Fpile(s_i)\}$

si $S_{N2} \neq \emptyset$ **alors**

$S_{N2}^{min} = \text{argmin}_{s \in S_{N2}} (des(c(s)))$ (pile qui a la destination la plus proche de celle de c)

 Charger le conteneur c dans la pile S_{N2}^{min}

 Mettre à jour la fragilité et la hauteur de la pile S_{N2}^{min}

break

sinon

 Charger le conteneur c dans une pile vide selon le type de traitement RP (**)

 Mettre à jour la fragilité et la hauteur de la pile

break

fin si

sinon

$Sc_{N1} = \{(s_i, s_{i+1}) / s_i, s_{i+1} \in S \text{ et } c(s_i) = c(s_{i+1}) \in N_1 \text{ et } \frac{1}{2}w(c) \leq \min(Fpile(s_i), Fpile(s_{i+1}))\}$

si $Sc_{N1} \neq \emptyset$ **alors**

$Sc_{N1}^{min} = \text{argmin}_{s \in Sc_{N1}} (des(c(s)))$

 Charger le conteneur c dans la pile Sc_{N1}^{min}

 Mettre à jour la fragilité et la hauteur de la pile Sc_{N1}^{min}

break

sinon

$Sc_{N2} = \{(s_i, s_{i+1}) / s_i, s_{i+1} \in S \text{ et } c(s_i), c(s_{i+1}) \in N_2 \text{ avec } c(s_i) \neq c(s_{i+1}) \text{ et niveau}(s_i) = \text{niveau}(s_{i+1}) \text{ et } \frac{1}{2}w(c) \leq \min(Fpile(s_i), Fpile(s_{i+1}))\}$

si $Sc_{N2} \neq \emptyset$ **alors**

$Sc_{N2}^{min} = \text{argmin}_{s \in Sc_{N2}} (des(c(s)))$

 Charger le conteneur c dans la pile Sc_{N2}^{min}

 Mettre à jour la fragilité et la hauteur de la pile Sc_{N2}^{min}

break

sinon

 Charger le conteneur c dans une pile vide selon le type de traitement RP (**)

 Mettre à jour la fragilité et la hauteur de la pile

break

fin si

fin si

fin si

fin si

fin pour

si c n'est pas encore chargé

 Ouvrir une nouvelle barge $t+1$ sans fermer l'ancienne

$B = B \cup \{t+1\}$

 Charger c dans la barge $t+1$ selon le type de traitement RP (**)

fin si

fin pour

5.3.2.5 Diversification de la population initiale

Afin d'introduire encore plus de diversité dans les individus de la population initiale, nous utilisons une autre procédure de randomisation supplémentaire. Puisque nous avons développé deux heuristiques de chargement de conteneurs dans un convoi de barges, HNF_MultiBarge et HFF_MultiBarge, nous en choisissons une au hasard à chaque génération d'un individu de la population. En raison de toutes les règles de randomisation utilisées, nous pouvons générer une population initiale très diversifiée de taille N .

5.3.3 Classification des individus

5.3.3.1 La notion de non-dominance

Avant de procéder à la sélection des individus pour l'étape de reproduction, nous rappelons la notion de non-dominance. Dans le NSGA-II, deux solutions sont comparées en fonction de la dominance de l'une sur l'autre (Deb et Goel, 2001). Cette dernière est décrite comme suit: on dit qu'une solution x domine une autre solution y , si les deux conditions (1) et (2) sont vraies :

- 1) La solution x n'est pas pire que y pour tous les objectifs
- 2) La solution x est strictement meilleure que y pour au moins un objectif

Si x ne domine pas y et vice versa, on dit que les deux solutions ne sont pas dominées ou non comparables, encore dites, des solutions de compromis.

5.3.3.2 Le mécanisme de sélection Pareto « *ranking* »

Une sélection Pareto est utilisée selon la relation de dominance pour affecter des rangs « *rank* » aux individus de la population, faisant apparaître la notion de front. Suivant cette technique de classification « *ranking* » (Goldberg, 1989 ; Srinivas et Deb, 1994), nous attribuons un numéro unique à chaque front et le représentons par $Front_k$, où k ($k \geq 1$) est le numéro du front. D'abord, les individus non dominés de toute la population P auront un rang égal à 1, constituant $Front_1$. Puis, les individus non dominés de la population P , privée de ceux appartenant au $Front_1$, auront un rang égal à 2 constituant ainsi $Front_2$. Ce processus est répété jusqu'à la classification complète de P . La **Figure IV-13** présente un exemple de classification d'une population en fronts de Pareto.

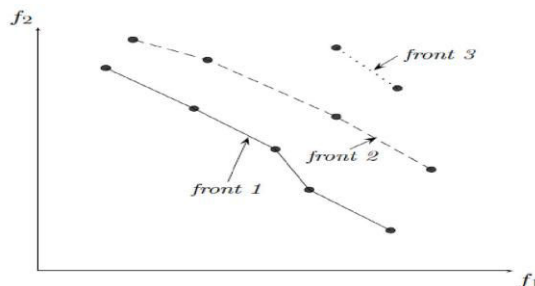


Figure IV-13. Illustration des fronts de Pareto dans un problème de minimisation bi-objectifs

Les différentes étapes de cette procédure de classification, connue sous le nom tri non-dominé rapide (*Fast Non-dominated Sorting*, NS), sont représentées dans l'algorithme 4 qui est introduit par Deb et al. (2002).

Algorithm 4: Génération des fronts de Pareto selon l'approche de tri non-dominé rapide (*Fast Non-dominated Sorting*, NS).

```

 $F_i = \emptyset$ 
  pour chaque solution  $p \in P$  faire
     $S_p = \emptyset$ 
     $n_p = 0$ 
    pour chaque  $q \in P$  faire
      si  $p$  domine  $q$  faire
         $S_p = S_p \cup \{q\}$ 
      sinon
        si  $q$  domine  $p$  faire
           $n_p = n_p + 1$ 
        fin si
      fin si
    fin pour
    si  $n_p = 0$  faire
       $p_{rang} = 1$ 
       $F_1 = F_1 \cup \{p\}$ 
    fin si
  fin pour
   $i = 1$ 
  tant que  $F_i \neq \emptyset$  faire
     $Q = \emptyset$ 
    pour chaque solution  $p \in F_i$  faire
      pour chaque solution  $q \in S_p$  faire
         $n_q = n_q + 1$ 
        si  $n_q = 0$  faire
           $q_{rang} = i + 1$ 
           $Q = Q \cup \{q\}$ 
        fin si
      fin pour
    fin pour
     $i = i + 1$ 
     $F_i = Q$ 
  fin tant que

```

5.3.3.3 Préservation de la diversité

NSGA II utilise aussi une seconde mesure visant à améliorer la diversité de la population dans l'espace des objectifs, appelée affectation des distances d'encombrement ou encore « *crowding distance assignment* » (De Jong, 1975 ; Blicke et Thiel, 1996). Comme le montre la **Figure IV-14**, la distance d'encombrement d'un individu i se calcule en fonction du cuboïde ayant comme sommets les deux individus adjacents de i sur chaque objectif. La stratégie de calcul de la distance *crowding* est détaillée dans l'algorithme 5 (Deb et al., 2001).

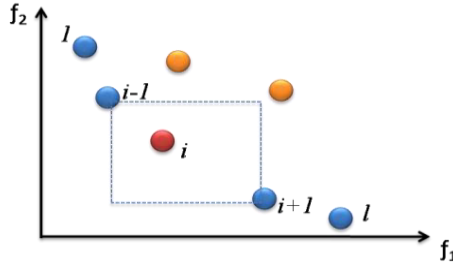


Figure IV-14. Calcul de distance d'encombrement (*crowding*)

Algorithm 5: Stratégie de distance d'encombrement (*crowding distance*).

$F \leftarrow$ front de Pareto

$l = |F|$

pour chaque individu $i \in F$ **faire**

$F[i]_{dis} = 0$

fin pour

pour chaque fonction objectif m **faire**

Trier les individus suivant l'objectif m par ordre croissant de magnitude

$F[1]_{dist} = \infty$

$F[l]_{dist} = \infty$

pour chaque solution i allant de 2 à $l - 1$ **faire**

$F[i]_{dist} = F[i]_{dist} + \frac{F[i+1]_m - F[i-1]_m}{f_m^{max} - f_m^{min}} (*)$

fin pour

fin pour

(*): $F[i]_m$ représente la valeur de la $m^{ième}$ fonction objectif relative au $i^{ième}$ individu de F
 f_m^{max}, f_m^{min} représentent, respectivement, la valeur maximale et minimale de l'objectif m

NSGA-II écarte les individus les plus encombrés dans l'espace des objectifs. Autrement dit, on préfère les solutions ayant les plus grandes distances d'encombrement.

5.3.4 Opérateur de sélection

L'opérateur de sélection a pour mission de choisir dans la population présente les futurs parents nécessaires à l'étape de reproduction. Il est utilisé aussi pour la reconstruction de la population dans l'étape de remplacement. Dans le 3D-CCSPP, nous choisissons la sélection par tournoi binaire qui consiste à choisir aléatoirement deux d'individus dans la population et de sélectionner l'individu qui est de meilleure qualité selon l'opérateur de comparaison encombré « *crowded comparison operator* ».

5.3.4.1 Opérateur de comparaison encombré

L'opérateur de comparaison encombré ($<_n$) permet de guider le processus de la sélection avec la répartition uniforme des solutions de Pareto. Chaque individu i de la population est caractérisé par:

- ▶ Le rang de non domination i_{rang}
- ▶ La distance de *crowding* i_{dist}

Soit les deux individus i et j , nous définissons l'ordre partiel $<_n$ comme suit:

$$i <_n j \text{ si } (i_{rang} < j_{rang}) \text{ ou } i_{rang} = j_{rang} \text{ et } i_{dist} > j_{dist}$$

En d'autres termes, on préfère la solution appartenant au front de Pareto d'ordre le plus petit si i et j appartenant à deux fronts de Pareto. Sinon, on choisit la solution qui a la distance d'encombrement la plus grande.

5.3.5 Etape de reproduction

5.3.5.1 Opérateur de croisement

L'opérateur de croisement est couramment utilisé dans les algorithmes évolutifs pour la production de solutions descendantes à partir de solutions parents. Généralement, deux solutions parents sont sélectionnées pour générer deux solutions descendantes après que les parents soient sélectionnés selon la probabilité de croisement P_c .

Dans le 3D-CCSPP, les plans de chargement des conteneurs dans le convoi résultants sont associés à plusieurs difficultés et peuvent être irréalisables si l'opérateur de croisement n'est pas correctement conçu. Il est donc impossible d'appliquer directement des opérateurs de croisement classiques. Dans ce qui suit, nous concevons un nouvel opérateur de croisement spécifique au 3D-CCSPP pour faire face à ces difficultés, à savoir, les contraintes de chargement imposées par la taille des conteneurs (20' et 40') et la fragilité dynamique de chaque pile. Le croisement proposé procède en deux phases :

- ▶ La première phase consiste à choisir d'une façon aléatoire deux lignes non vides à croiser (l_i et l_j) qui peuvent avoir des indices différents et appartenant à deux barges différentes, chacune appartenant à un parent, puis les échanger créant ainsi deux individus enfants.
- ▶ La faisabilité de ces nouveaux individus doit être vérifiée lors de la deuxième phase en utilisant l'heuristique (HVF) que nous expliquerons par la suite.

Dans la **Figure IV-15**, nous illustrons le fonctionnement du croisement proposé. Dans cet exemple, les lignes à échanger entre les individus parents P1 et P2 sont l_1 (ligne 2 de la barge 1 de P1) et l_2 (ligne 2 de la barge 2 de P2) représentées par deux couleurs différentes. Nous remarquons que les deux individus enfants E1 et E2 sont non réalisables juste en termes de répétition de quelques conteneurs et l'absence d'autres, et non pas au niveau des contraintes de taille des conteneurs et de fragilité des piles.

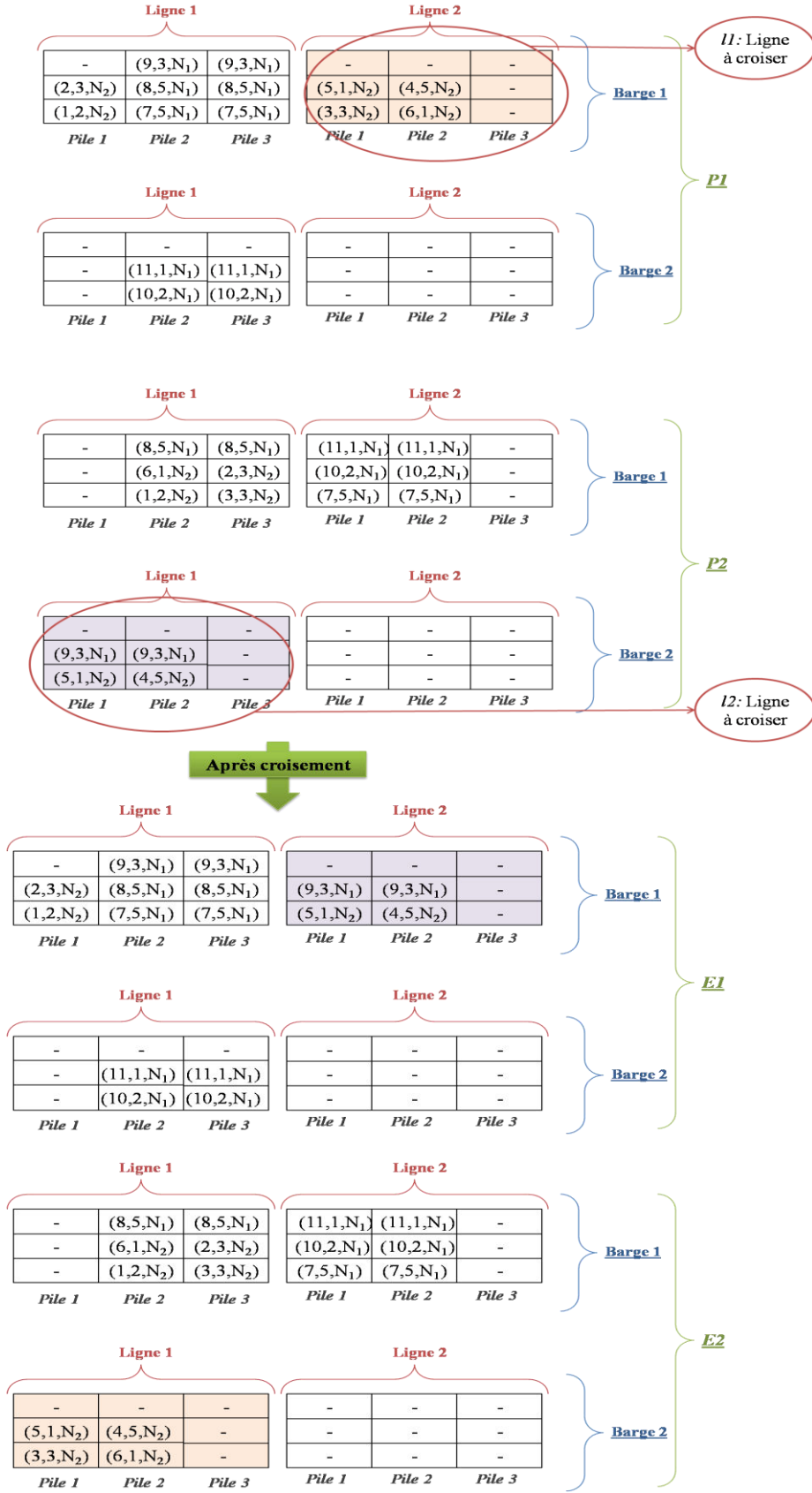


Figure IV-15. Fonctionnement de l'opérateur croisement proposé pour le 3D-CCSPP

5.3.5.2 Heuristique de la vérification de faisabilité des individus (HVF)

L'opération de croisement peut aboutir à des solutions ou des chromosomes non réalisables. Il est donc nécessaire de les vérifier et de les corriger. Les règles de vérification fondamentales sont les suivantes :

- ▶ Absence ou répétition des conteneurs
- ▶ Contraintes de taille des conteneurs
- ▶ Contraintes de fragilité des piles

L'opérateur de croisement proposé dans ce travail permet de diminuer les trois règles de vérification en une seule, puisque la notion de non faisabilité peut exister seulement au niveau des contraintes d'absence ou de répétition des conteneurs dans le convoi. De ce fait, nous proposons l'heuristique de vérification de la faisabilité « HVF » qui permet de recharger respectivement la ligne l_i et l_j , d'une façon aléatoire, tout en considérant les conteneurs non encore chargés et ceux déjà chargés. L'algorithme 7 décrit les étapes à suivre.

Algorithm 6: Heuristique de Vérification de la Faisabilité (HVF).

Entrées : *correction de l'individu E1*

$l1$: la ligne choisie du parent P1

$l2$: la ligne choisie du parent P2

E1 : l'enfant du parent P1 et la ligne $l2$

E2 : l'enfant du parent P2 et la ligne $l1$

C : l'ensemble des conteneurs appartenant à $l1 \cup l2$

pour tout n appartenant à C **faire**

si il existe un conteneur en commun dans $l1$ et $l2$ **break**

sinon

si il existe un conteneur dans la ligne $l2$ mais n'existe pas dans $l1$

 Supprimer n des autres lignes non croisées du E1

 Corriger les vides générés

 Mettre à jour la fragilité et la hauteur des piles

sinon

 Recharger n dans la première position valable dans E1 selon les règles RP1 ou RP2

fin si

fin si

fin pour

Nous notons que la suppression des conteneurs, lors de correction, se fait au niveau des autres lignes de l'individu enfant à part la ligne croisée, sinon ce dernier sera identique à son individu parent. Nous illustrons le fonctionnement de la stratégie de correction proposée dans les figures, **Figure IV-16** et **Figure IV-17** en utilisant les deux individus enfants obtenus dans la **Figure IV-15**, respectivement. Dans cet exemple, l'individu E1 est non réalisable en termes de la répétition du conteneur 9 et en termes d'absences des conteneurs 3 et 6. La couleur rouge représente les conteneurs à garder, la couleur orange indique les

conteneurs à supprimer des autres lignes du convoi de E1 à part la ligne croisée (l1), et la couleur verte représente les conteneurs à recharger dans le convoi. De même, nous illustrons la correction de l'individu E2 qui est, à son tour, non réalisable en termes de répétition des conteneurs 3 et 6 et en termes d'absences du conteneur 9. Dans l'étape de suppression des conteneurs répétés de taille 20', il faut toujours vérifier la taille des conteneurs qui se trouve au-dessus pour éviter la génération des espaces vides entre les niveaux de la pile. L'algorithme 7 propose une procédure de vérification pour supprimer un conteneur 20'.

Algorithm 7: Heuristique de Suppression d'un conteneur Vingt pieds (HSV).

```
c ← le conteneur à supprimer de taille 20'  
(i, j, h) ← la position du conteneur c  
Pour k allant de h + 1 à Niv  
    T ← le premier conteneur de taille 40'  
    Si T ≠ NULL  
        Si T occupe également la position (i, j + 1, k) faire  
            Supprimer le conteneur (c') de taille 20' qui se trouve dans la position (i, j + 1, h)  
            Ajouter c' dans la liste des conteneurs à recharger dans l'étape de correction selon HVF  
            break  
        Sinon  
            Supprimer le conteneur (c') de taille 20' qui se trouve dans la position (i, j - 1, h)  
            Ajouter c' dans la liste des conteneurs à recharger dans l'étape de correction selon HVF  
            break  
        fin si  
    fin si  
fin pour
```

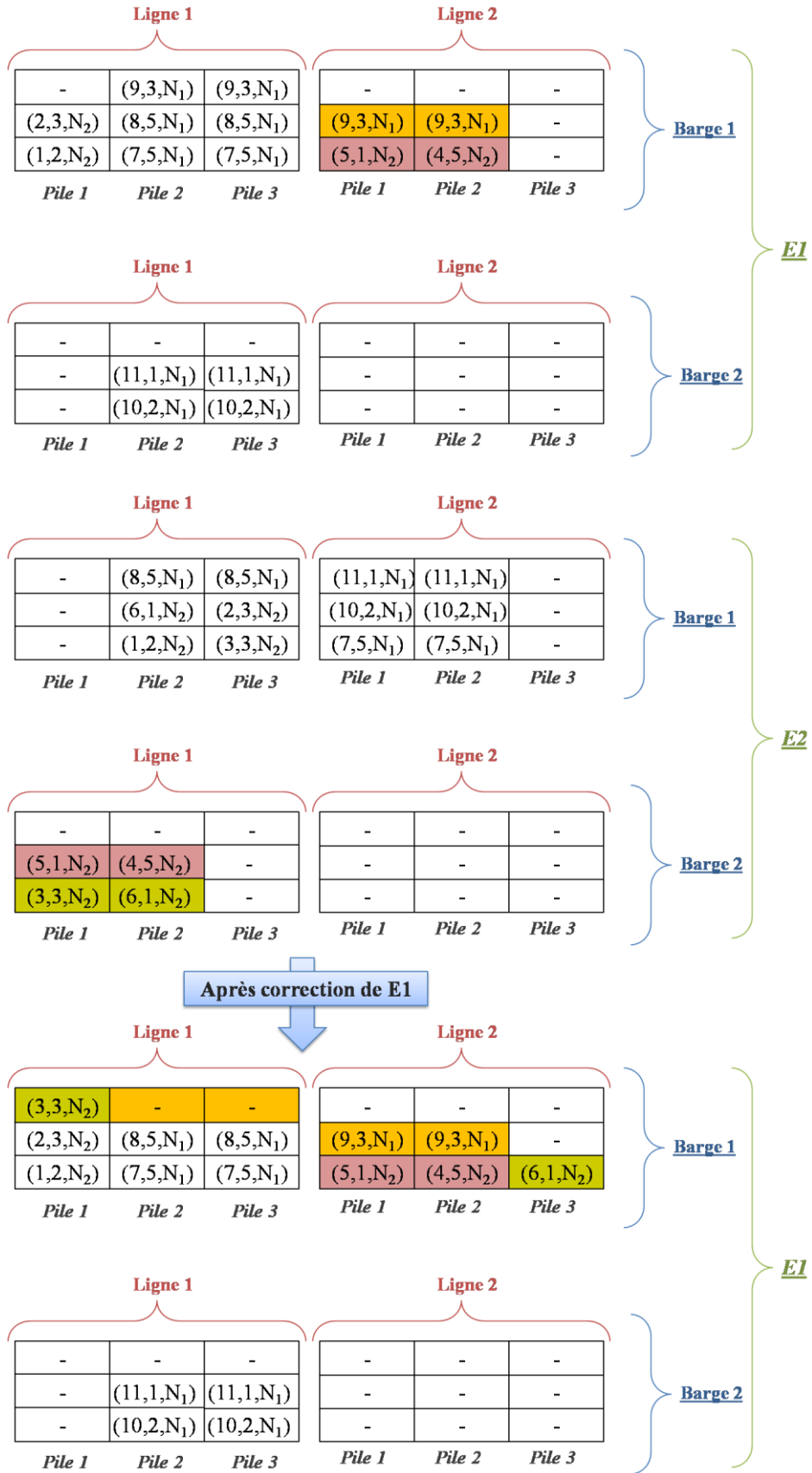


Figure IV-16. Fonctionnement de la stratégie de correction pour l'individu enfant E1

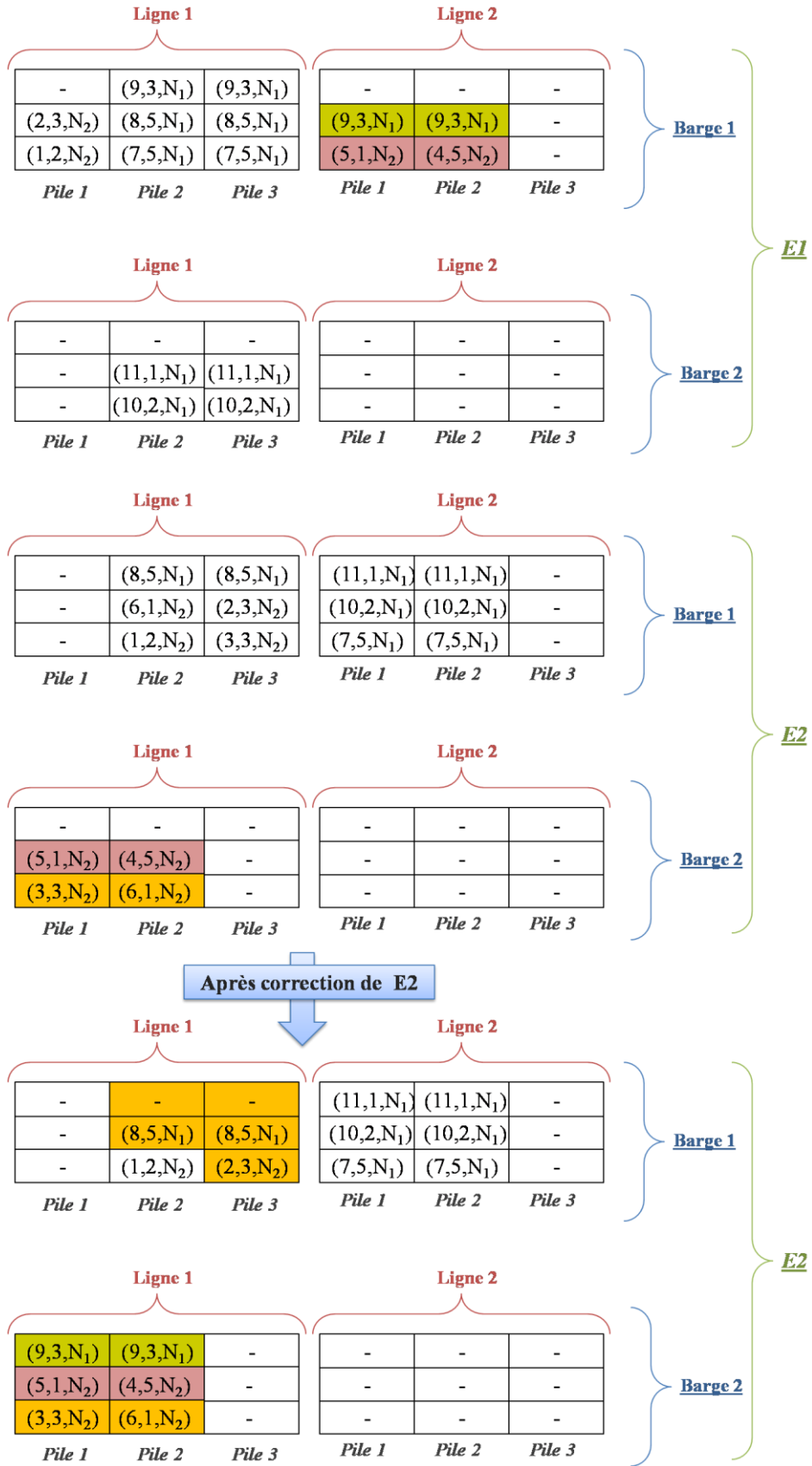


Figure IV-17. Fonctionnement de la stratégie de correction pour l'individu enfant E2

5.3.5.3 Opérateur de mutation

Une reproduction utilisant uniquement l'opérateur de croisement peut être piégée dans des optimums locaux. Les gènes des enfants sont limités par les gènes des parents, et si un gène n'est pas présent dans la population initiale (ou s'il disparaît à cause des reproductions), il ne pourra jamais se développer chez les descendants. L'opérateur de mutation a donc pour rôle d'introduire la diversité des solutions pour sortir des optima locaux. Elle consiste à modifier un ou plusieurs gènes d'un individu sélectionné par l'opérateur de sélection. Dans le 3D-CCSPP, nous proposons deux stratégies de mutation:

a. Mutation M1

- Choisir deux barges du convoi aléatoirement
- Permuter entre une ligne aléatoire de la première barge avec une autre ligne aléatoire de la deuxième barge

b. Mutation M2

- Choisir un ensemble B de barges réellement utilisées dans le convoi d'une façon aléatoire
- Pour chaque barge $t \in B$, permuter deux lignes choisies aléatoirement à partir de la même barge

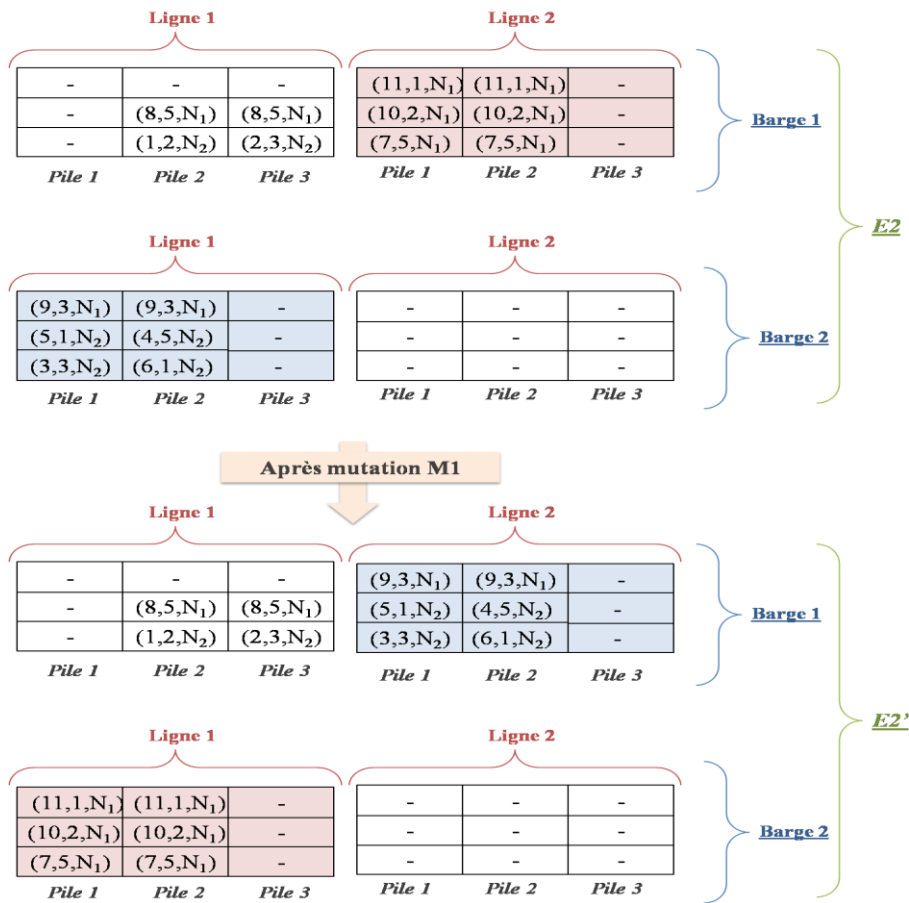


Figure IV-18. Fonctionnement de l'opérateur de mutation M1

Dans le NSGA-II, ces deux opérateurs de mutations sont choisis d'une façon aléatoire et appliqués aux solutions enfants avec la probabilité de mutation P_m .

5.3.6 Etape de remplacement

Les opérateurs de comparaison décrits précédemment ne servent pas uniquement lors des opérations de croisement et de mutation, mais interviennent aussi dans la sélection des individus à inclure dans la génération suivante (la population contenant les élites) à partir de la population globale (parent et enfant). L'itération de base de NSGA-II est décrite par la **Figure IV-19**.

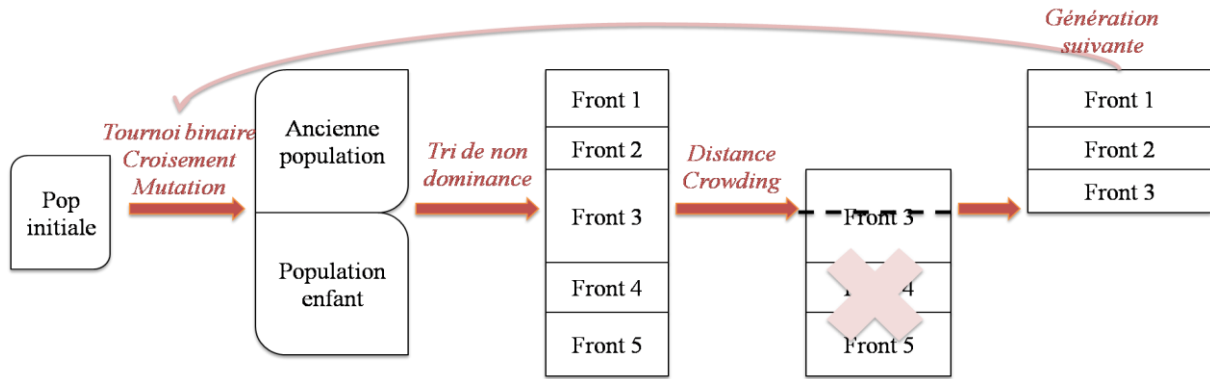


Figure IV-19. La procédure schématique du NSGA-II en une itération

Dans chaque génération du NSGA-II, des opérateurs génétiques sont appliqués à la population parents pour obtenir une population enfant de taille égale. Les parents et les enfants sont ensuite combinés pour former une population temporaire sur laquelle l'approche de tri non-dominé rapide (*Fast Non-dominated Sorting*, NS) est appliquée pour construire la population des parents de la génération suivante de même taille que la population initiale.

5.3.7 Critère d'arrêt

Compte tenu du fait que, dans le 3D-CCSPP, l'optimum de chaque fonction objectif étudiée n'est pas connu, le critère d'arrêt utilisé dans ce chapitre est basé sur le nombre de générations maximal. Après une analyse préliminaire du comportement du NSGA-II, le nombre maximal d'itérations a été fixé à 500.

5.4 Résultats numériques et expérimentaux

Pour tester la qualité et la performance du NSGA-II et du modèle proposé, nous considérons deux types d'instances. Le premier type est constitué d'un ensemble d'instances aléatoires de petite taille pouvant être résolues exactement par le logiciel d'optimisation CPLEX. Les instances de petite taille sont considérées dans ce chapitre pour comparer les résultats obtenus par la méthode NSGA-II avec ceux fournis par CPLEX. Le second type est basé sur une combinaison d'instances réelles des flux fluviaux du projet HAROPA proposées par le Grand Port Maritime du Havre (GPMH) avec un ensemble d'instances de moyenne et grande taille générées aléatoirement, présentant ainsi différentes situations rencontrées par

les services de navigation intérieure. Nous notons que le problème de faisabilité des instances n'est pas posé dans le cas du 3D-CCSPP.

5.4.1 Environnement expérimental et réglage des paramètres

Toutes les approches utilisées pour la résolution du 3D-CCSPP ont été implémentées avec Java en utilisant NetBeans8.0. Par défaut, toutes les expériences ont été menées sur un Intel® Core™ i5-4570 CPU @ 3.20 GHz, RAM 4 Go. Le réglage final des paramètres de l'algorithme proposé est indiqué dans le **Tableau IV-1**. Les valeurs présentées dans ce tableau sont le résultat de plusieurs études intensives que nous avons menées pour affiner le NSGA-II. Nous notons que la taille de la population est fixée alternativement à 50, 100 et 150, ce qui est une taille appropriée pour que les décideurs puissent faire des échanges des solutions au sein de la population. Dans nos expériences préliminaires, nous avons essayé d'accorder différentes combinaisons de P_c (probabilité de croisement) et P_m (probabilité de mutation) sur un ensemble d'instances, tout en conservant les autres paramètres. Pour chaque instance et chaque combinaison, 20 analyses indépendantes ont été effectuées. Nous pouvons confirmer que les effets de la probabilité de mutation montrent qu'un petit P_m est susceptible d'améliorer les valeurs des solutions obtenues lorsque P_c est grand. Par conséquent, nous définissons $P_c = 0.9$ et $P_m = 0.05$ en tant que paramètres définitifs.

Tableau IV-1. Réglage des paramètres de l'algorithme proposé

<i>Paramètre</i>	<i>Valeur</i>
Taille de la population	$\in \{50,100,150\}$
Probabilité de croisement (P_c)	0.9
Probabilité de mutation (P_m)	0.05
Nombre maximal d'itérations (tmax)	500

5.4.2 Génération des instances du 3D-CCSPP

Suivant la même structure considérée dans le chapitre III, nous générons un ensemble d'instances de petite, moyenne et grande taille. Dans un premier lieu, nous décrivons les caractéristiques relatives à tous les convois de barges utilisés dans ce chapitre pour générer les instances finales du 3D-CCSPP. Une série de 12 instances, correspondante à des convois de tailles réelles transportant des capacités allant de 32 à 1386 EVP, a été créée. Le **Tableau IV-2** décrit les caractéristiques de chaque convoi : l'identifiant « Id_Convoi », le nombre de barges maximal dans le convoi « Nbre_max », le nombre de lignes, le nombre de baies, le nombre de niveaux et la capacité maximale de chaque barge sont indiqués respectivement dans les colonnes « l », « L », « Niv » et « B_CapM ». Enfin, la capacité maximale de chaque convoi est donnée dans la dernière colonne « C_CapM ». Nous rappelons que $B_{CapM} = l * L * Niv$, et $C_{CapM} = Nbre_{max} * l * L * Niv$.

Tableau IV-2. Caractéristiques relatives aux convois de barges

<i>Id_Convoi</i>	<i>Nbre_max</i>	<i>Dim_Barge</i>			<i>B_CapM</i>	<i>C_CapM</i>
		<i>I</i>	<i>L</i>	<i>Niv</i>		
Cv1	2	2	4	2	16	32
Cv2	2	2	3	4	24	48
Cv3	2	2	5	3	30	60
Cv4	2	2	4	4	32	64
Cv5	2	3	4	3	36	72
Cv6	2	3	6	3	54	108
Cv7	6	5	6	3	90	540
Cv8	5	4	7	4	112	560
Cv9	5	5	8	4	160	800
Cv10	5	5	10	4	200	1000
Cv11	3	6	12	5	360	1080
Cv12	3	7	11	6	462	1386

Dans un deuxième lieu, nous rapportons, dans le **Tableau IV-4**, les paramètres relatifs à l'ensemble de conteneurs considérés dans ce chapitre. Pour chaque instance « InstC » indiquée dans la première colonne, un nombre $|N|$ de conteneurs de taille mixtes doit être transporté ($|N| = |N1| + |N2|$, où $|N1|$ indique le nombre de conteneurs 40 pieds et $|N2|$ le nombre des 20 pieds). $|N|$, $|N1|$ et $|N2|$ sont affichés respectivement dans la deuxième, la troisième et la cinquième colonne. Tandis que, les colonnes 4 et 6 indiquent, respectivement, le poids « W^{TM} » des conteneurs 40' et 20', il est donné en tonne métrique et est généré aléatoirement dans le cadre des intervalles de poids « L », « M » et « H » (comme expliqué dans le **Tableau IV-3**). Le nombre total d'EVP dans chaque instance est rapporté dans la dernière colonne « N_TEU ». Comme dans le chapitre III, nous estimons que la fragilité de chaque conteneur est égale à 3 fois son poids.

Tableau IV-3. Les intervalles de poids des conteneurs

<i>Taille</i>	<i>L</i>	<i>M</i>	<i>H</i>
20'	[3-15]	[16-24]	[25-29]
40'	[4-15]	[16-24]	[25-31]

L : le poids léger M : le poids moyen H : le poids lourd

Tableau IV-4. Informations relatives aux conteneurs

<i>InstC</i>	$ N $	<i>N1</i>		<i>N2</i>		<i>N_TEU</i>
		$ N1 $	W^{TM}	$ N2 $	W^{TM}	
InstC1	7	4	L/M	3	H	11
InstC2	10	2	L/H	8	M	12
InstC3	14	10	M/H	4	L	24
InstC4	15	3	H	12	L/M	18
InstC5	16	8	M	8	L/H	24
InstC6	20	15	L	5	M/H	35
InstC7	25	11	L/M	14	H	36
InstC8	27	7	L/H	20	M	34
InstC9	32	20	L	12	M/H	52
InstC10	70	34	L/M/H	36	L/M/H	104
InstC11	100	75	L/M/H	25	L/M/H	175
InstC12	200	150	L/M/H	50	L/M/H	350
InstC13	300	160	L/M/H	140	L/M/H	460
InstC14	400	211	L/M/H	189	L/M/H	611
InstC15	500	261	L/M/H	239	L/M/H	761

Finalement, les instances du 3D-CCSPP ont été créées en combinant le **Tableau IV-2** et le **Tableau IV-4** dans le **Tableau IV-5**. Pour chaque instance « InstCCSP », un convoi de barges « Id_Convoi » doit être amarré dans le port initial pour arrimer l'ensemble « InstC » de conteneurs et visiter « P » ports lors de la tournée. Nous supposons que la ligne fluviale du convoi contient au maximum 35 ports et que le nombre « P » de ports à visiter dans une tournée, indiqué dans la dernière colonne, varie de 5 à 35, avec $P \in \{5, 10, 15, 25, 35\}$.

Tableau IV-5. Génération des instances du 3D-CCSPP

<i>InstCCSP</i>	<i>Id_Convoi</i>	<i>InstC</i>	<i>P</i>	<i>InstCCSP</i>	<i>Id_Convoi</i>	<i>InstC</i>	<i>P</i>
CCSP1	Cv1	InstC1	5	CCSP37	Cv9	InstC10	15
CCSP2		InstC2	5	CCSP38			25
CCSP3		InstC3	5	CCSP39			35
CCSP4	Cv2	InstC1	5	CCSP40	Cv10	InstC11	15
CCSP5		InstC2	5	CCSP41			25
CCSP6	Cv3	InstC3	5	CCSP42			35
CCSP7			10	CCSP43	15		
CCSP8	Cv4	InstC4	5	CCSP44	InstC12	25	
CCSP9			10	CCSP45		35	
CCSP10		InstC5	5	CCSP46	15		
CCSP11	10		CCSP47	InstC13	25		
CCSP12	InstC6	5	CCSP48		35		
CCSP13		10	CCSP49	15			
CCSP14	Cv5	InstC7	5	CCSP50	InstC14	25	
CCSP15			10	CCSP51		35	
CCSP16		InstC8	5	CCSP52	15		
CCSP17	10		CCSP53	InstC15	25		
CCSP18	Cv6	InstC9	15		CCSP54	35	
CCSP19		Cv7	InstC10	15	CCSP55	15	
CCSP20	25			CCSP56	InstC13	25	
CCSP21	35			CCSP57		35	
CCSP22	Cv8	InstC11	15	CCSP58	Cv11	InstC14	15
CCSP23			25	CCSP59			25
CCSP24			35	CCSP60			35
CCSP25	InstC12	15	CCSP61	15			
CCSP26		25	CCSP62	InstC15	25		
CCSP27		35	CCSP63		35		
CCSP28	Cv9	InstC10	15	CCSP64	Cv12	InstC13	15
CCSP29			25	CCSP65			25
CCSP30			35	CCSP66			35
CCSP31	Cv10	InstC11	15	CCSP67	Cv13	InstC14	15
CCSP32			25	CCSP68			25
CCSP33			35	CCSP69			35
CCSP34	Cv11	InstC12	15	CCSP70	Cv14	InstC15	15
CCSP35			25	CCSP71			25
CCSP36			35	CCSP72			35

5.4.3 Résultats numériques et discussion

À notre connaissance, le 3D-CCSPP n'a pas été étudié dans la littérature. C'est pour cette raison que nous avons comparé les résultats obtenus par l'algorithme proposé (NSGA-II) avec ceux obtenus par le solveur CPLEX en utilisant de petites instances; lorsque CPLEX ne donne pas la solution optimale, nous avons considéré les meilleures bornes

trouvées si elles existent. Sinon, et avec l'objectif de mesurer la qualité de notre adaptation du NSGA-II au 3D-CCSPP sur de grandes instances, nous avons proposé un ensemble de méthodes analysant ainsi la performance de chaque fonction objectif.

5.4.3.1 Résolution du modèle mathématique par CPLEX

Nous avons modélisé le 3D-CCSPP avec trois objectifs : le premier correspond à la minimisation des frais de *shiftings* de toute la rotation, le deuxième représente la notion de stabilité du convoi de barges définie par la distance métacentrique de chaque barge, tandis que le troisième est lié à l'aspect du problème de *bin-packing* impliquant ainsi la minimisation de nombre de barges réellement utilisées dans le convoi. Pour valider le modèle proposé, nous résolvons un ensemble d'instances par le solveur CPLEX qui optimise un seul objectif à la fois en utilisant trois paramètres associés aux trois fonctions objectifs (α pour la première, β pour la deuxième et γ pour la troisième).

Le **Tableau IV-6** dresse les résultats obtenus pour l'ensemble d'instances de petites taille indiquées dans le **Tableau IV-5**, tels qu'obtenus par le solveur CPLEX. Une limite de temps d'une heure a été imposée au solveur. La première colonne de ce tableau « InstCCSP » indique l'identifiant de l'instance du 3D-CCSPP, la colonne « Shift » montre les résultats obtenus lorsque seul le nombre de *shiftings* est minimisé (avec $\alpha = 1$, $\beta = 0$ et $\gamma = 0$), la colonne « Stab » rapporte les résultats obtenus en maximisant uniquement la stabilité totale du convoi selon la distance métacentrique GM donnée en pieds (avec $\alpha = 0$, $\beta = 1$ et $\gamma = 0$) et la colonne « BNbre » indique les résultats obtenus en minimisant uniquement le nombre de barges réellement utilisées dans le convoi (avec $\alpha = 0$, $\beta = 0$ et $\gamma = 1$). Les valeurs sous l'en-tête « Val Obj » représentent la meilleure valeur objectif correspondante obtenue par CPLEX, les valeurs sous l'en-tête « Statut CPLEX » indiquent le statut du solveur à la fin du calcul, tandis que les valeurs sous l'en-tête « Statut temps (s) » représentent la durée totale d'exécution nécessaire pour atteindre les statuts correspondants.

D'après le **Tableau IV-6**, nous pouvons confirmer la validité de notre modèle mathématique proposé pour l'ensemble des instances de petite taille étudiées. En effet, les valeurs obtenues en minimisant chacune des deux fonctions objectifs « shift » et « BNbre » ont atteint la borne inférieure théorique qui est définie, respectivement, par zero *shifting* et le nombre de conteneurs en EVP (N_TEU) divisé par la capacité maximale de la barge du convoi correspondant (B_CapM), et dans le cas de la maximisation de la fonction objectif « Stab », toutes les valeurs obtenues sont supérieures à la plus petite distance métacentrique souhaitée en pratique (GM = 1 mètre = 3.28084 pieds). Nous observons que CPLEX a permis d'optimiser la majorité des cas des petites instances étudiées en quelques secondes pour les trois fonctions objectifs. Cependant, dans les autres cas, CPLEX est moins compétitif, car le délai de calcul était atteint sans que le solveur puisse reconnaître la solution actuelle comme étant optimale (indiquée par « Actuelle » dans « Statut CPLEX »), ou bien, il n'a pas réussi à trouver une solution réalisable en raison des ressources mémoire insuffisantes de la machine (indiquée par « Mém. Insuff. » dans « Statut CPLEX »).

Tableau IV-6. Résultats numériques des instances de petite taille du **Tableau IV-5** et leur statut correspondant, en optimisant le nombre de *shiftings*, la stabilité en GM et le nombre de barges du convoi, où « $\alpha=1, \beta=0$ et $\gamma=0$ », « $\alpha=0, \beta=1$ et $\gamma=0$ » et « $\alpha=0, \beta=0$ et $\gamma=1$ » respectivement

<i>InstCCSP</i>	<i>Shift</i>			<i>Stab</i>			<i>BNbre</i>		
	<i>Val Obj</i>	<i>Statut CPLEX</i>	<i>Statut temps (s)</i>	<i>Val Obj</i>	<i>Statut CPLEX</i>	<i>Statut temps (s)</i>	<i>Val Obj</i>	<i>Statut CPLEX</i>	<i>Statut temps (s)</i>
CCSP1	0*	P.O.	0.22	6*	P.O.	0.27	1*	P.O.	0.25
CCSP2	0*	P.O.	1.02	10.78*	P.O.	3.16	1*	P.O.	3.64
CCSP3	0*	P.O.	2.17	7.68*	P.O.	2.28	2*	P.O.	4.72
CCSP4	0*	P.O.	3.61	13.89*	P.O.	14.23	1*	P.O.	13.3
CCSP5	0*	P.O.	15.11	15*	P.O.	24.02	1*	P.O.	25.73
CCSP6	0*	P.O.	23.31	7.71*	P.O.	2.45	2	P.O.	2.42
CCSP7	0*	P.O.	16.39	12.84*	P.O.	16.94	1*	P.O.	28.03
CCSP8	0*	P.O.	47.44	13.62*	P.O.	38.55	1*	P.O.	65.74
CCSP9	0*	P.O.	210.66	13.36*	P.O.	43.52	1*	P.O.	69.94
CCSP10	0*	P.O.	93.83	8.72*	Actuelle	3600	1*	P.O.	108.77
CCSP11	0*	P.O.	108.89	9*	Actuelle	3600	1*	P.O.	28.33
CCSP12	0*	P.O.	67.45	10.01*	Actuelle	3600	2	Actuelle	3600
CCSP13	0*	P.O.	372.77	7.21*	Actuelle	3600	2	Actuelle	3600
CCSP14	0*	P.O.	166.97	14.76*	P.O.	145.83	2	Actuelle	3600
CCSP15	0*	P.O.	175.38	14.76*	P.O.	147.09	2	Actuelle	3600
CCSP16	0*	P.O.	175.61	14.97*	P.O.	161.41	2	Actuelle	3600
CCSP17	0*	P.O.	169.34	14.8*	P.O.	166.02	1*	P.O.	371.69
CCSP18	0*	P.O.	1342.98	-	Mém. Insuff.	-	-	Mém. Insuff.	-

P.O. : la valeur objective de la solution est optimale (Prouvée Optimale)

Actuelle : la meilleure valeur objective atteinte à la fin d'une heure de calcul sans que le solveur puisse la reconnaître comme étant optimale

Mém. Insuff. : ressources mémoire insuffisantes

* : atteindre la borne inférieure pour les colonnes « *Shift* » et « *BNbre* » et supérieure à GM souhaitée pour la colonne « *Stab* »

Pour illustrer la complexité du problème étudié, nous proposons de faire une étude sur les différentes caractéristiques des instances appliquées au modèle mathématique proposé et résolues par CPLEX dans le **Tableau IV-7**. Les deux premières colonnes sous l'entête « Instance » rapportent les différentes caractéristiques de chaque instance où « Id », « $|N|$ », « P », « B », « l », « L », et « Niv » représentent respectivement, l'identifiant de l'instance, le nombre de conteneurs à arrimer, le nombre de ports à visiter, le nombre de barge maximal dans le convoi, la largeur, la longueur et la hauteur de chaque barge. La troisième et la quatrième colonne indiquent, respectivement, le nombre de variables et de contraintes dans le modèle pour chaque instance. Le nombre d'itérations nécessaires pour trouver la solution en termes de nombre de *shifting*, de stabilité et de nombre de barges utilisées est donné, respectivement, dans les trois colonnes sous l'entête « Shift », « Stab » et « BNbre ». La dernière colonne représente la durée moyenne d'exécution nécessaire pour atteindre les solutions des trois fonctions objectifs.

Tableau IV-7. Caractéristiques des instances résolues par CPLEX

<i>Id</i>	<i>Instance</i>	<i>Nombre de variables</i>	<i>Nombre de contraintes</i>	<i>Nombre d'itérations</i>			<i>CPU (s)</i>
	<i>(/N/-P-B-I-L-Niv)</i>			<i>Shift</i>	<i>Stab</i>	<i>BNbre</i>	
CCSP1	(7-5-2-2-4-2)	3811	7190	8	51	101	0.25
CCSP2	(10-5-2-2-4-2)	7363	14140	48	1588	1555	2.67
CCSP3	(14-5-2-2-4-2)	13891	26820	68	158	189	2.35
CCSP4	(7-5-2-2-3-2)	10419	20310	61	33	467	3.47
CCSP5	(10-5-2-2-3-4)	20643	40600	505	688	318	14.21
CCSP6	(14-5-2-2-5-3)	37803	73906	107	138	394	24.35
CCSP7	(14-10-2-2-5-3)	37803	73906	127	138	394	15.17
CCSP8	(15-5-2-2-4-4)	60483	119374	1019	532	3576	50.57
CCSP9	(15-10-2-2-4-4)	60483	119374	43269	394	3576	108.04
CCSP10	(16-5-2-2-4-4)	68611	135360	179	370985	2541	1267*
CCSP11	(16-10-2-2-4-4)	68611	135360	24054	1126	2541	1450.66*
CCSP12	(20-5-2-3-4-3)	90723	178822	586	67955	22722	2422.48*
CCSP13	(20-10-2-3-4-3)	90723	178822	44461	81678	57820	2509.93*
CCSP14	(25-5-2-3-4-3)	140403	277208	1582	3253	198948	1306.8*
CCSP15	(25-10-2-3-4-3)	140403	277208	959	3253	136899	1320.34*
CCSP16	(27-5-2-3-4-3)	163299	322524	1309	3550	11486	1312.34*
CCSP17	(27-10-2-3-4-3)	163299	322524	948	3654	30999	707.34
CCSP18	(32-5-2-3-6-3)	342147	677242	5827	-	-	-

- : mémoire insuffisante

(*) interrompue par l'utilisateur

L'analyse du **Tableau IV-7** montre que le modèle mathématique appliqué à l'ensemble d'instances {CCSP1, ..., CCSP18} comporte un nombre de variables allant de 3 811 dans le cas de la plus petite instance considérée dans ce tableau (CCSP1) à 342 147 pour la plus grande (CCSP18), et un nombre de contraintes allant de 7 190 à 677 242 de façon correspondante, ce qui argumente le fait que le plan d'arrimage des conteneurs peut être généré par CPLEX sur des instances de petite taille. Cela revêt une importance particulière lorsque, comme indiqué précédemment, les approches de résolution exactes ne sont pas en mesure de fournir une solution réalisable en raison d'un manque de mémoire de la machine ou de la nécessité d'une grande quantité de temps de calcul, d'où l'utilisation des méta-heuristiques.

5.4.3.2 Résultats numériques obtenus par le NSGA-II

Dans cette section, nous rapportons les résultats numériques obtenus par la méthode NSGA-II. Nous commençons d'abord par les examiner du point de vue de la comparaison mono-objectif (CPLEX) afin d'évaluer et de mesurer leur qualité. La première colonne du **Tableau IV-8** contient l'identifiant des instances étudiées, les colonnes « Shift », « Stab » et « BNbre » représentent, respectivement, les résultats en fonction de nombre de *shiftings*, de stabilité totale GM (donnée en pieds) et de nombre de barges réellement utilisées dans le convoi. Dans chacune de ces trois colonnes nous présentons les résultats obtenus par le solveur CPLEX et la méthode NSGA-II. Les valeurs sous l'en-tête « Best sol » indiquent la meilleure solution trouvée par ces deux méthodes. Pour les solutions obtenues par CPLEX, la valeur de la fonction objectif correspondante indiquée dans la colonne « Val. Obj » représente la solution optimale, ou la meilleure borne trouvée dans les 3600 secondes. Pour les solutions trouvées par le NSGA-II, la valeur de la fonction objectif correspondante

donnée dans la colonne « Val. Obj » indique la moyenne des valeurs pour tout l'ensemble des individus du front de Pareto. Le CPU correspondant à l'exécution de chaque instance en secondes est rapporté dans les colonnes « CPU (s) ». Les valeurs sous l'en-tête « Dev » indiquent l'écart par rapport à la meilleure solution trouvée par le NSGA-II ou CPLEX, indiquée dans la colonne « Best sol » ($Dev = Best\ sol - Val\ Obj$).

Les résultats du **Tableau IV-8** confirment la performance du NSGA-II adapté au 3D-CCSPP. Ils montrent clairement l'efficacité de cette approche face aux instances de petite taille, offrant ainsi des solutions optimales. Pour les instances CCSP10-CCSP13 dans le cas de la maximisation de la stabilité et CCSP12-CCSP16 dans le cas de la minimisation de nombre de barges du convoi, CPLEX n'a pas été en mesure de fournir la solution optimale. Le résultat présenté pour ces instances représente la meilleure solution réalisable trouvée par CPLEX (indiqué par une étoile dans la colonne « Val Obj » correspondante). Pour ces sept instances, la méthode NSGA-II utilisée a trouvé une meilleure solution que la meilleure borne fournie par CPLEX dans un délai acceptable. Enfin, pour la dernière instance (CCSP18), CPLEX n'a pas été en mesure de fournir une solution réalisable, le NSGA-II donne ainsi la meilleure solution pour ce cas.

Les résultats de calcul obtenus sur les instances de grande taille par le NSGA-II sont résumés dans le **Tableau IV-9**. En pratique, il est toujours difficile d'afficher toutes les solutions optimales de Pareto en raison de son grand espace de solutions. Par conséquent, nous ne rapportons, par la suite, qu'un sous ensemble de solutions du front de Pareto choisies aléatoirement. La première colonne du tableau donne l'identifiant de chaque instance, les colonnes sous l'entête « Front Pareto » dressent l'ensemble des solutions choisies à partir du front de Pareto obtenu « S1 », « S2 », « S3 », « S4 » et « S5 ». Chacune de ces cinq colonnes rapporte les valeurs des solutions obtenues pour les trois fonctions objectifs étudiées « Shift », « Stab » et « BNbre », indiquant respectivement le nombre de *shiftings*, la stabilité GM (en pieds) et le nombre de barges utilisées. Nous pouvons remarquer que, pour chaque instance étudiée, les solutions obtenues surpassent d'autres solutions de la population dans au moins un objectif ; pour chaque solution appartenant au front de Pareto, il existe au moins une autre solution avec une pire valeur objective dans deux ou dans l'une des fonctions objectifs. À partir des alternatives proposées, le planificateur des plans de chargement et d'arrimage des conteneurs dans le convoi est chargé de choisir celle qui correspond le mieux à ses besoins, en fonction de son expérience personnelle.

Tableau IV-8. Résultats numériques obtenus par NSGA-II et CPLEX pour les instances de petite taille

<i>InstCCSP</i>	<i>Shift</i>							<i>Stab</i>							<i>BNbre</i>						
	<i>Best sol</i>	<i>CPLEX</i>			<i>NSGA-II</i>			<i>Best sol</i>	<i>CPLEX</i>			<i>NSGA-II</i>			<i>Best sol</i>	<i>CPLEX</i>			<i>NSGA-II</i>		
		<i>Val Obj</i>	<i>CPU (s)</i>	<i>Dev</i>	<i>Val Obj</i>	<i>CPU (s)</i>	<i>Dev</i>		<i>Val Obj</i>	<i>CPU (s)</i>	<i>Dev (en pieds)</i>	<i>Val Obj</i>	<i>CPU (s)</i>	<i>Dev (en pieds)</i>		<i>Val Obj</i>	<i>CPU (s)</i>	<i>Dev</i>	<i>Val Obj</i>	<i>CPU (s)</i>	<i>Dev</i>
CCSP1	0	0	0.22	0	0	7	0	6	6	0.27	0	6	7	0	1	1	0.25	0	1	7	0
CCSP2	0	0	1.02	0	0	7	0	10.78	10.78	3.16	0	10.43	7	0.35	1	1	3.64	0	1	7	0
CCSP3	0	0	2.17	0	0	5	0	7.68	7.68	2.28	0	7.68	5	0	2	2	4.72	0	2	5	0
CCSP4	0	0	3.61	0	0	7	0	13.89	13.89	14.23	0	12.99	7	0.9	1	1	13.3	0	2	7	1
CCSP5	0	0	15.11	0	0	11	0	15	15	24.02	0	12.14	11	2.86	1	1	25.73	0	2	11	1
CCSP6	0	0	23.31	0	0	12	0	7.71	7.71	2.45	0	7.65	12	0.06	2	2	2.42	0	2	12	0
CCSP7	0	0	16.39	0	0	12	0	12.84	12.84	16.94	0	10.12	12	2.72	1	1	28.03	0	1	12	0
CCSP8	0	0	47.44	0	0	12	0	13.62	13.62	38.55	0	11.54	12	2.08	1	1	65.74	0	2	12	1
CCSP9	0	0	210.66	0	0	12	0	13.36	13.36	43.52	0	12.03	12	1.33	1	1	69.94	0	2	12	1
CCSP10	0	0	93.83	0	1	12	1	11.34	8.72*	3600	2.62	11.34	12	0	1	1	108.77	0	2	12	1
CCSP11	0	0	108.89	0	0	13	0	10.64	9*	3600	1.64	10.64	13	0	1	1	28.33	0	2	13	1
CCSP12	0	0	67.45	0	0	13	0	11.15	10.01*	3600	1.14	11.15	13	0	1	2*	3600	1	1	13	0
CCSP13	0	0	372.77	0	0	13	0	9.74	7.21*	3600	2.53	9.74	13	0	2	2*	3600	0	2	13	0
CCSP14	0	0	166.97	0	0	13	0	14.76	14.76	145.83	0	10.86	13	3.9	2	2*	3600	0	2	13	0
CCSP15	0	0	175.38	0	0	13	0	14.76	14.76	147.09	0	10.79	13	3.97	2	2*	3600	0	2	13	0
CCSP16	0	0	175.61	0	0	14	0	14.97	14.97	161.41	0	10.91	14	4.06	2	2*	3600	0	2	14	0
CCSP17	0	0	169.34	0	0	14	0	14.8	14.8	166.02	0	11.13	14	3.67	1	1	371.69	0	2	14	1
CCSP18	0	0	1342.98	0	0	14	0	13.18	-	-	-	13.18	14	0	2	-	-	-	2	14	0

* interrompue par l'utilisateur

Tableau IV-9. Information des individus du front optimal de Pareto trouvé par le NSGA-II pour le cas des instances de grande taille

<i>Inst</i> <i>CCSP</i>	<i>Front Pareto</i>														
	<i>S1</i>			<i>S2</i>			<i>S3</i>			<i>S4</i>			<i>S5</i>		
	<i>Shift</i>	<i>Stab</i>	<i>BN bre</i>	<i>Shift</i>	<i>Stab</i>	<i>BN bre</i>	<i>Shift</i>	<i>Stab</i>	<i>BN bre</i>	<i>Shift</i>	<i>Stab</i>	<i>BN bre</i>	<i>Shift</i>	<i>Stab</i>	<i>BN bre</i>
CCSP19	15	8.87	2	12	8.44	2	13	8.54	2	10	7.78	2	5	7.36	2
CCSP20	9	8.4	2	10	8.74	2	11	8.85	2	5	7.71	2	7	8.05	2
CCSP21	12	8.26	2	14	8.7	2	7	7.06	2	5	7.04	2	9	7.66	2
CCSP22	14	10.05	3	15	10.24	3	13	9.95	3	10	9.56	3	8	9.33	3
CCSP23	19	8.52	2	21	8.64	2	17	8.22	2	15	8.06	2	11	8	2
CCSP24	16	8.51	2	21	8.61	2	13	8.27	2	15	8.37	2	24	8.78	3
CCSP25	31	14.76	4	39	14.94	4	45	15.11	4	38	14.92	4	30	14.74	4
CCSP26	47	13.74	4	56	13.78	4	36	13.57	4	55	13.75	4	37	13.72	4
CCSP27	40	12.17	4	52	12.26	4	39	12.16	4	36	12.13	4	44	12.23	4
CCSP28	7	8.23	2	14	9.08	2	12	8.93	2	11	8.81	2	8	8.61	2
CCSP29	16	10.9	2	22	11.38	2	25	11.49	2	11	9.61	2	15	10.5	2
CCSP30	18	10.05	2	15	10.02	2	22	10.16	2	13	9.51	2	8	9.38	2
CCSP31	32	10.25	2	23	9.73	2	26	9.93	2	20	9.51	2	19	9.46	2
CCSP32	39	12.04	2	34	11.68	2	31	11.5	2	28	11.19	2	22	10.99	2
CCSP33	39	9.76	2	30	9.59	2	28	9.5	2	22	8.82	2	24	9.29	2
CCSP34	72	15.51	4	74	15.55	4	65	15.38	4	57	15.28	4	53	15.07	4
CCSP35	51	13.88	4	52	13.91	4	48	13.82	4	66	13.95	4	57	13.94	4
CCSP36	45	15.91	4	42	15.9	4	53	16.25	4	40	15.7	4	46	16.14	4
CCSP37	14	10.55	2	21	11.14	2	19	10.79	2	25	11.29	2	18	10.6	2
CCSP38	16	5.48	1	18	5.62	1	15	5.44	1	22	5.72	1	23	6.29	1
CCSP39	24	6.24	1	19	5.78	1	21	6.19	1	20	5.85	1	17	5.31	1
CCSP40	34	11.29	2	31	11	2	22	10.89	2	28	10.97	2	18	10.16	2
CCSP41	44	12.3	2	40	12.14	2	30	11.72	2	36	11.95	2	21	11.16	2
CCSP42	22	9.51	2	18	9.07	2	16	8.98	2	19	9.43	2	17	3.99	2
CCSP43	48	11.16	3	45	11.08	3	51	11.26	3	54	11.32	3	40	10.99	3
CCSP44	83	11.27	3	61	10.88	3	74	11.2	3	72	10.9	3	51	10.55	3
CCSP45	54	10.93	3	59	11.33	3	55	11.31	3	65	11.35	3	48	10.6	3
CCSP46	139	11.34	3	116	11.19	3	131	10.95	3	107	10.16	3	98	10.12	3
CCSP47	131	11.13	3	130	11.08	3	122	10.77	3	126	10.96	3	101	9.11	3
CCSP48	95	10.55	3	102	10.61	3	109	10.97	3	105	10.7	3	107	10.85	3
CCSP49	191	13.92	4	163	13.4	4	170	13.62	4	165	13.52	4	193	13.95	4
CCSP50	157	12.34	4	160	12.53	4	187	12.64	4	177	12.6	4	168	12.58	4
CCSP51	153	18.91	4	176	19.01	4	158	18.95	4	147	18.72	4	149	18.83	4
CCSP52	216	18.85	5	180	18.59	5	178	18.56	5	189	18.75	5	202	18.79	5
CCSP53	177	19.81	5	181	19.94	5	168	19.74	5	190	20.08	5	164	19.72	5
CCSP54	213	16.8	5	236	17.05	5	221	16.96	5	216	16.89	5	212	16.74	5
CCSP55	119	9.85	2	120	9.97	2	91	9.49	2	105	9.84	2	92	9.83	2
CCSP56	122	10.09	2	111	9.98	2	112	10	2	106	9.62	2	108	9.88	2
CCSP57	155	11.14	2	130	10.6	2	145	10.92	2	139	10.87	2	137	10.79	2
CCSP58	156	14.56	3	142	14.22	3	148	14.52	3	166	14.6	3	168	14.81	3
CCSP59	225	14.5	3	205	13.97	3	218	14.01	3	223	14.06	3	222	14.05	3
CCSP60	182	14.91	3	177	14.73	3	185	14.94	3	169	14.72	3	187	14.98	3
CCSP61	266	14.72	3	260	14.51	3	265	14.56	3	252	14.4	3	259	14.41	3
CCSP62	274	14.59	3	236	14.35	3	231	14.26	3	244	14.39	3	257	14.48	3
CCSP63	226	14.49	3	215	14.4	3	217	14.44	3	224	14.47	3	221	14.46	3
CCSP64	126	11.25	2	108	11	2	116	11.22	2	127	11.27	2	111	11.13	2
CCSP65	122	12.18	2	119	12	2	117	11.99	2	100	10.22	2	92	9.16	2
CCSP66	139	11.29	2	125	11	2	126	11.23	2	116	10.6	2	121	10.7	2
CCSP67	154	16.56	3	148	16.5	3	141	16.28	3	144	16.3	3	120	12.66	3
CCSP68	163	15.16	3	160	15.07	3	161	15.13	3	168	15.2	3	159	15.02	3
CCSP69	182	13.9	3	191	14.26	3	187	14.22	3	183	14	3	204	14.28	3
CCSP70	221	16.68	3	225	16.72	3	222	16.7	3	230	16.81	3	218	16.54	3
CCSP71	285	9.01	3	260	8.9	3	293	9.03	3	257	8.85	3	246	8.84	3
CCSP72	244	14.5	3	239	14.44	3	242	14.45	3	230	14.36	3	228	14.21	3

5.4.3.3 Représentation graphique d'un ensemble de fronts de Pareto

Les quatre figures, **Figure IV-20**, **Figure IV-21**, **Figure IV-22** et **Figure IV-23** illustrent, respectivement, l'ensemble des solutions obtenues pour chacune des instances CCSP35, CCSP39, CCSP50 et CCSP64 sous la forme d'un front de Pareto en 3D. Chaque sous-figure donne une vue différente du front. Sur ces figures, chaque point (en rouge) représente une solution, c'est-à-dire, un plan d'arrimage répondant à toutes les contraintes établies par le modèle. L'axe des abscisses représente le nombre de *shiftings* durant tout le trajet du convoi, l'axe des ordonnées représente la stabilité totale du convoi selon la distance métacentrique GM et l'axe de profondeur indique le nombre de barges réellement utilisées pour transporter l'ensemble de conteneurs.

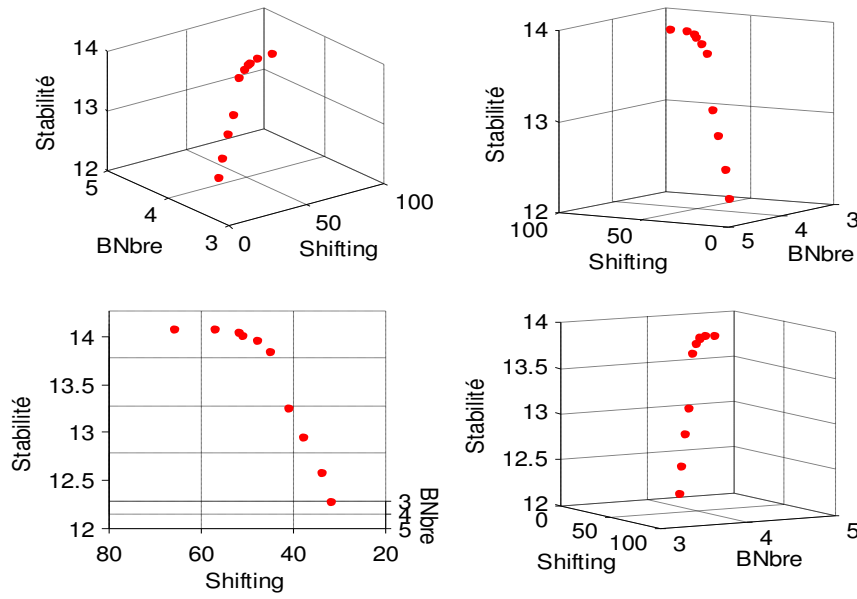


Figure IV-20. Front optimal de Pareto en 3D obtenu par le NSGA-II pour l'instance CCSP35

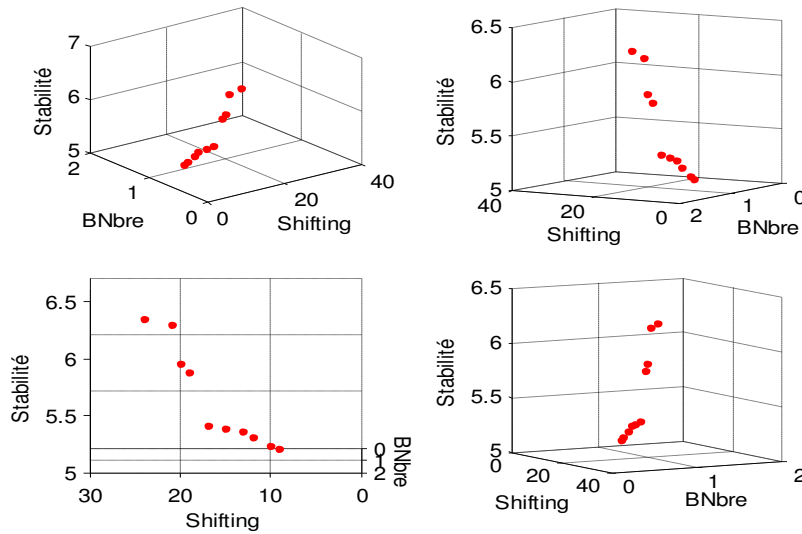


Figure IV-21. Front optimal de Pareto en 3D obtenu par le NSGA-II pour l'instance CCSP39

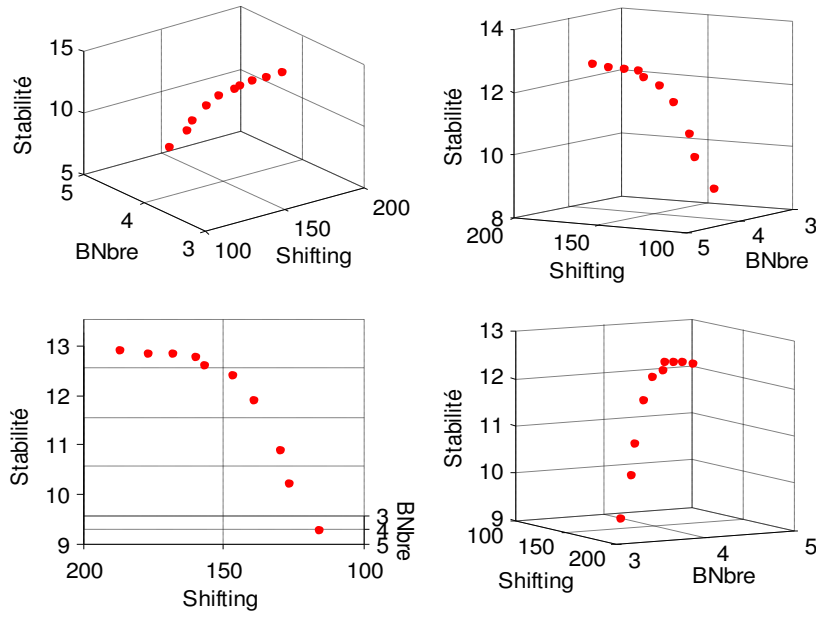


Figure IV-22. Front optimal de Pareto en 3D obtenu par le NSGA-II pour l'instance CCSP50

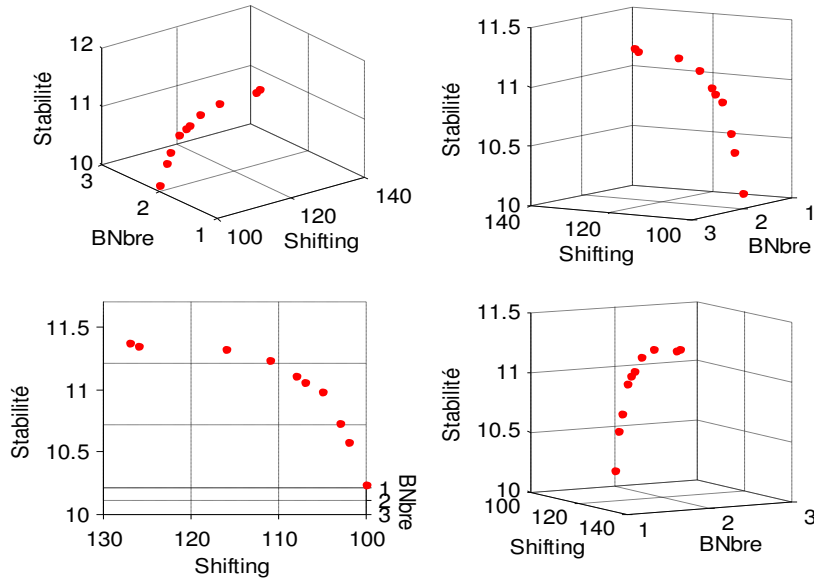


Figure IV-23. Front optimal de Pareto en 3D obtenu par le NSGA-II pour l'instance CCSP64

Pour l'ensemble des instances illustrées dans les quatre figures ci-dessus, nous remarquons que les individus du front de Pareto sont bien distribués tout au long du front, cela est dû à l'utilisation du principe de la distance *crowding* dans le NSGA-II. Nous constatons aussi que le nombre de barges utilisées dans chaque convoi reste le même d'un individu à un autre pour ces instances, ce qui génère une vue claire du front en 2D.

5.4.3.4 Analyse des résultats : comparaison et discussions

A. L'influence de la nature de la population initiale sur les résultats obtenus par le NSGA-II

Afin de montrer la façon dont la population finale du NSGA-II pourrait être affectée par la nature de la population initiale qui est générée, à l'origine, en utilisant un choix aléatoire entre les deux heuristiques de chargement de conteneurs proposées (HNF_MultiBarge et HFF_MultiBarge) lors de chaque exécution, comme indiqué auparavant, nous illustrons, respectivement, dans la **Figure IV-24** et la **Figure IV-25**, le front de Pareto de l'instance CCSP19 (comme exemple) obtenu par le NSGA-II en utilisant uniquement l'heuristique HNF_MultiBarge lors de la construction de la population initiale (noté par NF-NSGA-II), et celui obtenu par le NSGA-II lorsque seule l'heuristique HFF_MultiBarge est utilisée (noté par FF-NSGA-II).

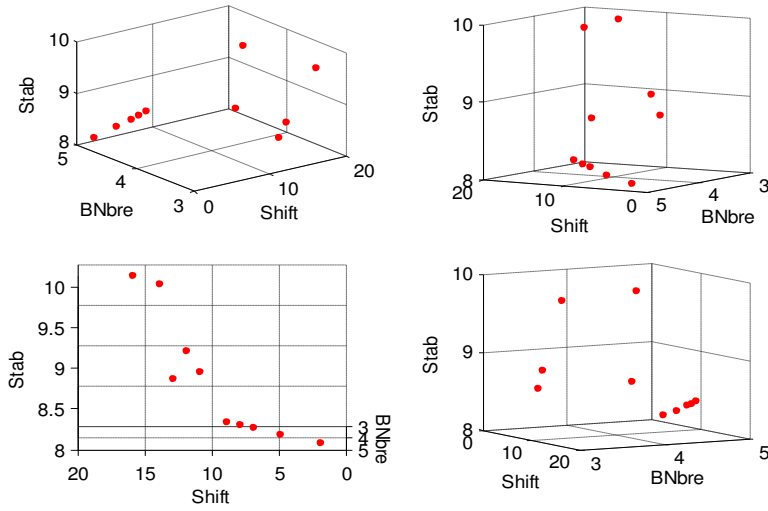


Figure IV-24. Front optimal de Pareto en 3D, de l'instance CCSP19, obtenu par le NSGA-II en utilisant l'heuristique HNF_MultiBarge

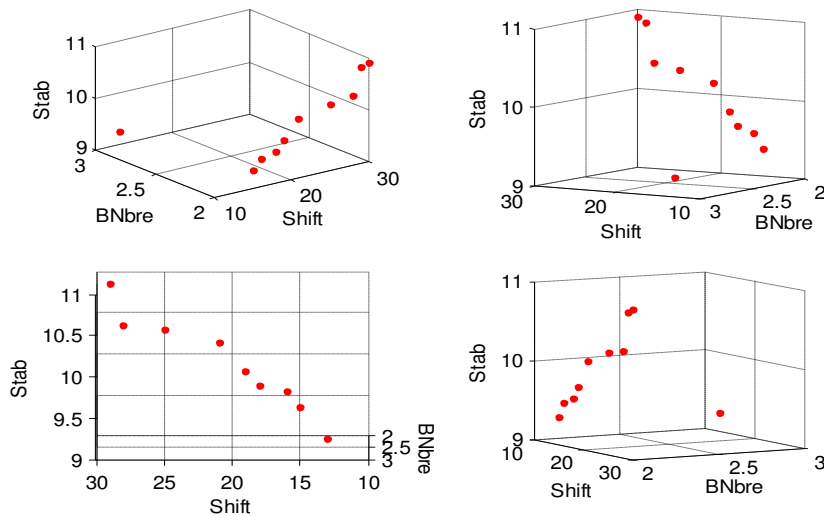


Figure IV-25. Front optimal de Pareto en 3D, de l'instance CCSP19, obtenu par le NSGA-II en utilisant l'heuristique HFF_MultiBarge

D'après les résultats illustrés dans les deux figures ci-dessus (**Figure IV-24** et **Figure IV-25**), nous confirmons que l'heuristique utilisée lors de la génération de la population initiale influence fortement les valeurs des solutions trouvées dans le dernier front. Nous remarquons qu'il y a une diversité au niveau du nombre de barges réellement utilisées dans le front obtenu par le NF-NSGA-II, beaucoup plus que dans celui obtenu par le FF-NSGA-II. Cela est dû à la nature des deux heuristiques *Next Fit* (NF) et *Firt Fit* (FF) du problème de *bin-packing*.

Pour donner une comparaison plus générale des deux heuristiques incluant les trois fonctions objectifs, nous illustrons dans la **Figure IV-26** les valeurs des solutions de chaque individu du front de l'instance CCSP19 par rapport au nombre de *shiftings*, au nombre de barges réellement utilisées et à la stabilité totale du convoi. Nous constatons que le NF-NSGA-II donne les meilleurs résultats en termes de minimisation du nombre de *shiftings*, tandis que le FF-NSGA-II donne les meilleurs résultats en termes de minimisation du nombre de barges utilisées et de maximisation de la distance métacentrique du convoi. Ainsi, chaque méthode présente son propre avantage. Pour cette raison, nous avons généré la population initiale du NSGA-II en se basant sur un choix aléatoire entre les deux heuristiques de chargement proposées HNF_MultiBarge et HFF_MultiBarge lors de chaque exécution.

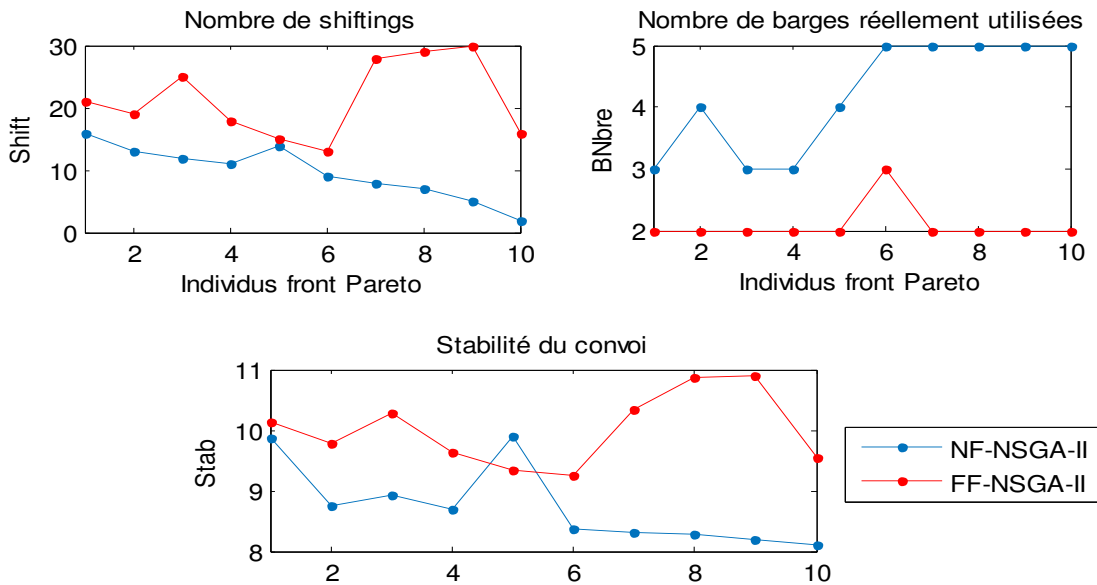


Figure IV-26. Comparaison entre NF-NSGA-II et FF-NSGA-II selon les valeurs de chaque fonction objectif

B. L'influence des caractéristiques des instances sur les plans d'arrimage

Dans cette section, nous étudions l'influence de la nature de l'instance sur la qualité des solutions obtenues par le NSGA-II.

a. L'influence de nombre de ports sur le nombre de *shiftings*

Nous illustrons, dans la **Figure IV-27**, les résultats obtenus par le NSGA-II pour un certain nombre d'instances afin d'observer l'influence de nombre de ports visités par le convoi sur le nombre total de *shiftings* réalisés durant tout le trajet. Dans ce sens, chaque sous-graphe montre le comportement de la fonction objectif « Shift » en fonction du nombre de ports « P » en fixant le nombre de conteneurs à arrimer « InstC » et la capacité du convoi utilisé « Cv ». Par exemple, le sous-graphe « Cv7 InstC10 » montre le nombre de *shiftings* obtenu pour les trois instances CCSP19, CCSP20 et CCSP21 qui correspondent aux cas de chargement de « InstC10 » de conteneurs dans le convoi « Cv7 » visitant ainsi 15, 25 et 35 ports. Chaque valeur de l'axe « Shift » représente la moyenne de nombre de *shiftings* trouvé pour les cinq individus du front de Pareto présenté dans le **Tableau IV-9**.

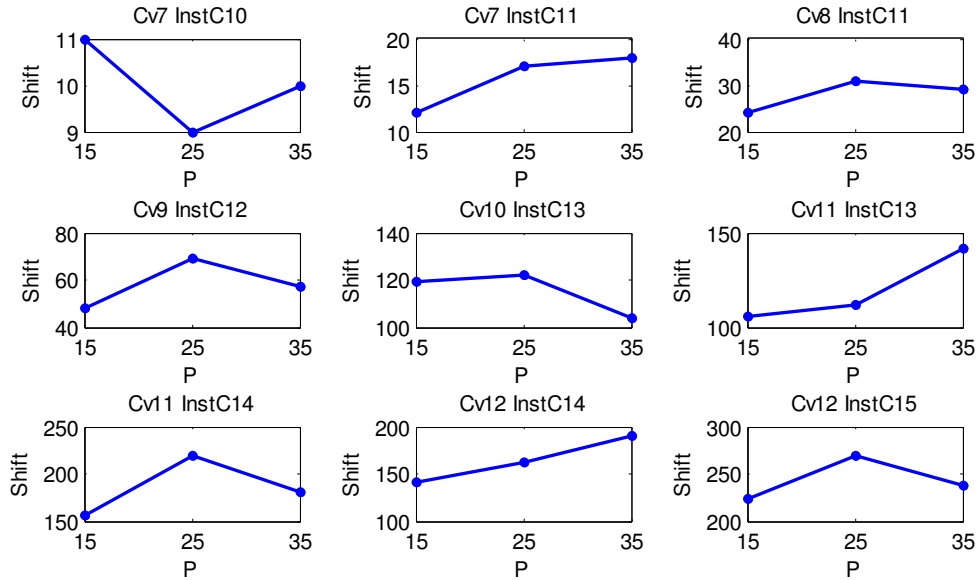


Figure IV-27. L'influence de P sur le nombre de *shiftings*

En analysant les graphes ci-dessus, nous pouvons constater qu'il n'existe pas de corrélation entre le nombre de ports à visiter et le nombre de *shifting* obtenu. De ce fait, si nous augmentons le nombre de ports, le nombre de *shifting* peut ne pas augmenter de la même façon. Par exemple, dans les sous-graphes « Cv9 InstC12 », « Cv10 InstC13 », « Cv11 InstC14 » et « Cv12 InstC15 » le nombre de *shiftings* nécessaire pour effectuer les opérations de déchargement des conteneurs pour 35 ports est inférieur à celui pour 25 ports. Cependant les sous-graphes « Cv7 InstC10 », « Cv7 InstC11 », « Cv11 InstC13 » et « Cv12 InstC14 » montrent un comportement totalement différent. Cela peut être expliqué par l'influence des contraintes de taille et de poids des conteneurs sur le plan de chargement généré.

b. L'influence de nombre de conteneurs sur le nombre de *shiftings*

Dans la **Figure IV-28**, nous illustrons le comportement des valeurs de la fonction objectif « Shift » par rapport à la variation du nombre de conteneurs à transporter « |N| », en fixant le nombre de ports à visiter « P » et le convoi utilisé « Cv ». Par exemple, le sous-graphe

« Cv7 P15 » donne les résultats du nombre de *shiftings* obtenu pour les trois instances CCSP19, CCSP22 et CCSP25 qui correspondent, respectivement, aux cas de chargement de « InstC10 », « InstC11 » et « InstC12 » de conteneurs dans le convoi « Cv7 » pour les transporter aux « P15 » ports. De même, chaque valeur de l'axe « Shift » représente la moyenne de nombre de *shiftings* trouvé pour les cinq individus du front de Pareto rapporté dans le **Tableau IV-9**.

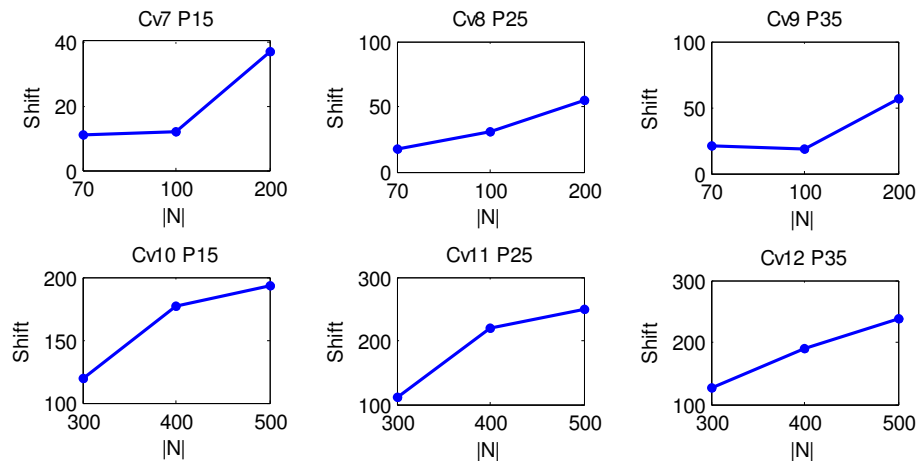


Figure IV-28. L'influence de $|N|$ sur le nombre de *shiftings*

Nous remarquons que le nombre de *shiftings* et le nombre de conteneurs arrimés dans chaque convoi varient dans le même sens dans tous les sous-graphes à part « Cv9 P35 » où le nombre de *shiftings* dans le cas de 100 conteneurs est inférieur à celui dans le cas de 70 conteneurs.

c. L'influence de nombre de conteneurs sur le nombre de barges utilisées

Avant d'aborder notre analyse, nous rappelons, dans le **Tableau IV-10**, les capacités des barges utilisées.

Tableau IV-10. Capacités des barges utilisées dans l'analyse de la **Figure IV-29**

<i>Id_Convoi</i>	Cv7	Cv8	Cv9	Cv10	Cv11	Cv12
<i>B_CapM</i>	90	112	160	200	360	462

Un autre point d'analyse porte sur l'influence du nombre total de conteneurs $|N|$ sur le nombre de barges réellement utilisées dans le convoi. La **Figure IV-29** illustre les différentes solutions obtenues par le NSGA-II pour un ensemble d'instances, chaque valeur « BNbre » indique la moyenne des solutions des cinq individus du front de Pareto correspondant rapporté dans le **Tableau IV-9**. Chaque sous-graphe montre le comportement de la fonction objectif « BNbre » en fonction de nombre conteneurs « $|N|$ » en fixant le nombre de ports à visiter « P » et la capacité du convoi utilisé « Cv ». Par exemple, dans le sous-graphe nommé « Cv7 P15 », figure le nombre de barges utilisées pour les trois instances CCSP19, CCSP22 et CCSP25 qui correspondent, respectivement, aux cas de chargement de « InstC10 », « InstC11 » et « InstC12 » de conteneurs dans le convoi « Cv7 » pour visiter « P15 » ports.

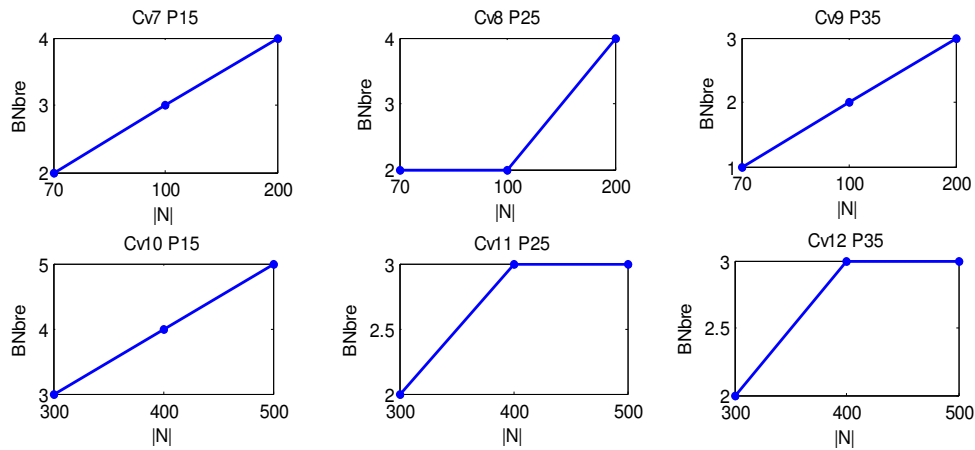


Figure IV-29. L'influence de N sur le nombre de barges réellement utilisées

La **Figure IV-29** confirme que le nombre de barges utilisées dans le convoi est fortement influencé par le nombre total de conteneurs à arrimer, dans le sens où la valeur de « BNbre » augmente si le $|N|$ augmente également.

d. L'influence de nombre de conteneurs sur la stabilité

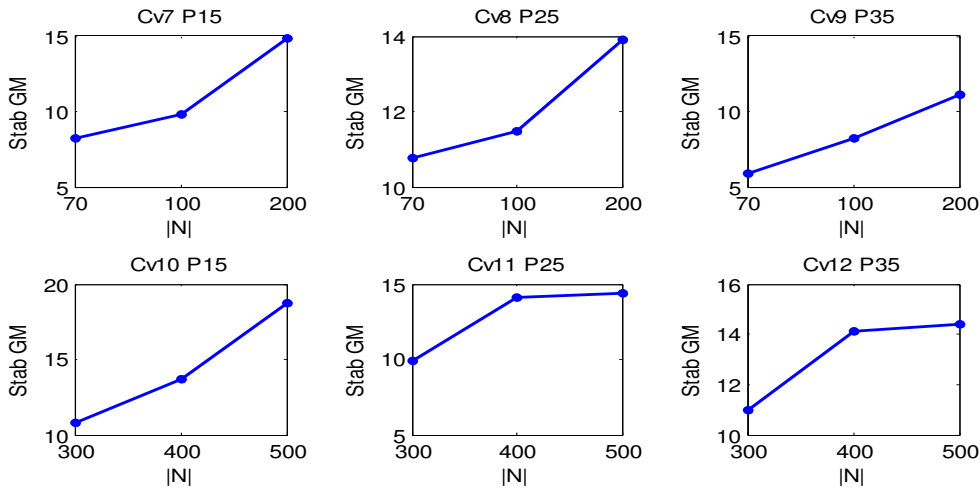


Figure IV-30. L'influence de $|N|$ sur la stabilité

Selon la **Figure IV-30**, nous pouvons remarquer que la stabilité totale du convoi croît dans le même sens que le nombre de conteneurs $|N|$.

5.4.3.5 Analyse de performance des plans de chargement obtenus par NSGA-II

Afin de mesurer la qualité des plans de chargement et d'arrimage obtenus par le NSGA-II par rapport aux trois fonctions objectifs étudiées, nous proposons trois mesures de performance. La première porte sur le nombre de *shiftings* dans les pires des cas, la deuxième sur la distance métacentrique souhaitable en pratique et la troisième sur le nombre de barges utilisées dans les meilleurs des cas.

a. Analyse de performance selon le nombre de *shiftings*

Nous définissons la notion « pire cas pour pile » en tant que pire nombre de *shiftings* nécessaires pour vider une seule pile d'une seule barge dans le cas où la pile est entièrement chargée avec des conteneurs de taille 20' selon un ordre croissant de leur destination à partir du bas de la pile, comme illustré dans la **Figure IV-31**.

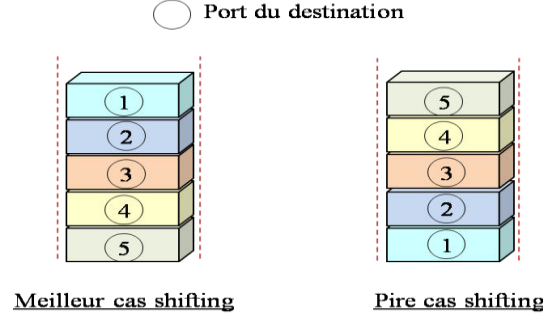


Figure IV-31. Pire et meilleur cas de *shifting* total pour une pile

De ce fait, nous pouvons calculer le nombre de *shiftings* total de tout le convoi de barges dans les pires des cas par la formule suivante :

$$Pire_{Sh} = \sum_{t=1}^{B'} \sum_{i=1}^l \sum_{j=1}^L \sum_{h=1}^{Niv-1} (Niv - h) = B' \times l \times L \times \sum_{h=1}^{Niv-1} (Niv - h)$$

Par ailleurs, pour une instance de 3D-CCSPP avec un convoi de B' barges réellement utilisées, chacune ayant la dimension $l \times L \times Niv$, nous introduisons le taux de *shifting*, qui est une mesure de performance relative pour les plans d'arrimage, selon l'équation suivante :

$$R = \frac{pire_{Sh} - Sh}{pire_{Sh}} \times 100$$

où Sh indique le nombre de *shifting* total obtenu durant toute la tournée du convoi. Parfois, nous nous référons également au rapport de *shifting* par $\frac{Sh}{2N}$ en tant que mesure de performance relative pour les plans d'arrimage, comme utilisé dans (Ding et Chou, 2015), avec N le nombre total de conteneurs à charger et Sh le nombre total de *shiftings* effectués.

Nous rapportons, dans le **Tableau IV-11**, le taux de *shiftings* par rapport au pire des cas pour l'ensemble des instances résolues par le NSGA-II dans le **Tableau IV-9**. L'identifiant de chaque instance est donné dans la colonne 1. La valeur moyenne du nombre de barges utilisées et celle du nombre de *shiftings* effectués pour les cinq individus du front de Pareto sont indiquées, respectivement, dans les colonnes 2 et 3. Nous rappelons qu'une barge est considérée 'totalement' utilisée même si au moins un seul conteneur est chargé à bord. Par conséquent la colonne « BNbre moy » représente le plus petit entier supérieur à la moyenne des nombres de barges utilisées. De même le nombre de *shiftings* est un entier. Les valeurs sous l'entête « Moy » indiquent, donc, la valeur moyenne initiale et les valeurs sous l'entête « Moy Entière > » représentent la partie entière de cette moyenne + 1. Le nombre de *shiftings* dans les pires des cas est donnée dans la colonne « Pire_sh ». Le taux de *shifting* est rapporté dans la dernière colonne « R ».

Tableau IV-11. Analyse de performance des résultats obtenus par NSGA-II en termes de nombres de *shiftings* par rapport au pire des cas

<i>InstCCSP</i>	<i>BNbre moy</i>	<i>Shift moyen</i>		<i>Pire_sh</i>	<i>R</i>
		<i>Moy</i>	<i>Moy Entière ></i>		
CCSP19	2	11	11	180	93,89
CCSP20	2	8,4	9	180	95
CCSP21	2	9,4	10	180	94,44
CCSP22	3	12	12	270	95,56
CCSP23	2	16,6	17	180	90,56
CCSP24	3	17,8	18	270	93,33
CCSP25	4	36,6	37	360	89,72
CCSP26	4	46,2	47	360	86,94
CCSP27	4	42,2	43	360	88,06
CCSP28	2	10,4	11	336	96,73
CCSP29	2	17,8	18	336	94,64
CCSP30	2	15,2	16	336	95,24
CCSP31	2	24	24	336	92,86
CCSP32	2	30,8	31	336	90,77
CCSP33	2	28,6	29	336	91,37
CCSP34	4	64,2	65	672	90,33
CCSP35	4	54,8	55	672	91,82
CCSP36	4	45,2	46	672	93,15
CCSP37	2	19,4	20	480	95,83
CCSP38	1	18,8	19	240	92,08
CCSP39	1	20,2	21	240	91,25
CCSP40	2	26,6	27	480	94,38
CCSP41	2	34,2	35	480	92,71
CCSP42	2	18,4	19	480	96,04
CCSP43	3	47,6	48	720	93,33
CCSP44	3	68,2	69	720	90,42
CCSP45	3	56,2	57	720	92,08
CCSP46	3	118,2	119	900	86,78
CCSP47	3	122	122	900	86,44
CCSP48	3	103,6	104	900	88,44
CCSP49	4	176,4	177	1200	85,25
CCSP50	4	169,8	170	1200	85,83
CCSP51	4	156,6	157	1200	86,92
CCSP52	5	193	193	1500	87,13
CCSP53	5	176	176	1500	88,27
CCSP54	5	219,6	220	1500	85,33
CCSP55	2	105,4	106	1440	92,64
CCSP56	2	111,8	112	1440	92,22
CCSP57	2	141,2	142	1440	90,14
CCSP58	3	156	156	2160	92,78
CCSP59	3	218,6	219	2160	89,86
CCSP60	3	180	180	2160	91,67
CCSP61	3	260,4	261	2160	87,92
CCSP62	3	248,4	249	2160	88,47
CCSP63	3	220,6	221	2160	89,77
CCSP64	2	117,6	118	2310	94,89
CCSP65	2	110	110	2310	95,24
CCSP66	2	125,4	126	2310	94,55
CCSP67	3	141,4	142	3465	95,9
CCSP68	3	162,2	163	3465	95,3
CCSP69	3	189,4	190	3465	94,52
CCSP70	3	223,2	224	3465	93,54
CCSP71	3	268,2	269	3465	92,24
CCSP72	3	236,6	237	3465	93,16

D'après l'analyse des résultats présentés dans le **Tableau IV-11**, nous pouvons confirmer la haute qualité des plans de chargement et d'arrimage obtenus par le NSGA-II par rapport aux valeurs de la fonction objectif *shifting*. En effet, pour la majorité des instances étudiées, l'écart entre le nombre de *shiftings* trouvé et celui dans les pires des cas est supérieur à 90%.

b. Analyse de performance selon la stabilité GM

Dans cette section, nous évaluons la qualité des solutions obtenues par le NSGA-II par rapport à la distance métacentrique souhaitable en pratique.

Dans le cas général, les règles de l'inspection du transport maritime prévoient que la valeur de GM doit être supérieure à 1m si les conteneurs sont arrimés, et de préférence, qu'elle ne doit pas dépasser 1.5 m, mais cela peut changer en fonction de la conception du navire et des conditions de chargement (Imai et al., 2002, 2006). De ce fait, pour évaluer nos résultats, nous comparons la moyenne des valeurs de GM obtenue par le NSGA-II, pour les cinq individus du front de Pareto rapportés dans le **Tableau IV-9**, avec la plage souhaitable $[GM_{min}, GM_{max}] = [1, 1.5]$ mètres d'une barge donnée. En outre, la nature du problème 3D-CCSP, qui est basée sur l'utilisation d'un ensemble de barges et non pas une seule, impose la redéfinition de l'intervalle $[GM_{min}, GM_{max}]$ pour chaque instance en fonction du nombre de barges utilisées dans le convoi. Nous rappelons que l'unité de GM considérée dans ce travail est en pieds et que 1 mètre est équivalent à 3.28084 pieds.

Le **Tableau IV-12** résume les résultats obtenus après la comparaison de GM souhaitable avec celle trouvée par notre algorithme. La première colonne indique l'identifiant de l'instance. La plage $[GM_{min}, GM_{max}]$ correspondante à chaque instance est donnée dans les colonnes 2 et 3, $[GM_{min}, GM_{max}] = [3.28084 * BNbre, 4.92126 * BNbre]$. La quatrième colonne donne la stabilité moyenne obtenue pour les cinq individus du front de Pareto. La déviation de GM de tout le convoi par rapport à la plage souhaitable est donnée dans la colonne « Dev » selon les cas suivants :

- ▶ Si $Stab_{moy} < GM_{min}$, alors $Dev = \frac{GM_{min} - Stab_{moy}}{GM_{min}} \times 100$
- ▶ Si $Stab_{moy} > GM_{max}$, alors $Dev = \frac{Stab_{moy} - GM_{max}}{GM_{max}} \times 100$
- ▶ Si $Stab_{moy} \in [GM_{min}, GM_{max}]$, alors $Dev = 0$

Les deux dernières colonnes « Dev min » et « Dev max » précisent, respectivement, le sens de la déviation générée par rapport à « GM_min » et « GM_max ». Les solutions qui appartiennent à l'intervalle souhaitable sont indiquées en gras dans la colonne « Dev ».

Tableau IV-12. Analyses de performance du facteur de stabilité GM obtenu par NSGA-II par rapport à 1m et 1.5m par barge

<i>InstCCSP</i>	<i>GM_min</i>	<i>GM_max</i>	<i>Stab moy</i>	<i>Dev</i>	<i>Dev min</i>	<i>Dev max</i>
CCSP19	6.56168	9.84252	8.195	0	-	-
CCSP20	6.56168	9.84252	8.346	0	-	-
CCSP21	6.56168	9.84252	7.743	0	-	-
CCSP22	9.84252	14.76378	9.822	0.21	x	-
CCSP23	6.56168	9.84252	8.286	0	-	-
CCSP24	6.56168	9.84252	8.504	0	-	-
CCSP25	13.12336	19.68504	14.89	0	-	-
CCSP26	13.12336	19.68504	13.712	0	-	-
CCSP27	13.12336	19.68504	12.189	7.12	x	-
CCSP28	6.56168	9.84252	8.729	0	-	-
CCSP29	6.56168	9.84252	10.772	9.44	-	x
CCSP30	6.56168	9.84252	9.821	0	-	-
CCSP31	6.56168	9.84252	9.773	0	-	-
CCSP32	6.56168	9.84252	11.479	16.63	-	x
CCSP33	6.56168	9.84252	9.392	0	-	-
CCSP34	13.12336	19.68504	15.355	0	-	-
CCSP35	13.12336	19.68504	13.898	0	-	-
CCSP36	13.12336	19.68504	15.977	0	-	-
CCSP37	6.56168	9.84252	10.871	10.45	-	x
CCSP38	3.28084	4.92126	5.706	15.95	-	x
CCSP39	3.28084	4.92126	5.87	19.28	-	x
CCSP40	6.56168	9.84252	10.86	10.34	-	x
CCSP41	6.56168	9.84252	11.853	20.43	-	x
CCSP42	6.56168	9.84252	8.193	0	-	-
CCSP43	9.84252	14.76378	11.161	0	-	-
CCSP44	9.84252	14.76378	10.957	0	-	-
CCSP45	9.84252	14.76378	11.101	0	-	-
CCSP46	9.84252	14.76378	10.75	0	-	-
CCSP47	9.84252	14.76378	10.609	0	-	-
CCSP48	9.84252	14.76378	10.733	0	-	-
CCSP49	13.12336	19.68504	13.68	0	-	-
CCSP50	13.12336	19.68504	12.535	4.48	x	-
CCSP51	13.12336	19.68504	18.882	0	-	-
CCSP52	16.4042	24.6063	18.704	0	-	-
CCSP53	16.4042	24.6063	19.858	0	-	-
CCSP54	16.4042	24.6063	16.885	0	-	-
CCSP55	6.56168	9.84252	9.794	0	-	-
CCSP56	6.56168	9.84252	9.914	0.73	-	x
CCSP57	6.56168	9.84252	10.86	10.34	-	x
CCSP58	9.84252	14.76378	14.54	0	-	-
CCSP59	9.84252	14.76378	14.117	0	-	-
CCSP60	9.84252	14.76378	14.853	0.6	-	x
CCSP61	9.84252	14.76378	14.518	0	-	-
CCSP62	9.84252	14.76378	14.412	0	-	-
CCSP63	9.84252	14.76378	14.45	0	-	-
CCSP64	6.56168	9.84252	11.173	13.52	-	x
CCSP65	6.56168	9.84252	11.108	12.86	-	x
CCSP66	6.56168	9.84252	10.96	11.35	-	x
CCSP67	9.84252	14.76378	15.66	6.07	-	x
CCSP68	9.84252	14.76378	15.116	2.39	-	x
CCSP69	9.84252	14.76378	14.13	0	-	-
CCSP70	9.84252	14.76378	16.689	13.04	-	x
CCSP71	9.84252	14.76378	8.923	9.34	x	-
CCSP72	9.84252	14.76378	14.389	0	-	-

« x » dans la colonne **Dev min** indique que la déviation correspondante est par rapport à GM_min

« x » dans la colonne **Dev max** indique que la déviation correspondante est par rapport à GM_max

« - » pas de deviation par rapport à GM souhaitable

Bien que la valeur GM obtenue soit faible pour quelques instances (indiquées par le caractère « x » dans la colonne « Dev min »), la plupart des solutions semblent raisonnables, ayant une distance GM limitée à la plage correspondante.

c. Analyse de performance selon le nombre de barges utilisées

Afin d'évaluer la qualité des solutions obtenues par le NSGA-II par rapport au nombre de barges réellement utilisées dans le convoi, nous introduisons la notion de nombre de barges dans le meilleurs cas « Meill Cas » comme suit :

$$Meill.Cas = \frac{N_{TEU}}{B_{CapM}}$$

Plus l'écart de la solution par rapport au meilleur cas est faible, plus la solution est meilleure en termes de nombre de barges réellement utilisées.

Dans le **Tableau IV-13**, nous comparons les solutions trouvées en termes de nombre de barges par le NSGA-II avec le meilleur cas. Les différentes caractéristiques des instances étudiées sont résumés dans les quatre premières colonnes « InstCCSP », « B_CapM », « |N| » et « N_TEU », indiquant respectivement le nom de l'instance, la capacité maximale de la barge constituant le convoi correspondant, le nombre de conteneurs à arrimer et le nombre de conteneurs en EVP en considérant un conteneur de taille 40' comme étant deux conteneurs de 20' ($N_{TEU} = 2 N1 + N2$). La valeur moyenne du nombre de barges utilisées des cinq individus du front de Pareto du **Tableau IV-9** est indiquée dans la colonne « BNbre moy ». Le meilleur cas est calculé dans la colonne « Meill. Cas », et sa partie entière +1 est indiquée dans la colonne « Meill. Cas Entier > ». La dernière colonne « Dev » représente la différence entre le nombre de barges obtenu par notre méthode et celui dans le meilleurs des cas ($Dev = [Meill.Cas] + 1 - BNbre_{moy}$).

Tableau IV-13. Analyse de performance des résultats obtenus par le NSGA-II en termes de nombre de barges réellement utilisées par rapport au meilleur cas

<i>InstCCSP</i>	<i>B_CapM</i>	<i>/N/</i>	<i>N_TEU</i>	<i>BNbre moy</i>	<i>Meill. Cas</i>	<i>Meill. Cas Entier ></i>	<i>Dev</i>
CCSP19	90	70	104	2	1,16	2	0
CCSP20				2	1,16	2	0
CCSP21				2	1,16	2	0
CCSP22		100	175	3	1,94	2	1
CCSP23				2	1,94	2	0
CCSP24				3	1,94	2	1
CCSP25		200	350	4	3,89	4	0
CCSP26				4	3,89	4	0
CCSP27				4	3,89	4	0
CCSP28	112	70	104	2	0,93	1	1
CCSP29				2	0,93	1	1
CCSP30				2	0,93	1	1
CCSP31		100	175	2	1,56	2	0
CCSP32				2	1,56	2	0
CCSP33				2	1,56	2	0
CCSP34		200	350	4	3,13	4	0
CCSP35				4	3,13	4	0
CCSP36				4	3,13	4	0
CCSP37	160	70	104	2	0,65	1	1
CCSP38				1	0,65	1	0
CCSP39				1	0,65	1	0
CCSP40		100	175	2	1,09	2	0
CCSP41				2	1,09	2	0
CCSP42				2	1,09	2	0
CCSP43		200	350	3	2,19	3	0
CCSP44				3	2,19	3	0
CCSP45				3	2,19	3	0
CCSP46	200	300	460	3	2,30	3	0
CCSP47				3	2,30	3	0
CCSP48				3	2,30	3	0
CCSP49		400	611	4	3,06	4	0
CCSP50				4	3,06	4	0
CCSP51				4	3,06	4	0
CCSP52		500	761	5	3,81	4	1
CCSP53				5	3,81	4	1
CCSP54				5	3,81	4	1
CCSP55	360	300	460	2	1,28	2	0
CCSP56				2	1,28	2	0
CCSP57				2	1,28	2	0
CCSP58		400	611	3	1,70	2	1
CCSP59				3	1,70	2	1
CCSP60				3	1,70	2	1
CCSP61		500	761	3	2,11	3	0
CCSP62				3	2,11	3	0
CCSP63				3	2,11	3	0
CCSP64	462	300	460	2	1,00	1	1
CCSP65				2	1,00	1	1
CCSP66				2	1,00	1	1
CCSP67		400	611	3	1,32	2	1
CCSP68				3	1,32	2	1
CCSP69				3	1,32	2	1
CCSP70		500	761	3	1,65	2	1
CCSP71				3	1,65	2	1
CCSP72				3	1,65	2	1

La **Figure IV-32** illustre les résultats de comparaison d'une façon plus claire, dans laquelle, nous pouvons confirmer que les résultats obtenus par le NSGA-II sont de très bonne qualité. Toutefois, nous remarquons que le NSGA-II renvoie des solutions avec, au plus, une seule barge dans le cas où la déviation n'est pas nulle. Cela peut être justifié par le fait que le meilleur cas calculé ne prend pas en considération les contraintes de taille des conteneurs et de fragilité des piles.

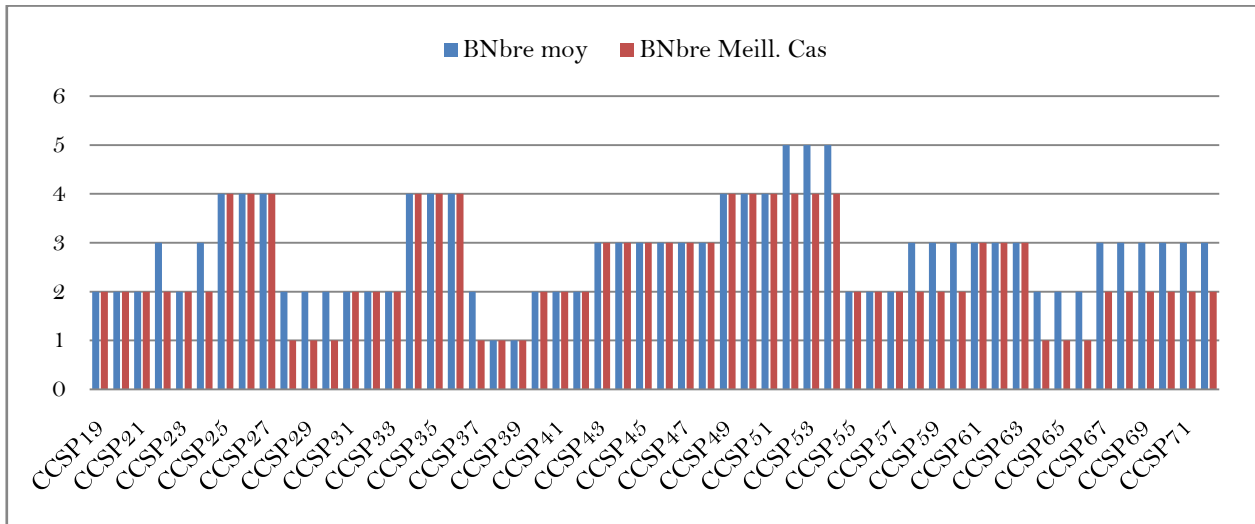


Figure IV-32. Comparaison entre le nombre de barges réellement utilisées dans le convoi obtenu par NSGA-II et celui dans le meilleur cas

6 Conclusion

Dans ce chapitre, nous avons d'abord proposé un modèle mathématique non linéaire en nombres entiers du problème d'optimisation des plans de chargement et d'arrimage des conteneurs en considérant un cas réel où le chargement est effectué dans un ensemble de barges et non pas une seule, constituant ainsi un système de convoi dans lequel plusieurs exigences réelles relatives à la fois aux conteneurs et au convoi doivent être considérées. Cette variante du problème, notée 3D-CCSPP, est traitée sur la base de sa relation avec le problème de *bin-packing* (BPP). Nous considérons l'aspect multi-objectif du 3D-CCSPP avec trois objectifs réalistes à optimiser : le nombre de *shiftings* durant tout le trajet, la stabilité du convoi selon le facteur de la distance métacentrique et le nombre de barges réellement utilisées dans le convoi. C'est un aspect nouveau et efficace compte tenu de l'état de l'art des problèmes de chargement des barges.

Une série de tests expérimentaux est ensuite effectuée à l'aide du solveur CPLEX. Les résultats de calcul rapportés démontrent l'efficacité du modèle proposé. Puis, une nouvelle adaptation du NSGA-II est proposée en se basant sur un ensemble d'heuristiques introduites par les méthodes de résolution du problème de *bin-packing* en 1D.

Pour mesurer la qualité de l'algorithme proposé (NSGA-II intégrant les heuristiques de *bin-packing*), nous avons comparé ses résultats avec les solutions optimales ou les mieux connues données par le solveur CPLEX pour les instances de petite taille. Pour les instances de moyenne et de grande taille, pour lesquelles CPLEX ne peut trouver de solution réalisable, nous avons généré

un ensemble de mesures de performances pour analyser la qualité des solutions obtenues. Les résultats obtenus démontrent l'efficacité de la méthode utilisée. Les plans de chargement et d'arrimage optimisés répondent à toutes les exigences définies, en fonction des trois objectifs étudiés, ils sont affichés au planificateur sous la forme d'un front de Pareto. C'est ensuite son rôle de sélectionner la solution qui correspond le mieux à ses besoins en se basant sur son expérience antérieure.

Conclusion générale et travaux futurs

La conteneurisation est un puissant facteur d'intégration intermodale, permettant aux modes de transport maritime et fluvial, d'une part, et terrestre, d'autre part, de s'interconnecter plus efficacement. Toutefois, pour faire face à l'énorme croissance des échanges conteneurisés dans le monde, tous les processus opérationnels de manutention des conteneurs doivent être repensés et correctement optimisés. En effet, les coûts de la construction de postes à quai et de l'exploitation d'équipements d'un terminal à conteneurs, y compris des grues de quai, sont très élevés pour les exploitants de terminaux à conteneurs, de même, les coûts d'accostage et de manutention sont extrêmement élevés pour les exploitants de navires fluviaux. Cela souligne l'importance d'achever tous les processus aussi efficacement que possible, dans le but d'obtenir une réduction significative des coûts et une augmentation des profits à la fois pour les opérateurs portuaires et fluviaux.

Compte tenu de ces différents enjeux, l'optimisation des mouvements de chargement et déchargement est primordiale pour l'efficacité de la chaîne logistique conteneurisée. Notre travail a été consacré à l'étude de l'influence des mouvements improductives de chargement/déchargement, appelés « *shiftings* » sur le coût total du transport. Deux aspects principaux ont été pris en compte.

- ▶ Dans le cas du transport routier où le plan de chargement est connu à l'avance, nous avons traité le problème qui consiste à chercher la tournée optimale d'un véhicule en considérant, en plus du coût distance, le coût *shifting*
- ▶ Dans le cas du transport fluvial où la tournée est connue à l'avance, notre objectif était de chercher le plan de chargement optimal des conteneurs dans les unités fluviales « barges » en prenant en considération plusieurs facteurs, tels que, le coût *shifting*, les mesures de stabilité, la notion de fragilité des piles, la taille des conteneurs et la structure physique des barges. Nous avons traité le cas d'une seule barge et le cas d'un convoi à plusieurs barges.

Nous avons commencé ce mémoire par des généralités autour du concept de conteneurisation, les interconnexions des différents modes de transport, les différents enjeux des problèmes de chargement/déchargement et les différentes formes du problème de *bin-packing* (*Bin-Packing Problem*, (BPP)). Tous ces éléments introductifs nous ont permis d'identifier les enjeux des opérations *shiftings* et leurs influences sur l'efficacité de transport de marchandise.

Dans le deuxième chapitre, nous avons formellement introduit le *shifting* dans le problème de voyageur de commerce (*Shifting Traveling Salesman Problem*, (ShTSP)). Notre objectif était de rechercher un circuit hamiltonien optimal qui prenne en compte le coût de déplacement et le coût de *shifting* représenté par l'enlèvement temporaire des produits dans le véhicule, causé par les opérations de déchargement et de rechargement ayant lieu chez chaque nœud du circuit. Nous avons proposé un modèle mathématique approprié, puis, à cause de son NP-complétude, nous avons résolu notre problème en utilisant une méthode directe exhaustive et deux méthodes approchées : la première consiste à adapter la méta-heuristique des colonies de fourmis séquentielles et la seconde introduit une nouvelle adaptation des fourmis parallèles, les algorithmes sont testés pour un certain nombre d'instances de caractéristiques différents à partir des données de référence de TSPLIB. Les résultats de calcul montrent l'efficacité de la version améliorée de l'algorithme qui repose sur le concept parallèle, pour les instances de petite et de grande taille.

Le troisième chapitre a eu pour objectif l'intégration d'un cas inspiré par des problèmes réels de chargement et d'arrimage des conteneurs dans les terminaux portuaires fluviaux, où les coûts de chargement/déchargement sont non-uniformes. Le problème concerne l'optimisation des plans d'arrimage d'un ensemble de conteneurs ayant des tailles, des poids, des fragilités et des ports de destination différents dans une seule barge. Tous les conteneurs doivent être chargés à bord en optimisant le coût total de *shiftings* et la stabilité de la barge. Nous avons proposé un modèle mathématique approprié dont l'objectif est de minimiser le nombre de *shiftings* et de maximiser la stabilité totale de la barge selon les deux mesures de stabilité transversale et longitudinale. Une série de tests expérimentaux est ensuite effectuée à l'aide du solveur CPLEX afin de vérifier la robustesse du modèle proposé et démontrer son efficacité.

Le quatrième chapitre a été dédié à l'étude du problème d'optimisation des plans de chargement et d'arrimage des conteneurs dans un système de convoi de barges qui est principalement motivé par le concept de massification dans le transport fluvial. Nous avons étudié ce problème dans le cas multi-objectif selon sa relation avec le problème de *bin-packing* en 3D. Nous avons proposé un modèle mathématique approprié dont l'objectif est d'optimiser le nombre de *shiftings*, la stabilité du convoi selon la distance métacentrique et le nombre de barges réellement utilisées dans le convoi. Une série d'exemples pratiques résolus par le solveur CPLEX est ensuite rapportée afin de montrer l'efficacité du modèle. Ensuite, nous avons proposé une nouvelle adaptation du NSGA-II en se basant sur un ensemble d'heuristiques introduites par les méthodes de résolution du problème de *bin-packing* en une dimension. L'algorithme peut produire un ensemble de solutions non-dominées, c'est-à-dire, un front de Pareto, les décideurs peuvent ensuite choisir la solution la plus prometteuse pour une mise en œuvre pratique en fonction de leur expérience et de leurs préférences. L'ensemble des résultats numériques obtenus est enfin présenté et commenté en étudiant l'influence des caractéristiques des instances sur les plans d'arrimages, puis évalué, en utilisant des mesures de performances que nous avons proposées en se basant sur le pire et le meilleur des cas.

Nos travaux de recherche ont été publiés dans des revues internationales à comité de lecture, et d'autres, ont été présentés dans des manifestations scientifiques internationales et nationales :

- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., «The Shifting Traveling Salesman Problem: from modeling to resolution», *Journal of Engineering and Applied Sciences*, 13: 5911-5925, 2018. DOI: 10.3923/jeasci.2018.5911.5925
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Multi-objective optimization of the 3D container stowage planning problem in a barge convoy system : modelling and resolution using NSGA-II with Bin-Packing features » (*soumis à une revue scientifique*)
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « 3D container stowage planning problem with non-uniform *shifting* fees and dynamic stack weight limit : the case of a single barge » (*soumis à une revue scientifique*)
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Multi-objective river-sea-going container barge stowage planning problem with container fragility and barge stability factors », In : 4th International IEEE Conference on Logistics Operations Management (GOL'18), p. 1-6, Université Le Havre Normandie, Le Havre, France, les 10-12 April 2018. DOI: 10.1109/GOL.2018.8378102
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Multi-objective container barge transport problem », In : 3th International IEEE Conference on Logistics Operations Management (GOL'16), p. 1-6, FST Fès, Université Sidi Mohamed Ben Abdellah Fès, Maroc, les 23, 24 et 25 Mai 2016. DOI: 10.1109/GOL.2016.7731664
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Non-dominated sorting genetic algorithm-II applied to multi-objective environmentally-friendly single-barge loading problem with weight distribution limitation ». 88e Journées du Groupe de Travail Européen « Aide Multicritère à la Décision », Lisbonne, Portugal, Septembre 27-29, 2018
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Génération des plans de chargement des barges avec stabilité », 5^{ème} Rencontre du Laboratoire Modélisation et Calcul Scientifique (RLMCS'2017), Faculté des Sciences et Techniques de Fès, Université Sidi Mohamed Ben Abdellah Fès, Maroc, les 23 et 24 Octobre 2017
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Container barge stowage problem with *shifting* constraints », 5^{ème} Congrès International de la Société Marocaine des Mathématiques Appliquées (SM2A), Université Moulay Ismail, Meknès, Maroc, les 16, 17 et 18 Mars 2017
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Problème de *bin-packing* et ses variantes », 1^{er} colloque national sur les Techniques Economiques Décisionnelles (TED'16), FSJES Meknès, Université Moulay Ismail, Meknès, Maroc, le 14 Mai 2016

- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Le '*shifting*' dans le problème de voyageur de commerce », 17^{ème} édition du Congrès de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF), Université de Technologie Compiègne, France, les 10, 11 et 12 Février 2016.
http://roadef2016.utc.fr/papiers/ROADEF_2016_paper_247.pdf
- ▶ EL YAAGOUBI A., EL HILALI ALAOUI A., BOUKACHOUR J., « Problème d'arrimage des conteneurs dans les barges », 1^{er} Forum des doctorants pôle de Recherche Technologie de l'Information et de Communication, Systèmes et Modélisation (TICSM), Faculté des Sciences et Techniques de Fès, Université Sidi Mohamed Ben Abdellah Fès, Maroc, le 17 Décembre 2015

Les problèmes de chargement/déchargement traités dans cette thèse représentent un axe de recherche très intéressant et très vaste, surtout, dans le cadre du transport conteneurisé. Par ailleurs, d'autres extensions méritent d'être approfondies. Nous présentons, dans ce qui suit, un ensemble d'axes qui peuvent être l'objet de nos futurs travaux :

- ▶ Le premier problème, que nous sommes entrain de développer, est le problème de chargement et d'arrimage des conteneurs dans un convoi fluvial poussé de barges lors de sa rotation multiports, où à chaque port d'autres conteneurs peuvent être chargés rejoignant la rotation du convoi. Compte tenu des informations relatives aux conteneurs d'exportation dans chaque port, nous cherchons à optimiser le nombre total des mouvements *shiftings* effectués tout au long du voyage, le nombre de barges réellement utilisées et la stabilité de chaque barge dans le convoi (au lieu de considérer la stabilité totale comme dans le chapitre IV). Nous avons proposé une approche de résolution basée sur l'heuristique MultiBagre, introduite dans le chapitre IV, mais en s'appuyant, cette fois-ci, sur la stratégie *Best Fit* du problème de *bin-packing*, dont l'objectif consiste à remplir la barge la plus chargée (ou pleine) en répétant cette procédure à chaque port, avec l'hypothèse que la barge n'est pas vide. Il est à noter qu'une pile d'une barge est considérée non vide si sa fragilité est inférieure à la fragilité maximale initiale ;
- ▶ D'autres types de conteneurs peuvent être intégrés, imposant ainsi un conditionnement et un arrimage plus particulier, notamment, les conteneurs frigorifiques (*reefer*) qui permettent le transport de marchandises sous température dirigée et les conteneurs de matières dangereuses qui sont distingués par les règlements internationaux et nationaux selon les risques d'explosions et d'incendies élevés ;
- ▶ Pour un problème plus réaliste, nous envisageons, également, de combiner le problème de planification des grues de quai avec celui de l'optimisation des plans de chargement et d'arrimage dans un convoi de barges, puisque, en pratique, le plan doit tenir compte des possibilités d'interférences entre les grues de quai lors des opérations de chargement/déchargement. Une étude intéressante peut traiter ce problème en utilisant une approche alternative appelée Simulation-Optimisation ;
- ▶ Enfin, une autre étude aussi importante peut porter sur le problème d'optimisation des plans de chargement et d'arrimage des conteneurs dans un convoi de barges dans le cas où l'ensemble de conteneurs à charger dans les ports intermédiaires n'est pas connu à l'avance.

Liste de références

- AIMU American Institute of Marine Underwriters. (Retrieved 12 Novembre 2018). On deck stowage of containers. *www.aimu.org*.
- Abeysooriya, R. P., Bennell, J. A., & Martinez-Sykora, A. (2018). Jostle heuristics for the 2D-irregular shapes bin packing problems with free rotation. *International Journal of Production Economics*, 195, 12-26.
- Agerschou, H. (2004). Planning and design of ports and marine terminals. Thomas Telford.
- Al-Dhaheri, N., & Diabat, A. (2017). A Lagrangian relaxation-based heuristic for the multi-ship quay crane scheduling problem with ship stability constraints. *Annals of Operations Research*, 248 (1-2), 1-24.
- Alfandari, L., Davidović, T., Furini, F., Ljubić, I., Maraš, V., & Martin, S. (2019). Tighter MIP models for barge container ship routing. *Omega*, 82, 38-54.
- Ambrosino, D., Anghinolfi, D., Paolucci, M., & Sciomachen, A. (2010, May). An experimental comparison of different heuristics for the master bay plan problem. In *International Symposium on Experimental Algorithms* (pp. 314-325). Springer, Berlin, Heidelberg.
- Ambrosino, D., Paolucci, M., & Sciomachen, A. (2017). Computational evaluation of a MIP model for multi-port stowage planning problems. *Soft Computing*, 21(7), 1753-1763.
- Ambrosino, D., Sciomachen, A., & Tanfani, E. (2004). Stowing a containership: the master bay plan problem. *Transportation Research Part A: Policy and Practice*, 38(2), 81-99.
- Ambrosino, D., Sciomachen, A., & Tanfani, E. (2006). A decomposition heuristics for the container ship stowage problem. *Journal of Heuristics*, 12(3), 211-233.
- Araújo, E. J., Chaves, A. A., de Salles Neto, L. L., & de Azevedo, A. T. (2016). Pareto clustering search applied for 3D container ship loading plan problem. *Expert Systems with Applications*, 44, 50-57.
- Aslidis, A. H. (1989). *Combinatorial algorithms for stacking problems* (Doctoral dissertation, Massachusetts Institute of Technology).
- Avriel, M., & Penn, M. (1993). Exact and approximate solutions of the container ship stowage problem. *Computers & industrial engineering*, 25(1-4), 271-274.
- Avriel, M., Penn, M., & Shpirer, N. (2000). Container ship stowage problem: complexity and connection to the coloring of circle graphs. *Discrete Applied Mathematics*, 103(1-3), 271-279.
- Avriel, M., Penn, M., Shpirer, N., & Witteboon, S. (1998). Stowage planning for container ships to reduce the number of shifts. *Annals of Operations Research*, 76, 55-71.

- Azevedo, A. T., de Salles Neto, L. L., Chaves, A. A., & Moretti, A. C. (2018). Solving the 3D stowage planning problem integrated with the quay crane scheduling problem by representation by rules and genetic algorithm. *Applied Soft Computing*, 65, 495-516.
- Baldi, M. M., Crainic, T. G., Perboli, G., & Tadei, R. (2014). Branch-and-price and beam search algorithms for the variable cost and size bin-packing problem with optional items. *Annals of Operations Research*, 222(1), 125-141.
- Bansal, N., Liu, Z., & Sankar, A. (2009). Bin-packing with fragile objects and frequency allocation in cellular networks. *Wireless Networks*, 15(6), 821-830.
- Barrass, B. (2000). *Ship stability: notes and examples*. Elsevier.
- Barrass, B., & Derrett, C. D. (1999). *Ship stability for masters and mates*, 5th Edition, Butterworth-Heinemann. URL <https://www.123library.org/ebook/isbn/9780080480886/>
- Battarra, M., Erdoĝan, G., Laporte, G., & Vigo, D. (2010). The traveling salesman problem with pickups, deliveries, and handling costs. *Transportation Science*, 44(3), 383-399.
- Bektas, T. (2006). The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, 34(3), 209-219.
- Berkey, J. O. (1988, October). Massively parallel computing applied to the one-dimensional bin-packing problem. In *Frontiers of Massively Parallel Computation, 1988. Proceedings., 2nd Symposium on the Frontiers of* (pp. 317-319). IEEE.
- Berkey, J. O., & Wang, P. Y. (1994). A systolic-based parallel bin-packing algorithm. *IEEE transactions on parallel and distributed systems*, 5(7), 769-772.
- Bigras, L. P., Gamache, M., & Savard, G. (2008). The time-dependent traveling salesman problem and single machine scheduling problems with sequence dependent setup times. *Discrete Optimization*, 5(4), 685-699.
- Blickle, T., & Thiele, L. (1996). A comparison of selection schemes used in evolutionary algorithms. *Evolutionary Computation*, 4(4), 361-394.
- Bloemhof, J. M., Van der Laan, E. A., & Beijer, C. (2011). Sustainable Inland Transportation. *International Journal of Business Insights & Transformation*, 3(3), 26-33.
- Boutevin, C., Gourgand, M., & Norre, S. (2003, July). Bin packing extensions for solving an industrial line balancing problem. In *ISATP* (pp. 115-121). IEEE.
- Bouzidi, S., Zhang, Y., & Ye, Q. C. (2014). Barge processing policies in container terminals: A multi-agent simulation study.
- Carrabs, F., Cordeau, J. F., & Laporte, G. (2007). Variable neighborhood search for the pickup and delivery traveling salesman problem with LIFO loading. *INFORMS Journal on Computing*, 19(4), 618-632.
- Cherkesly, M., Desaulniers, G., & Laporte, G. (2015). Branch-price-and-cut algorithms for the pickup and delivery problem with time windows and last-in-first-out loading. *Transportation Science*, 49(4), 752-766.
- Cherkesly, M., Desaulniers, G., Irnich, S., & Laporte, G. (2016). Branch-price-and-cut algorithms for the pickup and delivery problem with time windows and multiple stacks. *European Journal of Operational Research*, 250(3), 782-793.
- Christiansen, M., Fagerholt, K., & Ronen, D. (2004). Ship routing and scheduling: Status and perspectives. *Transportation science*, 38(1), 1-18.

- Cordeau, J. F., Iori, M., Laporte, G., & Salazar González, J. J. (2010). A branch-and-cut algorithm for the pickup and delivery traveling salesman problem with LIFO loading. *Networks*, 55(1), 46-59.
- Cordeau, J. F., Laporte, G., Legato, P., & Moccia, L. (2005). Models and tabu search heuristics for the berth-allocation problem. *Transportation science*, 39(4), 526-538.
- Crainic, T. G., & Kim, K. H. (2007). Intermodal transportation. *Handbooks in operations research and management science*, 14, 467-537.
- Cruz-Reyes, L., Hernández, P., & Melin, P. (2013). Constructive algorithm for a benchmark in ship stowage planning. In *Recent Advances on Hybrid Intelligent Systems* (pp. 393-408). Springer, Berlin, Heidelberg.
- De Jong, K. A. (1975). Analysis of the behavior of a class of genetic adaptive systems.
- Deb, K., & Goel, T. (2001, March). Controlled elitist non-dominated sorting genetic algorithms for better convergence. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp. 67-81). Springer, Berlin, Heidelberg.
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2), 182-197.
- Delgado, A. (2013). *Models and Algorithms for Container Vessel Stowage Optimization* (Doctoral dissertation, PhD thesis, IT University Copenhagen).
- Delgado, A., Jensen, R. M., & Schulte, C. (2009, September). Generating optimal stowage plans for container vessel bays. In *International Conference on Principles and Practice of Constraint Programming* (pp. 6-20). Springer, Berlin, Heidelberg.
- Delgado, A., Jensen, R. M., Janstrup, K., Rose, T. H., & Andersen, K. H. (2012). A constraint programming model for fast optimal stowage of container vessel bays. *European Journal of Operational Research*, 220(1), 251-261.
- Derrett, D. (1999). R: Ship stability for Master and Mates.
- Ding, D., & Chou, M. C. (2015). Stowage planning for container ships: a heuristic algorithm to reduce the number of shifts. *European Journal of Operational Research*, 246(1), 242-249.
- Dorigo, M., & Blum, C. (2005). Ant colony optimization theory: A survey. *Theoretical computer science*, 344(2-3), 243-278.
- Douma, A., Schutten, M., & Schuur, P. (2009). Waiting profiles: An efficient protocol for enabling distributed planning of container barge rotations along terminals in the port of Rotterdam. *Transportation Research Part C: Emerging Technologies*, 17(2), 133-148.
- Dubrovsky, O., Levitin, G., & Penn, M. (2002). A genetic algorithm with a compact solution encoding for the container ship stowage problem. *Journal of Heuristics*, 8(6), 585-599.
- Dyckhoff, H. (1990). A typology of cutting and packing problems. *European Journal of Operational Research*, 44(2), 145-159.
- ECMT (European Conference of Ministers of Transport), 2006, *Inland Waterways and Environmental Protection*, OCDE, Paris.
- El Hayek, J. (2006). *Le problème de bin-packing en deux-dimensions, le cas non-orienté: résolution approchée et bornes inférieures* (Doctoral dissertation, Université de Technologie de Compiègne).

- El Yaagoubi, A., Alaoui, A. E. H., & Boukachour, J. (2018). The Shifting Traveling Salesman Problem: From Modeling to Resolution. *Journal of Engineering and Applied Sciences*, 13: 5911-5925. DOI: 10.3923/jeasci.2018.5911.5925.
URL: <http://medwelljournals.com/abstract/?doi=jeasci.2018.5911.5925>
- El Yaagoubi, A., Alaoui, A. E. H., & Boukachour, J. (2016, May). Multiobjective container barge transport problem. In *IEEE 3rd International Conference on Logistics Operations Management (GOL)*, 2016 (pp. 1-6).
- Erdogan, G., Battarra, M., Laporte, G., & Vigo, D. (2012). Metaheuristics for the traveling salesman problem with pickups, deliveries and handling costs. *Computers & Operations Research*, 39(5), 1074-1086.
- Erdogan, G., Cordeau, J. F., & Laporte, G. (2009). The pickup and delivery traveling salesman problem with first-in-first-out loading. *Computers & Operations Research*, 36(6), 1800-1808.
- Expósito-Izquierdo, C., Lalla-Ruiz, E., De Armas, J., Melián-Batista, B., & Moreno-Vega, J. M. (2017). Maritime Container Terminal Problems. *Handbook of Heuristics*, 1-27.
- Fan, L., Low, M. Y. H., Ying, H. S., Jing, H. W., Min, Z., & Aye, W. C. (2010, March). Stowage planning of large containership with tradeoff between crane workload balance and ship stability. In *Proceedings of the International MultiConference of Engineers and Computers Scientists* (Vol. 3, pp. 1-7).
- Frémont, A., & Franc, P. (2008). VOIES NAVIGABLES ET DESSERTE PORTUAIRE Massifier les flux pour intégrer le transport fluvial dans les chaînes logistiques portuaires: étude des impacts économiques et environnementaux. Le cas du transport fluvial conteneurisé.
- Frémont, A., Franc, P., & Slack, B. (2009). Inland barge services and container transport: the case of the ports of Le Havre and Marseille in the European context. *Cybergeo: European Journal of Geography*.
- Garey, M. R., et Johnson, D. S. (1979). Computers and intractability: a guide to the theory of NP-completeness. San Francisco, LA: Freeman, 58.
- George, J. A., & Robinson, D. F. (1980). A heuristic for packing boxes into a container. *Computers & Operations Research*, 7(3), 147-156.
- Goldberg, D. E. (1989). Genetic algorithm. *Search, Optimization and Machine Learning*, 343-349.
- Gomez, J. C., & Terashima-Marín, H. (2018). Evolutionary hyper-heuristics for tackling bi-objective 2d bin packing problems. *Genetic Programming and Evolvable Machines*, 19(1-2), 151-181.
- Government of Gujarat. (10-13 January 2017). Handling containerized barges at the anchorage ports in Gujarat, 8th Global Summit.
- Guan, Y., & Cheung, R. K. (2004). The berth allocation problem: models and solution methods. *OR spectrum*, 26(1), 75-92.
- Guan, Y., Yang, K. H., & Zhou, Z. (2013). The crane scheduling problem: models and solution approaches. *Annals of Operations Research*, 203(1), 119-139.
- Günther, H. O., & Kim, K. H. (2006). Container terminals and terminal operations. *OR Spectrum*. 28(4), 437-445. <https://doi.org/10.1007/s00291-006-0059-y>
- Hakan Akyüz, M., & Lee, C. Y. (2014). A mathematical formulation and efficient heuristics for the dynamic container relocation problem. *Naval Research Logistics (NRL)*, 61(2), 101-118.
- Hamed, M. (2011). *Containership Load Planning with Crane Operations* (Doctoral dissertation).
- Haulk, C. J. (1998). Inland Waterways as Vital National Infrastructure: Refuting "corporate Welfare" Attacks. Allegheny Institute for Public Policy, Pittsburgh, Pennsylvania. Report No. 97-04, p. 24.

- Hifi, M., Negre, S., & Wu, L. (2014). Hybrid greedy heuristics based on linear programming for the three-dimensional single bin-size bin-packing problem. *International Transactions in Operational Research*, 21(1), 59-79.
- Hong, S., Zhang, D., Lau, H. C., Zeng, X., & Si, Y. W. (2014). A hybrid heuristic algorithm for the 2D variable-sized bin-packing problem. *European Journal of Operational Research*, 238(1), 95-103.
- Hu, M., & Cai, W. (2017, April). Multi-objective optimization based on improved genetic algorithm for containership stowage on full route. In *Industrial Engineering and Applications (ICIEA), 2017 4th International Conference on* (pp. 224-228). IEEE.
- Ichoua, S., Gendreau, M., & Potvin, J. Y. (2003). Vehicle dispatching with time-dependent travel times. *European journal of operational research*, 144(2), 379-396.
- Imai, A., Nishimura, E., Papadimitriou, S., & Sasaki, K. (2002). The containership loading problem. *International Journal of Maritime Economics*, 4(2), 126-148.
- Imai, A., Sasaki, K., Nishimura, E., & Papadimitriou, S. (2006). Multi-objective simultaneous stowage and load planning for a container ship with container rehandle in yard stacks. *European Journal of Operational Research*, 171(2), 373-389.
- Iori, M., & Martello, S. (2010). Routing problems with loading constraints. *Top*, 18(1), 4-27.
- Kang, J. G., & Kim, Y. D. (2002). Stowage planning in maritime container transportation. *Journal of the Operational Research Society*, 53(4), 415-426.
- Kantorovich, L. V. (1960). Mathematical methods of organizing and planning production. *Management science*, 6(4), 366-422.
- Kim, K. H., & Park, Y. M. (2004). A crane scheduling method for port container terminals. *European Journal of operational research*, 156(3), 752-768.
- Li, J., Zhang, Y., & Ji, S. (2017, August). A two-phase approach for inland container ship stowage on full route on Yangtze River. In *Transportation Information and Safety (ICTIS), 2017 4th International Conference on* (pp. 799-805). IEEE.
- Louis, B. (29 novembre 2016). Le Transport Par Barge Comme Alternative Au Transport Intermodal, Nouveau Monde Maritime. URL <http://nm-maritime.com/fr/le-transport-par-barge-comme-alternative-au-transport-intermodal/>
- Mahvash, B., Awasthi, A., & Chauhan, S. (2018). A column generation-based heuristic for the three-dimensional bin packing problem with rotation. *Journal of the Operational Research Society*, 69(1), 78-90.
- Martello, S., Pisinger, D., & Vigo, D. (2000). The three-dimensional bin-packing problem. *Operations Research*, 48(2), 256-267.
- Mihyeon Jeon, C., & Amekudzi, A. (2005). Addressing sustainability in transportation systems: definitions, indicators, and metrics. *Journal of infrastructure systems*, 11(1), 31-50.
- Monaco, M. F., Sammarra, M., & Sorrentino, G. (2014). The terminal-oriented ship stowage planning problem. *European Journal of Operational Research*, 239(1), 256-265.
- Oudani, M. (2016). *Optimisation des problèmes de transport multimodal* (Doctoral dissertation, Le Havre).
- Pacino, D. (June 2012). Fast Generation of Container Vessel Stowage Plans – Using mixed integer programming for optimal master planning and constraint based local search for slot planning. Ph. D. thesis, IT University of Copenhagen.

- Pacino, D., & Jensen, R. M. (2012). Fast Generation of Container Vessel Stowage Plans: using mixed integer programming for optimal master planning and constraint based local search for slot planning.
- Pacino, D., Delgado, A., Jensen, R. M., & Bebbington, T. (2011, September). Fast generation of near-optimal plans for eco-efficient stowage of large container vessels. In *International Conference on Computational Logistics* (pp. 286-301). Springer, Berlin, Heidelberg.
- Pacino, D., Delgado, A., Jensen, R. M., & Bebbington, T. (2012, September). An accurate model for seaworthy container vessel stowage planning with ballast tanks. In *International Conference on Computational Logistics* (pp. 17-32). Springer, Berlin, Heidelberg.
- Pacino, D., Jensen, R. M., & Bebbington, T. (2012). Constraint-based local search for container stowage slot planning. In *Proceedings of the International MultiConference of Engineers and Computer Scientists* (Vol. 2).
- Paquay, C., Limbourg, S., & Schyns, M. (2018). A tailored two-phase constructive heuristic for the three-dimensional Multiple Bin Size Bin Packing Problem with transportation constraints. *European Journal of Operational Research*, 267(1), 52-64.
- Parreño, F., Pacino, D., & Alvarez-Valdes, R. (2016). A GRASP algorithm for the container stowage slot planning problem. *Transportation Research Part E: Logistics and Transportation Review*, 94, 141-157.
- Petersen, H. L., & Madsen, O. B. (2009). The double travelling salesman problem with multiple stacks—Formulation and heuristic solution approaches. *European Journal of Operational Research*, 198(1), 139-147.
- Pielage, B. J., Konings, R., Rijsenbrij, J., & Schuylenburg, M. V. (2007, March). Barge hub terminals: a perspective for more efficient hinterland container transport for the Port Rotterdam. In *48th Annual Transportation Research Forum, Boston University, Boston, MA* (pp. 19-27).
- Pillot, D., Guiot, B., Le Cottier, P., Perret, P., & Tassel, P. (2016, May). Exhaust emissions from in-service inland waterways vessels. In *TAP 2016, 21st International Transport and Air Pollution Conference* (Vol. 6, No. 4, pp. pp-205). Scienpress Ltd.
- Quiroz-Castellanos, M., Cruz-Reyes, L., Torres-Jimenez, J., Gómez, C., Huacuja, H. J. F., & Alvim, A. C. (2015). A grouping genetic algorithm with controlled gene transmission for the bin-packing problem. *Computers & Operations Research*, 55, 52-64.
- Rodrigue, J. P., Comtois, C., & Slack, B. (2017). The Geography of Transport Systems, Hofstra University, Department of Global Studies & Geography. <http://people.hofstra.edu/geotrans>
- Scheithauer, G. (2018). Two-Dimensional Bin Packing. In *Introduction to Cutting and Packing Optimization* (pp. 227-244). Springer, Cham.
- Sciomachen, A., & Tanfani, E. (2003). The master bay plan problem: a solution method based on its connection to the three-dimensional bin-packing problem. *IMA Journal of Management Mathematics*, 14(3), 251-269.
- Sciomachen, A., & Tanfani, E. (2007). A 3D-BPP approach for optimising stowage plans and terminal productivity. *European Journal of Operational Research*, 183(3), 1433-1446.
- Shields, J. J. (1984). Containership stowage: A computer-aided preplanning system.
- Srinivas, N., & Deb, K. (1994). Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary computation*, 2(3), 221-248.
- Stahlbock, R., & Voß, S. (2008). Operations research at container terminals: a literature update. *OR spectrum*, 30(1), 1-52.

- Steenken, D., Voß, S., & Stahlbock, R. (2004). Container terminal operation and operations research-a classification and literature review. *OR spectrum*, 26(1), 3-49.
- Swamidass, P. M. (Ed.). (2000). *Encyclopedia of production and manufacturing management*. Springer Science & Business Media.
- Tagmouti, M., Gendreau, M., & Potvin, J. Y. (2007). Arc routing problems with time-dependent service costs. *European Journal of Operational Research*, 181(1), 30-39.
- Tagmouti, M., Gendreau, M., & Potvin, J. Y. (2011). A dynamic capacitated arc routing problem with time-dependent service costs. *Transportation Research Part C: Emerging Technologies*, 19(1), 20-28.
- Tan, K. C., Lee, T. H., & Khor, E. F. (2002). Evolutionary algorithms for multi-objective optimization: Performance assessments and comparisons. *Artificial intelligence review*, 17(4), 251-290.
- Tanaka, S., & Tierney, K. (2018). Solving real-world sized container pre-marshalling problems with an iterative deepening branch-and-bound algorithm. *European Journal of Operational Research*, 264(1), 165-180.
- Tavares de Azevedo, A., Cassilda Maria, R., De Sena, G. J., Chaves, A. A., Neto, L. L. S., & Moretti, A. C. (2014). Solving the 3D container ship loading planning problem by representation by rules and meta-heuristics. *International Journal of Data Analysis Techniques and Strategies*, 6(3), 228-260.
- Vacca, I., Bierlaire, M., & Salani, M. (2007). Optimization at container terminals: status, trends and perspectives. In *Swiss Transport Research Conference* (No. TRANSP-OR-CONF-2006-013).
- Van Hassel, E. (2011). *Developing a small barge convoy system to reactivate the use of the small inland waterway network* (Doctoral dissertation, Universiteit Antwerpen (Belgium)).
- Van Slobbe, R. (2002, April). Larger volumes, increasing scale, environmental aspects and road congestion require intermodal solutions. In *unpublished presentation, ITMMAPS Conference, Antwerp*.
- Vander Wiel, R. J., & Sahinidis, N. V. (1996). An exact solution approach for the time-dependent traveling-salesman problem. *Naval Research Logistics (NRL)*, 43(6), 797-820.
- Veenstra, M., Cherkesly, M., Desaulniers, G., & Laporte, G. (2017b). The pickup and delivery problem with time windows and handling operations. *Computers & Operations Research*, 77, 127-140.
- Veenstra, M., Roodbergen, K. J., Vis, I. F., & Coelho, L. C. (2017a). The pickup and delivery traveling salesman problem with handling costs. *European Journal of Operational Research*, 257(1), 118-132.
- Verdonck, L., Caris, A., Ramaekers, K., & Janssens, G. K. (2014). Analysis of the operations of an intermodal barge terminal. *International Journal of Simulation and Process Modelling*, 9(1-2), 3-15.
- Vis, I. F., & De Koster, R. (2003). Transshipment of containers at a container terminal: An overview. *European journal of operational research*, 147(1), 1-16.
- Vrugt, J. A., & Robinson, B. A. (2007). Improved evolutionary optimization from genetically adaptive multimethod search. *Proceedings of the National Academy of Sciences*, 104(3), 708-711.
- Wang, N., Zhang, Z., & Lim, A. (2014, June). The stowage stack minimization problem with zero rehandle constraint. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* (pp. 456-465). Springer, Cham.
- Wäscher, G., Haußner, H., & Schumann, H. (2007). An improved typology of cutting and packing problems. *European journal of operational research*, 183(3), 1109-1130.
- Wei-Ying, Z., Yan, L., & Zhuo-Shang, J. (2005). Model and algorithm for container ship stowage planning based on bin-packing problem. *Journal of Marine Science and Application*, 4(3), 30-36.

- Wilson, I. D., & Roach, P. A. (1999). Principles of combinatorial optimization applied to container-ship stowage planning. *Journal of Heuristics*, 5(4), 403-418.
- Wilson, I. D., & Roach, P. A. (2000). Container stowage planning: a methodology for generating computerised solutions. *Journal of the Operational Research Society*, 51(11), 1248-1255.
- Wilson, I. D., Roach, P. A., & Ware, J. A. (2001). Container stowage pre-planning: using search to generate solutions, a case study. In *Research and Development in Intelligent Systems XVII* (pp. 349-362). Springer, London.
- Worldbank. (2016). La Banque mondiale, URL <https://data.worldbank.org>
- Zhang, E., Mei, Q., Liu, M., & Zheng, F. (2018). Stowage Planning in Multiple Ports with Shifting Fee Minimization. *Scientific Programming*, 2018. <https://doi.org/10.1155/2018/3450726>.
- Zhang, Z., & Lee, C. Y. (2016). Multiobjective approaches for the ship stowage planning problem considering ship stability and container rehandles. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(10), 1374-1389.