



HAL
open science

Approches multicritères pour le traitement des débris spatiaux

D. Madakat

► **To cite this version:**

D. Madakat. Approches multicritères pour le traitement des débris spatiaux. Physique de l'espace [physics.space-ph]. Université Paris Dauphine - Paris IX, 2014. Français. NNT: . tel-01059180

HAL Id: tel-01059180

<https://theses.hal.science/tel-01059180>

Submitted on 29 Aug 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ PARIS-DAUPHINE - ÉCOLE DOCTORALE DE
DAUPHINE



N° attribué par la bibliothèque

--	--	--	--	--	--	--	--	--	--

Thèse de Doctorat

Spécialité : **Informatique**

présentée et soutenue publiquement par

Dalal MADAKAT

le 16 juin 2014

Approches multicritères pour le traitement des débris spatiaux

Jury

Directeur de thèse :

Daniel VANDERPOOTEN, Professeur

LAMSADE, Université Paris-Dauphine

Co-directeur :

Jérôme MORIO, Ingénieur de recherche, HDR

Onera, The French aerospace lab

Rapporteurs :

José R. FIGUEIRA, Professeur

Université technique de Lisbonne

Patrice PERNY, Professeur

LIP6, Université Paris-VI

Examineurs :

M.Ali ALOULOU, Maître de conférences

LAMSADE, Université Paris-Dauphine

André ROSSI, Maître de conférences, HDR

Lab-STICC, Université de Bretagne-Sud



Table des matières

Table des matières	1
1 Motivations : nécessité du retrait des débris spatiaux	13
1.1 Population des débris spatiaux et son évolution	14
1.2 Aspects généraux sur les collisions débris-satellites	17
1.2.1 Risques liés aux débris spatiaux	17
1.2.2 Historique des collisions avérées	17
1.3 Solutions potentielles pour réduire les risques de collisions et leurs conséquences	19
1.3.1 Manœuvrer le satellite	19
1.3.2 Blindage du satellite	19
1.3.3 Retrait de débris	20
1.3.4 Solutions techniques de retrait des débris	20
1.4 Apport de l'aide à la décision au domaine spatial	21
Conclusion	22
2 Catégorisation des débris d'orbites basses	25
2.1 Objectifs de la classification	26
2.2 Construction des critères du choix	27
2.2.1 Critères utilisés dans la littérature	27
2.2.2 Aspects à prendre en compte lors de la catégorisation des débris	28
2.2.3 Présentation de la famille de critères utilisée	29
2.3 Mode de construction de g_1 et g_2	30
2.3.1 Sources des données	30
2.3.2 Estimation de la probabilité de collision entre débris et satellites	31
2.3.3 Evaluation des positions des débris par rapport aux zones encombrées	33
2.4 Procédure de catégorisation des débris	35
2.5 Sélection des débris selon la capacité de traitement	41
2.5.1 Utilisation d'une seule navette de désorbitation	41
2.5.2 Utilisation de plusieurs navettes de désorbitation	44
Conclusion	47
3 Présentation du problème de retrait des débris spatiaux	49
3.1 Planification d'une mission de retrait de débris spatiaux	50

3.1.1	Réalisation d'un rendez-vous spatial entre la navette et un débris	51
3.1.2	Calcul du coût d'un transfert de Lambert	51
3.2	Notations	52
3.3	Formulation du problème	53
3.4	État de l'art sur le problème de voyageur de commerce biobjectif et dépendant du temps	55
	Conclusion	59
4	Développement d'une procédure de résolution pour le problème BiTDTSP	61
4.1	Optimisation multiobjectifs	62
4.1.1	Introduction	62
4.1.2	Procédures par séparation et évaluation multiobjectifs	64
4.2	Description de la méthode	69
4.2.1	Arborescence de recherche	70
4.2.2	Adaptation de la méthode	70
	Conclusion	78
5	Implémentation et expérimentation de la méthode	79
5.1	L'algorithme par séparation et évaluation	80
5.1.1	L'algorithme principal	80
5.1.2	Stockage des solutions partielles	81
5.2	Expérimentation et résultats	84
5.2.1	L'impact de l'ordre des solutions partielles et du numérotage de débris sur le temps de calcul	85
5.2.2	L'impact des relations de dominance	86
5.2.3	L'impact du pas de temps	86
5.3	Etude de quelques frontières de Pareto	87
5.3.1	Qualité des solutions du pool	87
5.3.2	L'impact de l'ordre de visite	90
	Conclusion	98
	Bibliographie	101

Liste de Figures

1.1	Progression du nombre d'objets spatiaux d'origine humaine entre 1956 et 2010	14
1.2	Augmentation de la taille de la population spatiale suite aux deux chocs entre 2007 et 2010	15
1.3	Progression du nombre des débris de collisions et d'explosions	15
1.4	Encombrement des orbites basses par les débris spatiaux	16
1.5	Encombrement des orbites géostationnaires par les débris spatiaux	16
1.6	Collision entre le satellite français Cerise et un débris	18
2.1	Graphe correspondant à l'ordre induit par la relation R_1	37
2.2	Graphe correspondant à l'ordre induit par la relation $R_1 \cup R_2$	38
2.3	Graphe correspondant à l'ordre induit par la relation $R = \bigcup_{i=1}^3 R_i$	40
2.4	Inclinaison d'une orbite par rapport au plan de l'écliptique	42
3.1	Tranfert entre orbite 1 et orbite 2 effectué selon la méthode de Lambert.	52
3.2	Progression du graphe de la population des objets étudiés	54
3.3	Les possibilités de transfert si la navette atteint l'orbite i à t_i	55
4.1	Ensemble des solutions réalisables dans l'espace des décisions et celui des critères	63
4.2	Utilisation du point idéal	66
4.3	Bornes multicritères.	67
4.4	Comparaison entre sous-problèmes.	68
4.5	La variation du coût de transfert en fonction de sa durée.	72
4.6	Illustration de l'hypervolume associé à une frontière de Pareto.	74
5.1	L'insertion d'une nouvelle solution dans la liste de travail	82
5.2	L'insertion d'une nouvelle solution dans la liste de travail	84
5.3	L'insertion d'une nouvelle solution dans la liste de travail	85
5.4	Solutions de pool et vecteurs non dominés obtenus avec plusieurs instances	89
5.5	Orbites des débris de l'instance	90
5.6	Ordres minimisant le changement d'inclinaison et d'altitude	92
5.7	Ordres minimisant le changement d'inclinaison	93
5.8	Ordres réalisant un compromis entre le changement d'altitude et d'inclinaison	94

5.9 Ordres réalisant un compromis entre le changement d'altitude et d'inclinaison	95
5.10 Ordres maximisant le changement d'inclinaison	96
5.11 Ordres maximisant le changement d'inclinaison	97

Résumé

Les débris spatiaux constituent une menace pour l'exploration et l'exploitation de l'espace. Leur nombre ne cesse d'augmenter et continuera à grandir même si on arrête toute activité spatiale, augmentant ainsi la probabilité d'entrer en collision avec un satellite actif. Le retrait des débris s'avère le seul moyen de protéger ces satellites. Le nombre des débris spatiaux étant très élevé, il convient préalablement de repérer les plus dangereux.

Dans la première partie de la thèse, nous avons élaboré une approche multicritère afin de classer les débris selon leur degré de priorité d'enlèvement. Les débris de la classe la plus prioritaire, feront l'objet d'une mission spatiale de retrait de débris. La planification d'une telle mission est étudiée dans la deuxième partie de la thèse. Elle doit être réalisée en minimisant deux critères : le coût de la mission ainsi que la durée nécessaire pour traiter tous les débris.

La navette se déplace d'une orbite à une autre, traite les débris un par un puis retourne à son orbite initiale. Etant donné que le transfert entre deux orbites de débris peut être effectué de multiples façons, chacune correspondant à un compromis possible entre la durée et le coût de transfert, et que ces coûts et durées dépendent des moments de départ et d'arrivée sur les orbites, l'ensemble des solutions réalisables est défini sur un multigraphe dynamique orienté. Un tour dans un tel graphe définit un scénario de mission possible. Il s'agit de trouver l'ensemble des tours non dominés dans un tel multigraphe. Ceci revient à résoudre un problème de voyageur de commerce biobjectif et dépendant du temps.

Nous avons développé un algorithme basé sur la technique de séparation et évaluation pour restituer l'ensemble de ces tours. L'optimisation de l'algorithme est faite sur deux niveaux :

- On limite le nombre des transferts possibles entre deux orbites en évitant de calculer le coût pour les transferts qui s'avèreraient dominés.
- Des règles de dominance sont utilisées pour couper certaines branches de l'arborescence de recherche qui ne mèneront pas à des solutions efficaces.

Des résultats expérimentaux illustrent l'efficacité de la procédure.

Mots-clés : Débris spatiaux, catégorisation, planification.

Abstract

Space debris are a threat for the space exploitation and exploration. Their number will continue to increase even if we stop all space activities, making collisions between debris and operational satellites more likely to happen. Debris removal proves necessary to protect active satellites. Since the number of space debris is very high, we should first deal with the most dangerous ones.

In the first part of this thesis, we have developed a multicriteria approach to categorize debris depending on their removal priority degree. Debris belonging to the most urgent category will be dealt with during a space mission. The planning of such a space mission is studied in the second part of this thesis. The planning should be designed while optimizing two criteria : mission cost and mission duration.

The space shuttle moves from an orbit to another, deals with the debris one by one and then returns to its initial orbit. There are several ways to perform a transfer between two orbits. Each transfer possibility corresponds to a possible compromise between cost and transfer duration. Transfer costs and durations depend on the moment the shuttle reaches and leaves each orbit. The set of feasible solutions is thus defined on a dynamic digraph. Each tour on such a graph defines a possible mission scenario. The objective is to find the set of non dominated tours, which can be done by solving a time dependent biobjective traveling salesman problem.

We have developed a branch and bound algorithm to deal with this problem. The algorithm optimization is done on two levels :

- We limit the number of transfer possibilities between a pair of orbits to avoid computing costs of some transfers that will be dominated.
- Fathoming rules are used to cut some branches that will not lead to efficient solutions.

Experimental results illustrates the approach efficiency.

Key words : Space debris, categorization, space missions planning.

Remerciements

Je souhaite tout d'abord remercier Daniel de m'avoir encadrée et orientée pendant toute ma thèse. Daniel merci de ta patience, ta disponibilité, ta rigueur, tes idées et tes conseils.

Je remercie également Jérôme de sa disponibilité, son soutien et ses conseils.

Je remercie tous les membres du jury. Un grand merci à Mr Figueira et Mr Perny pour avoir relu et commenté mon manuscrit et à Mr Aloulou et Mr Rossi pour avoir accepté d'être présents à ma soutenance et de commenter mon travail.

Au cours de cette thèse, j'ai échangé avec un grand nombre de chercheurs notamment à l'Onera. Je tiens à remercier Hélène, tu es une mine de culture ! Nos discussions dépassaient largement le cadre professionnel !

Rudy, Rata, Christelle et Damien, merci de votre joie de vivre. Votre bonne humeur et optimisme me donnaient inspiration et motivation. Nos discussions rendaient la science plus belle à mes yeux ! Walid, Joseph et Mathieu merci de votre présence, grâce à votre générosité d'âme mes journées à l'Onera étaient plus agréables. Thierry et François merci de m'avoir fait profiter de votre expérience dans cette vie.

Ma présence au Lamsade a été très enrichissante, aussi bien sur le plan personnel que professionnel. Je tiens à remercier Renaud pour ses conseils, Lydia pour son soutien, son écoute et son amour inconditionné. Amine, tes conseils m'ont toujours été d'une grande aide. Tom, Nico et Nathanael discuter avec vous a toujours été un vrai plaisir très enrichissant. Raphael, Florian et Marek votre présence faisait du lamsade un endroit plus sympathique.

Abdallah, Edouard et Emeric, les soirées qu'on a passées ensemble m'ont fait découvrir les merveilleuses personnes que vous êtes !

Au cours de cette thèse, de nombreuses personnes m'ont entourée de leurs attentions bienveillantes, Maman, Papa, Salma et Meryem, merci d'avoir toujours été ce que vous êtes et merci d'avoir toujours été là pour moi ! Je vous aime plus que tout ! Ahlame, merci de m'avoir épaulée à chaque fois que j'en avais besoin !

Je tiens à remercier en vrac Nahid, Yasmina, Saphia, Nezha, Aida, Bassem, Soumaya, Mehdi, Adil de leur amitié et bienveillance.

Je remercie tous les autres amis, parents, proches, camarades, ou collègues avec qui j'ai passé du temps pendant ma thèse, et dont la liste serait trop longue pour figurer extensivement ici.

Introduction

Depuis le début de la deuxième moitié du siècle dernier, les activités spatiales n'ont pas cessé, mettant en orbite, par exemple, des satellites ou des stations spatiales. Ces activités ont bien sûr généré de nombreux débris, tels que des étages de lanceurs mais aussi des satellites devenus inactifs, qui orbitent également autour de la Terre. Ces débris sont problématiques pour les satellites actuellement lancés car ils constituent l'essentiel de la population des objets orbitant autour de la Terre. En effet, une collision entre un débris de plus de 1 cm de diamètre et un satellite actif peut potentiellement engendrer des dommages importants sur celui-ci. Par conséquent, afin d'éviter une collision avec des débris, un satellite doit manœuvrer et donc consommer de l'énergie ce qui réduit sa durée de vie.

Des mesures d'atténuation des effets des débris consistant à programmer une rentrée contrôlée des satellites après 25 ans d'exploitation ne suffisent pas pour stabiliser la taille de la population spatiale d'après différentes études récentes [42, 43, 44] et celle-ci resterait en constante augmentation même si toute activité spatiale était arrêtée. Il y a à cela plusieurs raisons. Tout d'abord, un débris de forte taille peut se scinder en plusieurs débris en explosant. La première explosion de débris en orbite haute a eu lieu en orbite géostationnaire (GEO) en juin 1967 augmentant considérablement la taille de la population spatiale [63]. Les débris peuvent également entrer en collision entre eux ou avec des satellites opérationnels, ce qui génère également de nouveaux débris. L'événement le plus récent témoignant de la situation date de mai 2013 lorsque le seul satellite équatorien en activité a cessé de fonctionner suite à une collision avec un ancien réservoir de fusée soviétique. Au milieu des années 70, Kessler et Cour-Palais furent les premiers à prévoir que, quelques décennies plus tard, les collisions seraient à l'origine de la création de nouveaux débris, plutôt que les explosions[33]. Ainsi, la population de débris spatiaux devrait continuer à croître d'abord suite aux collisions entre objets entiers puis entre fragments engendrés par des collisions antérieures. Ce phénomène est appelé syndrome de Kessler.

Il est donc nécessaire d'agir pour limiter l'impact des débris spatiaux sur les futurs lancements de satellites. Une voie possible d'action serait d'organiser une mission de retrait de débris spatiaux à l'aide d'un véhicule de transfert orbital. En effet, le retrait de 5 à 20 débris par an choisis de manière pertinente pourrait suffire à stabiliser la taille de la population des débris [44]. Nous avons essayé lors de cette thèse d'apporter des éléments de réponse aux questions relatives au retrait

des débris spatiaux en se basant sur des approches d'aide multicritère à la décision.

Parmi toutes les questions que peut susciter ce type de problèmes, nous avons choisi durant cette thèse de répondre principalement à deux questions. Ces deux questions et les réponses que nous avons essayé d'y apporter constituent chacune une partie de ce manuscrit.

Quels sont les débris à retirer de manière prioritaire si l'on souhaite protéger une constellation de satellites ?

Le critère le plus souvent utilisé pour déterminer si un débris est prioritaire pour un enlèvement est le produit de la masse du débris par sa probabilité de causer une collision [43, 44]. Plus ce critère est élevé, plus le débris doit être retiré au plus tôt. Néanmoins, ce critère ne permet pas de prendre en considération les différents aspects traduisant la priorité d'enlèvement d'un débris, et autorise la compensation entre la probabilité de collision et la masse du débris. Nous avons donc proposé une approche multicritère originale pour choisir les débris qui feront l'objet d'une première mission de retrait. Cela consiste à analyser les différents aspects à prendre en considération afin de juger de la dangerosité d'un débris. Ces aspects vont être utilisés pour construire la famille de critères que l'on utilisera pour définir une catégorisation de l'ensemble des débris. Les débris de la première classe devront faire l'objet d'une première mission spatiale de retrait.

Comment optimiser la planification d'une mission de retrait de débris spatiaux ?

Peu de travaux ont traité le problème de l'optimisation du retrait des débris spatiaux, l'étude réalisée par Cerf [16] étant quasiment la seule traitant ce problème. Cependant, un problème voisin, le problème de visite et maintenance des satellites a fait l'objet de plusieurs études et notamment Wadsley et al. [81] et Stodgell et al. [73]. Ainsi Wadsley et al. ont étudié ce problème en considérant comme critère unique le coût de la mission. Une instance de 20 satellites a été résolue en utilisant une approche de programmation dynamique. Le même problème a été étudié également par Stodgell et al. en prenant en considération le critère temps. Les auteurs sont capables d'approcher la frontière des vecteurs non dominés pour une instance de 6 satellites en utilisant un algorithme génétique multiobjectif.

Durant cette thèse, nous avons cherché à proposer des approches visant à concilier à la fois les performances des méthodes approchées et celles des méthodes exactes. Pour ce faire, nous nous sommes intéressés à concevoir des algorithmes exacts efficaces qui peuvent se transposer en méthodes approchées. Ainsi, nous nous sommes tournés principalement vers des approches de type "branch and bound", une méthode de résolution exacte qui est à la base de nombreuses méthodes d'approximation. La méthode de branch and bound (procédure par séparation et évaluation) consiste à énumérer les solutions de manière intelligente en ce sens que, en utilisant certaines propriétés du problème en question, cette technique

arrive à éliminer des solutions partielles qui ne mènent pas à la solution que l'on recherche. De ce fait, on arrive à obtenir la solution recherchée en des temps raisonnables. Les résultats de cette thèse concernent l'optimisation des algorithmes de branch and bound en caractérisant des relations de dominance entre solutions partielles d'une part et entre solutions partielles et solutions complètes d'autre part. Ce cadre permet de proposer des planifications de missions spatiales permettant de désorbiter jusqu'à 10 débris. Ces planifications correspondent à des compromis entre les coûts des missions et leurs durées.

Ce travail s'organise en deux parties. La première partie traite la problématique de choix des débris. Nous commençons par expliquer plus en détail le besoin du retrait des débris. Nous construisons ensuite une famille de critères avant de détailler la méthode de choix et les résultats de son application sur l'ensemble des débris LEO.

La deuxième partie est dédiée à la planification de ce genre de missions spatiales. Nous commencerons par décrire le déroulement de la mission, ensuite nous proposerons une modélisation de la situation qui permet de prendre en compte le caractère elliptique des orbites et qui retient différentes manières d'effectuer les transferts entre deux orbites différentes en optimisant les deux critères coût et durée de la mission. Les algorithmes utilisés et les différentes techniques d'optimisation sont ensuite présentés. Les résultats d'application de l'algorithme sur des instances de débris LEO sont donnés en dernier lieu.

Nous résumons nos contributions et présenterons quelques pistes de recherche dans une conclusion générale.

Chapitre 1

Motivations : nécessité du retrait des débris spatiaux

Sommaire

1.1	Population des débris spatiaux et son évolution	14
1.2	Aspects généraux sur les collisions débris-satellites	17
1.2.1	Risques liés aux débris spatiaux	17
1.2.2	Historique des collisions avérées	17
1.3	Solutions potentielles pour réduire les risques de collisions et leurs conséquences	19
1.3.1	Manœuvrer le satellite	19
1.3.2	Blindage du satellite	19
1.3.3	Retrait de débris	20
1.3.4	Solutions techniques de retrait des débris	20
1.4	Apport de l'aide à la décision au domaine spatial	21
	Conclusion	22

Depuis la deuxième moitié du siècle dernier, plusieurs lancements ont été effectués mettant en orbite plusieurs satellites. Aujourd'hui, la grande majorité de ces satellites n'est plus opérationnelle et constitue désormais des débris spatiaux. De plus, ces satellites inactifs finissent par exploser en orbite ou entrent en collision avec d'autres objets spatiaux générant ainsi de nouveaux débris. L'encombrement de l'espace en débris est problématique car si l'on continue ainsi, aucune activité spatiale ne sera possible d'ici quelques années. Par ailleurs, les débris spatiaux peuvent entrer en collision avec des satellites opérationnels à tout moment. Depuis le milieu des années 90, des mesures pour réduire le nombre des débris spatiaux ont été implémentées. Néanmoins ces mesures s'avèrent insuffisantes pour stabiliser la taille de la population spatiale. Nous expliquons dans ce chapitre introductif les risques liés aux débris spatiaux et la nécessité de leur enlèvement, nous y présentons aussi les différents traitements que l'on peut faire subir aux débris.

1.1 Population des débris spatiaux et son évolution

L'espace entourant la Terre est très encombré par des milliers d'objets d'origine humaine, et ces objets ne sont pas tous actifs aujourd'hui. En effet, entre 1957 et 2008, près de 4600 lancements ont été effectués mettant en orbite 6000 satellites. Seuls 800 de ces satellites sont actuellement opérationnels. Les satellites restant constituent désormais des débris spatiaux. S'ajoutent à ces satellites inactifs, les débris de plus petites tailles causés par les explosions, les fragmentations et les collisions pour former la population des milliers d'objets spatiaux. La figure 1.1 montre la progression des objets spatiaux catalogués entre 1956 et 2010. On peut noter la forte présence des débris dûs à la fragmentation comparée à celle des corps de fusées ou des vaisseaux spatiaux.

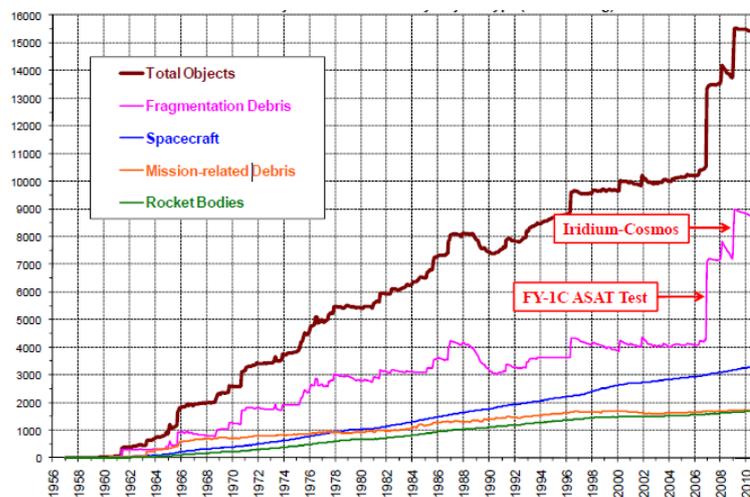


FIGURE 1.1 – Progression du nombre d'objets spatiaux d'origine humaine entre 1956 et 2010

Il est possible de décomposer la progression de la taille de la population spatiale en trois grande phases :

- 1960-1996 la progression est quasi-linéaire la population croît d'environ 260 débris par an.
- 1996-2006 La progression est toujours quasi-linéaire, probablement suite à l'implémentation des mesures pour réduire la taille de la population spatiale.
- 2006-2010 Pendant cette période, deux principaux événements ont été à l'origine de la création de plus de 1250 débris de diamètre dépassant les 10cm par an. En effet, la Chine a lancé le 11 Janvier 2007 un missile antisatellite, technologie que seuls les USA et la Russie maîtrisaient alors, pour détruire son satellite météorologique Feng Yun 1C. Cette opération a généré plus de 40 000 débris dont le diamètre dépasse 1 centimètre. Une deuxième collision catastrophique s'est produite entre les satellites Iridium-33 et Cosmos-2251

le mardi 10 février 2009 causant une augmentation de 117% du nombre d'objets en orbite dans la région d'altitude en deçà de 1000 km comme le montre la Figure 1.2.

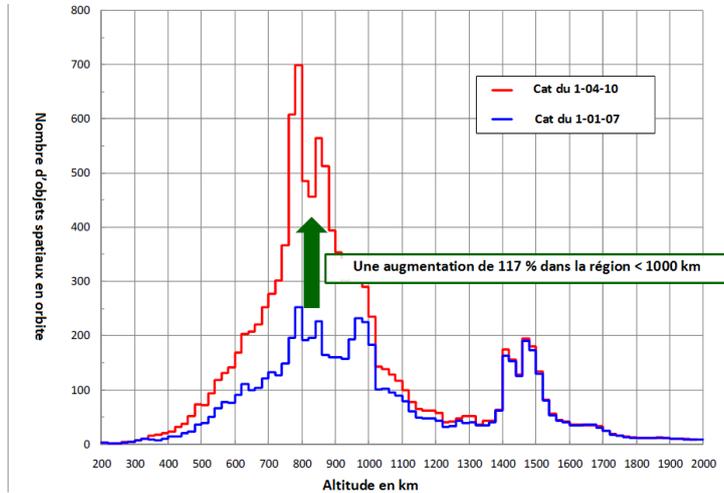


FIGURE 1.2 – Augmentation de la taille de la population spatiale suite aux deux chocs entre 2007 et 2010

Les études effectuées ultérieurement par [40] montrent que la population des débris spatiaux actuelle est essentiellement constituée des fragments dûs aux explosions. Cependant, plusieurs modèles prévoient également que les collisions seront à l'origine de la génération des débris spatiaux futurs comme le montre la Figure 1.3.

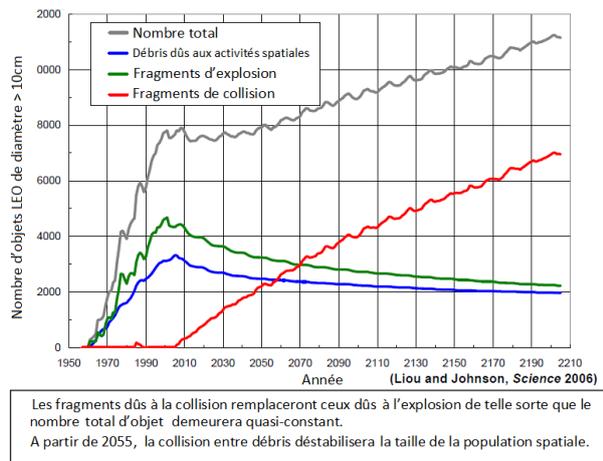


FIGURE 1.3 – Progression du nombre des débris de collisions et d'explosions

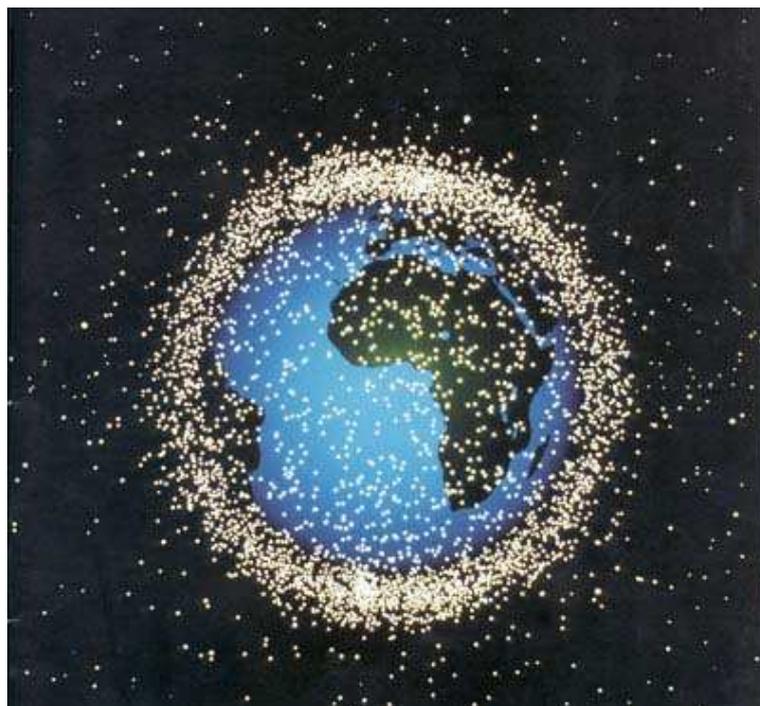


FIGURE 1.4 – Encombrement des orbites basses par les débris spatiaux

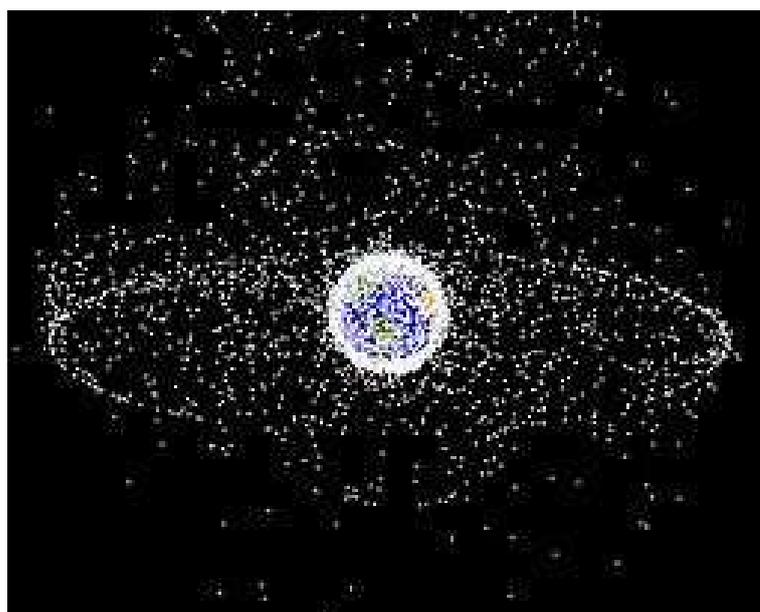


FIGURE 1.5 – Encombrement des orbites géostationnaires par les débris spatiaux

Les débris spatiaux constituent donc une des principales menaces pour l'exploration et l'exploitation de l'espace. Ils sont concentrés surtout sur les orbites proches de la Terre notamment dans la zone LEO (Low Earth Orbit) pour les objets dont l'altitude est comprise entre 200 et 2000 km et GEO (Geostationary Earth Orbit) pour les objets dont l'altitude dépasse 35000 km comme le montrent les figures 1.4 et 1.5. La plupart des collisions sont prévues dans la zone LEO et peu de collisions ont été prévues pour la zone GEO.

Ainsi, le nombre de débris augmente et par conséquent, le risque de collision avec un ou plusieurs satellites opérationnels augmente aussi. La section suivante décrit rapidement les conséquences d'une collision avec un débris pour un satellite opérationnel.

1.2 Aspects généraux sur les collisions débris-satellites

1.2.1 Risques liés aux débris spatiaux

La durée de vie en orbite est limitée par la présence de l'atmosphère terrestre même ténue. L'atmosphère va ralentir les objets et par conséquent, pour ceux qui évoluent sur des orbites basses, provoquer à long terme leur rentrée sur Terre. Pour fixer un ordre de grandeur, un objet orbitant à une altitude de 800 km a besoin de 200 ans pour rentrer en atmosphère. Le risque en orbite inclut les risques de collision avec des objets opérationnels du fait de leurs durées de vie en orbite importantes, les risques liés à la production de débris suite au vieillissement des matériaux et des impacts éventuels avec d'autres débris ou météorites, ainsi que les risques d'éventuelles explosions. Les dommages engendrés par les débris spatiaux peuvent être relativement importants même si la taille du débris est petite. Ceci est tout simplement dû à la vitesse orbitale des débris qui est très élevée (8-10 km/s) et par conséquent l'énergie cinétique n'en est que plus importante.

De ce fait, les débris d'une taille inférieure à 0.01 cm ne feront qu'éroder les surfaces de satellites opérationnels sur le long terme (effet cumulatif) générant par exemple le détachement d'écailles de peinture tandis que les débris d'une taille comprise entre 0.01 et 1 cm provoqueront des dommages significatifs comme des perforations d'équipements dont les conséquences peuvent être variables en fonction de l'équipement atteint (dysfonctionnement mineur à total de l'équipement). Ceci a été notamment observé, par exemple, sur les panneaux solaires du télescope Hubble.

Par contre, les débris d'une taille comprise entre 1 cm et 10 cm risquent d'engendrer des dommages très importants compte tenu de leur énergie cinétique. Ils présentent un réel danger du fait qu'ils ne sont pas catalogués à ce jour [82]. Les débris d'une taille supérieure à 10 cm auront des conséquences catastrophiques pour le satellite atteint pouvant aller jusqu'à sa perte voire son explosion. Nous résumons dans le tableau 1.1, les dégâts potentiellement engendrés par des débris de différentes tailles.

1.2.2 Historique des collisions avérées

A ce jour on dénombre 5 cas de collisions avérées entre objets catalogués :

- 24 juillet 1996 : la première collision répertoriée a eu lieu en 1996 entre le satellite français Cerise et un débris issu d'une explosion d'un étage supérieur

Taille des débris	Caractéristiques
< 0.01 cm	Erosion des surfaces
entre 0.01 et 1cm	Dommages significatifs Perforations Conséquences variables suivant l'équipement atteint
entre 1 et 10cm	Dommages très importants
> 10cm	Conséquences catastrophiques pour un satellite

TABLE 1.1 – Dégats générés par les débris selon de leur taille

d'Ariane (voir la Figure 1.6).

- 17 janvier 2005 : collision entre un étage lanceur américain (Thor) et un débris de lanceur chinois (CZ-4).
- En décembre 1991 (collision identifiée en 2005) : collision entre satellite de navigation russe (COSMOS 1991) et un débris de satellite russe (COSMOS 926).
- 10 février 2009 : cette collision a impliqué deux satellites, dont un actif (IRIDIUM 33), et un satellite russe inactif mais intact (COSMOS 2251). Les débris générés par cet événement sont nombreux (de l'ordre du millier) et se situent à des altitudes très peuplées (vers 800km).
- 23 mai 2013 : l'unique satellite de l'Equateur, mis en orbite le 25 avril 2013, a été heurté par les débris d'une fusée russe lancée dans l'espace en 1985, il est trop tôt pour évaluer les dégâts.

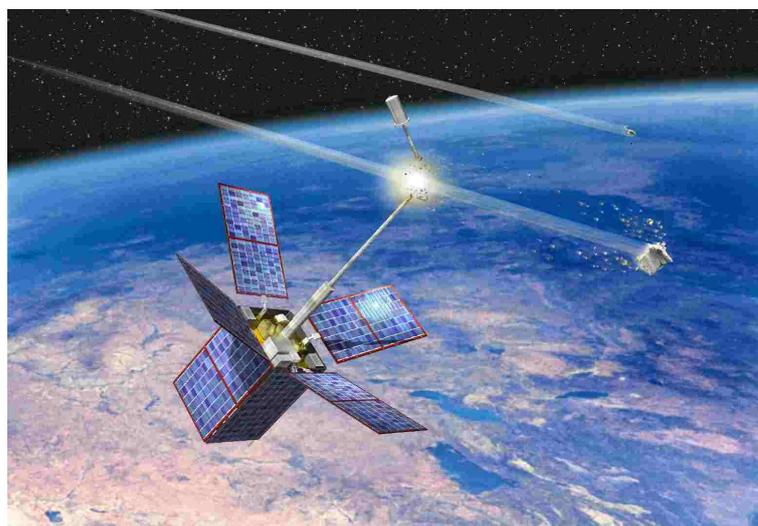


FIGURE 1.6 – Collision entre le satellite français Cerise et un débris

D'autres collisions catastrophiques pourraient survenir si des mesures ne sont pas prises pour stabiliser la taille de la population spatiale et pour protéger les satellites opérationnels.

1.3 Solutions potentielles pour réduire les risques de collisions et leurs conséquences

1.3.1 Manœuvrer le satellite

Afin d'éviter les potentiels dégâts d'une collision, plusieurs stratégies sont possibles. L'évitement est la stratégie la plus évidente. Cela consiste à réaliser des manœuvres afin d'éviter toute collision avec les débris dans le cas d'un satellite ou bien à modifier l'heure de tir du lancement de celui-ci. Cette solution ne peut être mise en œuvre que pour éviter les 15000 débris catalogués, c'est-à-dire, ceux régulièrement suivis par un système de surveillance de l'espace. Les débris catalogués ont une taille supérieure à 10 cm. Par ailleurs, les données relatives à la trajectoire des objets spatiaux sont imprécises, ce qui peut conduire à de fausses alertes. Ces évitements ne sont pas toujours faciles à réaliser et réduisent la durée de vie d'un satellite. De ce fait, cela est réservé aux lancements, et aux satellites sensibles. L'évitement ne couvre donc qu'une partie du risque (moins de 5%) et ne concerne que les objets catalogués.

1.3.2 Blindage du satellite

Des mesures de protection des satellites peuvent être mises en œuvre pour atténuer les effets d'un impact avec des débris : les blindages spécifiques ou intrinsèques. Les blindages spécifiques sont des blindages multi-couches (Kevlar ou Nextel). Ils consistent à rajouter des surfaces de protection autour de l'élément à protéger. Les blindages intrinsèques consistent à utiliser les parois du satellite comme boucliers. Une conception intelligente de l'architecture du satellite permet de protéger les équipements fragiles en les plaçant, soit sur des faces qui seront peu exposées, soit, par exemple, à l'intérieur du satellite ou derrière des équipements moins sensibles si cela est possible. Toutes ces mesures engendrent souvent un surcoût économique mais aussi en terme de masse. Ces mesures n'auront bien entendu un effet positif que si la taille des débris est inférieure à 1 cm. Cette solution ne couvre donc aussi qu'une partie du risque.

D'autres mesures sont possibles, consistant à éviter de générer de nouveaux débris et à limiter ainsi la prolifération des débris. Il s'agit par exemple de modifier la conception des systèmes spatiaux en incorporant des mesures de prévention de l'explosion [82] et de procéder à la passivation des objets spatiaux après 25 années opérationnelles. Certes, ces mesures d'atténuation sont considérées comme moins coûteuses et plus réalisables à court terme que les missions a posteriori visant le retrait des débris. Cependant, elles demeurent insuffisantes pour stabiliser la taille

de la population spatiale et pour limiter le risque de collision entre débris. Des missions de retrait effectuées a posteriori semblent s'avérer indispensables pour une utilisation permanente des orbites basses.

1.3.3 Retrait de débris

Le retrait des débris spatiaux est devenu une nécessité comme le montrent plusieurs études publiées dans la littérature. Ainsi, l'analyse effectuée par Liou dans [41] montre que la taille de la population spatiale ne cessera d'augmenter et atteindra le triple de sa taille actuelle au bout d'une centaine d'années si aucune mesure de réduction n'est prise en compte. Il souligne dans [42, 43, 44] que les mesures consistant à abaisser les orbites des satellites en fin de vie pour forcer leur rentrée en atmosphère au bout de 25 ans ne sont pas considérées comme un enlèvement de débris actifs. Procéder à une mission de retrait des débris spatiaux est une nécessité notamment dans la région LEO. Les simulations effectuées par Anselmo et al. [4] confirment l'importance de la passivation des vaisseaux spatiaux et des fusées pour éviter l'explosion en orbite, mais la désorbitation demeure indispensable afin de limiter le risque de collision et pour échapper à l'explosion exponentielle du nombre de débris spatiaux en orbites basses. Le seul remède à cette situation consiste à retirer 5 à 20 objets annuellement [44]. La date initiale idéale pour entamer la mission a été également discutée dans [42], qui estime très optimiste d'espérer avoir développé des technologies matures pour le retrait des débris spatiaux d'ici l'an 2020. Le nombre prédit de collisions si le retrait annuel de 5 débris démarre en 2060 au lieu de 2020 passe de 25 collisions à 32. Ces collisions engendreront environ 2000 débris supplémentaires dans l'environnement pendant les 200 prochaines années [42]. Afin d'arriver à un consensus entre la date de début de la mission et le nombre de collisions prévu, d'autres études doivent encore être effectuées.

1.3.4 Solutions techniques de retrait des débris

Plusieurs solutions techniques pour retirer des débris peuvent être envisagées. Ces solutions peuvent être regroupées sous deux catégories.

Solutions avec contact

Les solutions les plus fiables sont basées sur un rendez-vous entre la navette et le débris suivi d'une capture en utilisant un bras robotique. Les débris ne sont pas des cibles coopératives. Un signal de guidage doit être élaboré sans aucune information émanant de la cible. Plusieurs solutions ont été présentées telles que la technique du Flash-Lidar développé par Ball aerospace [51] ou la reconnaissance de formes optiques développée et testée par JAXA (Japan Aerospace Exploration Agency) [12]. La possibilité de capture du débris avec un bras robotique dépend fortement du mouvement résiduel du débris lors de la capture. Des techniques visant à affaiblir le

mouvement du débris en vue de le capturer facilement ont également été proposées. On peut par exemple citer la proposition de JAXA consistant à lancer des boules molles à des débris afin d'amortir leur mouvement.

Une fois le débris capturé, il doit être désorbité, trois possibilités sont à considérer :

1. Son orbite est abaissée de manière à programmer sa rentrée en atmosphère immédiatement.
2. Son orbite est abaissée de manière à ce que sa rentrée en atmosphère s'effectue dans moins de 25 ans en respectant les règles des conventions internationales.
3. On le désorbite à une hauteur supérieure à 2000 km.

Solutions sans contact

L'une des solutions les plus étudiées afin de désorbiter un débris sans contact consiste à utiliser des lasers, appliqués soit à partir de la Terre ou à partir d'une orbite. Les premières études réalisées dans ce domaines ont été publiées au début des années 90 [68].

1.4 Apport de l'aide à la décision au domaine spatial

Plusieurs travaux ont fait l'objet de l'application de l'aide à la décision au domaine spatial.

Ainsi, dans [13, 14], l'aide à la décision a été appliquée afin de partager une constellation de satellites entre différents agents (pays, entreprises, agences civiles ou militaires . . .). Le partage peut être fait de manière à satisfaire des contraintes physiques d'exploitation et en satisfaisant l'ensemble des agents en maximisant les utilités de leurs parts. Deux manières d'approcher le problème ont été présentées.

- Utiliser une fonction d'utilité collective représentant le préordre leximin, son fonctionnement est fondé sur une comparaison successive des utilités des agents, du plus pauvre au plus riche, jusqu'à trouver une différence qui permet alors de discriminer les deux profils : il s'agit donc d'une comparaison lexicographique sur les vecteurs d'utilité triés.
- Aborder le problème directement dans le cadre de l'optimisation multicritère et développer une algorithmique dédiée.

Différents algorithmes ont été introduits. Ils s'appuient sur la puissance du cadre de la programmation par contraintes pour calculer des solutions leximin-optimales de différentes manières. La plupart de ces algorithmes sont fondés sur des mécanismes de propagation de contraintes.

Le problème de prise de vue par des satellites a été également abordé dans [26]. Un satellite suit plusieurs orbites afin de prendre des vues correspondant à des

demandes de plusieurs clients. Les opérations journalières effectuées par un tel satellite seraient de déterminer une séquence de prises faisable et satisfaisant au mieux la demande. Le nombre de séquences possibles augmente significativement avec le nombre de prises demandées et l'évaluation d'une séquence se fait en se basant sur plusieurs critères conflictuels à savoir : le nombre de demande satisfaites par la séquence, le degré de priorité des demandes satisfaites et le nombre de prises effectuées qui traduit l'utilisation du satellite. Le problème est formulé comme la sélection de chemins multicritères dans un graphe sans circuits. L'approche de résolution se fait en deux étapes : la génération des chemins efficaces dans un premier temps et la sélection d'un chemin satisfaisant en utilisant une procédure multicritère interactive.

L'optimisation multicritère voit aussi ses applications dans le domaine de conception multidisciplinaire des satellites. Dans [66], Ravanbakhsh et al. développent un algorithme génétique biobjectif afin de concevoir des satellites en minimisant la masse totale du satellite tout en maximisant la proportion du budget consacré à la conception et la réalisation du satellite. D'autres études utilisent des algorithmes génétiques pour la conception de satellites, nous citons par exemple [67, 74, 56].

Très peu de travaux traitent le problème de retrait de débris. Cerf, [16] a proposé une approche basée sur un algorithme par séparation et évaluation. Son étude permet de sélectionner et désorbiter à moindre coût 5 débris parmi une population initiale de taille quelconque. Le temps a été considéré comme une contrainte dans cette approche et non pas comme un objectif à minimiser.

En revanche, certains travaux traitent de problématiques voisines. Wadsley et al. [81] ont proposé une approche permettant de planifier une mission spatiale pour la maintenance des satellites. Pour appréhender le problème, une approche exacte basée sur la programmation dynamique a été utilisée. Des instances allant jusqu'à 20 satellites sont alors résolues.

Stodgell et al. [73] abordent le problème de visite de satellites en minimisant le coût et la durée de la mission. Le problème est donc formulé comme un problème de voyageur de commerce biobjectif. Une instance comportant 6 satellites a été résolue en utilisant un algorithme génétique multiobjectif.

Conclusion du chapitre

Les débris spatiaux présentent un danger de plus en plus menaçant pour les satellites actifs. Les techniques de stabilisation utilisées jusqu'à maintenant s'avèrent insuffisantes pour stabiliser la population de débris spatiaux. Le retrait de certains débris est nécessaire pour assurer la possibilité de l'utilisation de l'espace dans le futur d'une part et la protection des satellites opérationnels d'autre part. Différentes manières de traiter les débris ont été proposées dans la littérature.

Comme les missions de retrait s'avèrent extrêmement coûteuses, on ne peut

éliminer les débris dans leur totalité. L'idée est de les catégoriser selon leur priorité à être enlevés puis planifier une mission pour leur retrait.

Chapitre 2

Catégorisation des débris d'orbites basses

Sommaire

2.1	Objectifs de la classification	26
2.2	Construction des critères du choix	27
2.2.1	Critères utilisés dans la littérature	27
2.2.2	Aspects à prendre en compte lors de la catégorisation des débris	28
2.2.3	Présentation de la famille de critères utilisée	29
2.3	Mode de construction de g_1 et g_2	30
2.3.1	Sources des données	30
2.3.2	Estimation de la probabilité de collision entre débris et satellites	31
2.3.3	Evaluation des positions des débris par rapport aux zones encombrées	33
2.4	Procédure de catégorisation des débris	35
2.5	Sélection des débris selon la capacité de traitement	41
2.5.1	Utilisation d'une seule navette de désorbitation	41
2.5.2	Utilisation de plusieurs navettes de désorbitation	44
	Conclusion	47

On se propose dans ce chapitre d'élaborer une méthode de catégorisation des débris spatiaux afin de remédier au problème de la taille de la population spatiale qui ne cesse d'augmenter. Cette méthode est conçue de telle sorte à protéger prioritairement une famille bien déterminée de satellites contre les dangers des débris spatiaux. Etant donné le nombre très élevé de débris, il importe de choisir ceux qui s'avèrent plus dangereux pour la constellation et qui sont plus urgents à éliminer. Pour ce faire, nous construisons, dans un premier temps, une famille de critères que l'on utilisera pour affecter les débris aux différentes catégories. Nous évaluerons ensuite les performances des débris sur chaque critère et nous présenterons enfin les résultats de la catégorisation.

2.1 Objectifs de la classification

Les satellites en mouvement en orbites basses sont ceux susceptibles d'être endommagés puisque les débris spatiaux évoluant dans cette région sont plus nombreux et plus rapides. On se fixe pour objectif de protéger prioritairement une constellation de satellites européens évoluant en zone LEO. Notons par D l'ensemble des débris LEO et par S la constellation de satellites européens à protéger évoluant en basses orbites. Nous allons choisir parmi les éléments de D ceux qui vont faire l'objet d'une mission de retrait de débris. Un débris est d'autant plus prioritaire à être choisi qu'il risque de causer une collision avec un satellite de la constellation. Le degré de dangerosité d'un débris est assez variable. Nous distinguons trois types de débris :

- Les débris susceptibles de présenter un danger direct pour au moins un élément de S . Ces débris ont ainsi une forte probabilité de collision avec au moins un satellite de la constellation pendant les quelques années à venir.
- Les débris ayant une forte chance de générer d'autres débris qui entreraient en collision avec les éléments de S . C'est notamment le cas des débris appartenant aux zones encombrées.
- Les débris qui ne présentent pas de danger pour la constellation. Il s'agit de débris tellement éloignés de tout autre objet qu'ils ne constituent pas de danger direct ou indirect.

Le traitement qu'un débris va subir dépend de son degré de dangerosité. Ainsi distinguons-nous trois catégories de débris :

- La première catégorie C_1 contient les débris qui présentent un danger pour la constellation, ces débris devraient être retirés le plus rapidement possible.
- La deuxième catégorie C_2 contient les débris dont le danger n'est pas négligeable mais dont le retrait n'est pas aussi urgent que ceux de la première catégorie, ces débris vont être surveillés de près et retirés dans la limite du possible.
- La troisième catégorie C_3 , contient les débris qui ne présentent aucun danger pour la constellation et qui ne nécessitent ainsi aucun traitement.

Nous construirons dans la section suivante les critères à utiliser lors de la catégorisation.

2.2 Construction des critères du choix

A notre connaissance, il n'existe pas dans littérature de travaux de catégorisation de l'ensemble des débris en utilisant plusieurs critères. Cependant, des études présentent certaines indications pour choisir les débris à éliminer. Nous présentons, ci-après ces travaux.

2.2.1 Critères utilisés dans la littérature

Liou et al. [44] ont examiné l'efficacité du retrait des débris spatiaux (ADR, Active Debris Removal) et ont conclu que le retrait des débris est indispensable. Les travaux de Liou et Johnson [42, 43] ont montré que la stabilisation de la taille de la population des objets LEO nécessite :

1. Une implémentation réussie des moyens de précautions.
2. Un retrait annuel de cinq débris bien choisis.

Tout en décrivant une stratégie de sélection des débris consistant à :

- a) Sélectionner les objets ayant une forte probabilité de collision.
- b) Sélectionner les objets qui potentiellement généreront plus de fragments en cas de collision.

Ainsi, il vaudrait mieux commencer par l'élimination des objets dont le diamètre est au moins égal à 10 cm car ils sont plus volumineux et risquent d'engendrer plus de dégâts en cas de collision. Bien que cette limite soit imposée par la capacité des senseurs à détecter les objets, elle est également justifiée par le fait que la masse des objets dont le diamètre dépasse ce seuil constitue 95% de la masse totale des objets spatiaux. Comme les zones de l'espace encombrées correspondent à de faibles excentricités, Liou suggère de restreindre l'ensemble des débris candidats au retrait aux orbites dont l'excentricité est inférieure à 0.5. Pour ce faire, Liou [42] estime qu'un simple critère agrégeant la probabilité d'entrer en collision et la criticité de la collision est suffisant pour déterminer la priorité des objets. En se basant sur ce simple argument physique, le critère de sélection proposé est le produit de la masse du débris et de sa probabilité d'entrer en collision.

Ce critère peut tout de même être critiqué. En premier lieu, le critère accorde aux débris qui risquent d'entrer en collision avec un satellite opérationnel et ceux qui risquent d'heurter un autre débris la même priorité d'enlèvement. Ceci peut être gênant puisqu'un débris qui constitue un danger direct pour un satellite devrait être enlevé prioritairement.

D'autre part, la manière d'agréger les deux aspects masse et probabilité fait qu'il est possible de compenser la masse d'un débris par sa probabilité d'entrer en collision. Le critère affecte ainsi à un débris de masse 1 *Kg* ayant une probabilité de collision de 10^{-3} la même priorité d'enlèvement qu'un autre débris de masse 1 *tonne* et de probabilité de collision 10^{-6} . Or le premier débris est nettement plus dangereux puisque sa masse est suffisante pour générer des dégâts catastrophiques

et qu'il a de fortes chances d'entrer en collision. De plus, l'adoption d'un seul critère, même s'il permet d'établir un ordre complet sur l'ensemble des débris, ne permet pas de définir un seuil à partir duquel le produit de la masse et de la probabilité correspond à un débris potentiellement dangereux.

Enfin, ce critère fait abstraction de tout aspect juridique faisant qu'un pays ne peut intervenir sur les débris d'un autre et que réaliser un accord international peut être très compliqué. Le critère ne prend pas en considération la nature du débris et traite indifféremment certains débris qui sont plus facilement manipulables que d'autres.

2.2.2 Aspects à prendre en compte lors de la catégorisation des débris

Les débris présentant un danger direct sont ceux qui risquent d'entrer en collision avec un satellite de la constellation et qui, en cas de collision, causeront des dégâts considérables. Dans le modèle de collision adopté par [4], la criticité d'une collision entre un débris et un satellite opérationnel se mesure en fonction de la valeur du rapport entre l'énergie cinétique du satellite et la masse du débris. Si ce rapport est inférieur à un seuil critique Q^* , le choc provoquera une cratérisation à la surface du satellite. Des valeurs supérieures à Q^* risquent de provoquer une fragmentation complète du satellite. Les valeurs acceptables de Q^* sont comprises dans l'intervalle [30000; 60000] J/Kg. Ces valeurs correspondent à de faibles diamètres de débris.

Etant donné que les objets catalogués dépassent tous 10 cm, tout objet catalogué s'avère très dangereux.

Un débris présentera un danger indirect pour la constellation s'il a de fortes chances d'entrer en collision avec d'autres débris. Les collisions sont naturellement plus probables dans les zones encombrées.

La région LEO qui s'étend jusqu'à une altitude de 2000 km, est la plus densément peuplée de l'espace en terme de nombre d'objets. Trois zones proches de la Terre ont été identifiées comme les plus critiques à cause de leur grande concentration en masse. Ces régions sont caractérisées par les altitudes et inclinaisons suivantes [15].

(a) altitude = 1000 km \pm 100 km à inclinaison = $82^\circ \pm 1^\circ$

(b) altitude = 800 km \pm 100 km à inclinaison = $99^\circ \pm 1^\circ$

(c) altitude = 850 km \pm 100 km à inclinaison = $71^\circ \pm 1^\circ$

Le plus grand nombre de collisions catastrophiques est prévu dans la région (a) dans les prochaines 200 années [15]. 317 objets ont été identifiés dans cette région. Ces objets ont une grande espérance de vie et excepté certains cas, appartiennent à la même constellation. Leurs masses vont de 500 kg à 1500 kg.

La région (b) correspond aux conditions héliosynchrones (une orbite héliosynchrone est une orbite autour du Soleil telle que la période de révolution d'un objet sur cette

orbite coïncide avec la période de rotation du Soleil). Cette région est importante pour les applications commerciales et scientifiques qui utilisent ce type d'orbites. 140 objets ont été identifiés dans cette région. La masse de la population de ces objets est équitablement distribuée entre les intervalles de masse 1 – 65 kg, 125 – 990 kg et 1000 – 4000 kg.

63 débris ont été identifiés dans la région (c) : principalement des corps de fusées ayant une masse importante et une grande surface de contact. Environ 20 débris ont une masse d'environ 3200 kg et les autres ont une masse dépassant les 8200 kg.

Nous différencions entre les débris qu'on peut traiter et ceux dont le traitement pose des problèmes juridiques ou techniques. C'est ainsi que, les débris de type corps de fusée sont plus simples à éliminer que les autres de par leur forme cylindrique facilement manipulable et que les débris européens peuvent être traités sans soucis contrairement aux débris non européens. Ces derniers nécessitent un accord juridique avec le pays source du débris. Un débris est affecté à la troisième catégorie s'il a une faible probabilité de collision avec l'ensemble des satellites et s'il est très éloigné des zones d'encombrement. Nous allons présenter dans la partie suivante une formalisation de la méthode d'affectation et des critères utilisés.

2.2.3 Présentation de la famille de critères utilisée

La famille de critères est construite de telle sorte qu'elle prend en considération les aspects précédemment évoqués. Quatre critères sont ainsi pris en compte, soient $g_i, i = 1, \dots, 4$, ces quatre critères.

Ces critères sont définis par :

- g_1 pour un débris $d \in D$, $g_1(d)$ représente la probabilité de collision entre d et au moins un élément de la constellation S .
- g_2 la position du débris, qui est une valeur comprise entre 0 et 1. La valeur de g_2 pour un débris appartenant à une région encombrée est égale à 1, et à 0 si le débris est très éloigné des zones encombrées. Le critère prend une valeur intermédiaire si le débris n'est pas trop éloigné des zones encombrées sans être dedans.
- g_3 , renseigne sur la nationalité du débris, il prend la valeur 1 si le débris est européen, 0 sinon.
- g_4 le dernier critère dépend du type du débris. Il prend la valeur 1 pour les corps de fusées, 0 sinon.

Notons que g_3 et g_4 sont des attributs traduisant des préférences binaires simples. Nous explicitons dans les sections suivantes le mode de construction des deux premiers critères.

2.3 Mode de construction de g_1 et g_2

2.3.1 Sources des données

Pour avoir la description des orbites des objets spatiaux étudiés, nous utilisons la source la plus complète et qui est disponible aujourd'hui au public. Cette source fournit des informations sur les éléments orbitaux appelés les two-line elements (TLEs) fournis par le NORTH American aerospace Defense command (NORAD). Le NORAD distribue des informations orbitales pour un très grand nombre d'objets spatiaux sous forme de TLEs. Un TLE est une représentation standardisée des paramètres orbitaux des objets en orbite terrestre. Ces éléments sont mesurés et calculés par le NORAD et la NASA, non seulement pour les satellites artificiels, mais aussi pour les débris spatiaux.

Ils permettent de calculer la position des objets en orbite à tout instant car ils suivent les lois de Kepler et Newton. Compte tenu des nombreuses perturbations dont ils font l'objet (influences de l'attraction de la Lune et du Soleil, freinage atmosphérique, vent solaire mais aussi, les corrections volontaires d'orbite), ces paramètres doivent cependant être régulièrement mis à jour et ne sont valables que pour une période limitée. Toutes ces données sont accessibles au grand public sur le site SPACETRACK qui est mis à jour quotidiennement.

D'autres sources de données sont à notre disposition pour évaluer les performances des débris sur l'ensemble des critères utilisés. Nous disposons ainsi du rapport de situation des satellites qui liste l'ensemble des objets qui sont actuellement en orbite et ceux qui ont un jour orbité autour de la Terre et qui sont détectables. Le rapport fournit pour chaque corps spatial, en plus de quelques éléments orbitaux, sa source, sa nature et sa date de lancement. Nous avons exploité ces deux sources afin d'évaluer les performances des débris sur la famille de critères. A tout moment, on doit être capable de connaître les positions de la navette et des débris étudiés. Pour ce faire, nous avons besoin d'un modèle de propagation des positions des différents objets étudiés. Afin d'obtenir de bonnes prédictions, l'estimation des positions doit être faite d'une manière compatible avec la manière dont les TLEs ont été générés par le NORAD [31]. Les TLEs doivent ainsi être utilisés avec le modèle de propagation : SGP4. Le SGP4 est un algorithme développé par la NASA qui est accessible au public [53, 18].

2.3.2 Estimation de la probabilité de collision entre débris et satellites

Etant donné un débris $d \in D$ et un satellite $s \in S$, la probabilité de collision entre d et s n'est pas constante par rapport au temps. Cependant, en se restreignant sur une durée d'étude, on suppose que la population est stable et que la probabilité de collision entre un satellite et un débris ne dépend que de leurs positions relatives. Cette probabilité est notée P_{col} . On considérera qu'il y a collision si la distance entre 2 objets est en dessous de la distance d_{lim} . La taille de la population des débris spatiaux d'orbites basses étant très grande, le nombre d'estimations à effectuer est par conséquent également très important. Certains débris ont des orbites très éloignées des orbites des satellites de la constellation, la distance entre leurs positions relatives demeure assez grande pour supposer qu'ils ne risquent pas de se croiser. Pour ces débris, il paraît légitime de considérer que la probabilité de collision est nulle. Pour les autres débris, une estimation plus fine doit être faite. Ainsi le travail est effectué en deux étapes. La première étape consiste à filtrer les couples (débris, satellite) ayant des probabilités de collision négligeables, ensuite il faudra affiner l'estimation de la probabilité pour les autres débris par un calcul plus précis.

Filtrage des couples ayant une probabilité de collision négligeable

Pour chaque couple (débris, satellite), nous prévoyons les positions des deux corps pendant une durée T en commençant à l'an 2020, année à laquelle les techniques de désorbitation des débris seraient implémentées. Cet instant est noté t_0 dans la suite. La durée T doit être assez longue pour couvrir toutes les positions relatives des deux corps. La prédiction des positions des objets spatiaux en fonction du temps est effectuée grâce à l'algorithme SGP4 (Simplified General Perturbations) [53, 18] développé également par le NORAD. Ce code est déterministe et la position d'un objet à un instant donné ne dépend donc que de l'incertitude sur la position initiale. Le code SGP4 peut être considéré comme une fonction boîte noire. Celle-ci prend en entrées la position et la vitesse d'un objet à l'instant t_0 et renvoie en sortie, la prédiction de sa position et de sa vitesse à l'instant $t_0 + t$, où t est choisi par l'utilisateur.

Si à un instant donné $t \in [t_0, t_0 + T]$ la distance entre les positions des deux corps passe en deçà d'un seuil critique $d'_{lim} > d_{lim}$, nous considérons que les chances qu'une collision se produise ne sont pas négligeables. Cette situation se produit lorsque

$$\exists t \in [t_0, t_0 + T] \ d(p(s,t), p(d,t)) \leq d'_{lim}$$

où $p(x,t)$ définit la position d'un objet x à un instant t et $d(p(s,t), p(d,t))$ dénote la distance entre deux positions $p(s,t)$ et $p(d,t)$. Si, pour un couple (débris, satellite), la distance entre les positions prévues est à tout moment supérieure à d'_{lim} , nous considérons que la probabilité de collision entre ces deux corps est nulle. Ce calcul

permet de limiter le nombre de couples pour lesquels l'estimation de la probabilité va être effectuée.

Affinage du calcul pour les couples ayant des probabilités de collision non négligeables

En fait, la position des objets à l'instant t_0 est aléatoire. Celle-ci dépend de la précision des mesures du senseur mis en oeuvre pour détecter l'objet. Nous considérons que la position initiale d'un débris $p(d, t_0)$ ou d'un satellite $p(s, t_0)$ suit une loi normale multivariée N dont la moyenne est la position de l'objet donnée par les TLE et dont la matrice de covariance est définie par l'erreur commise par le senseur.

Cette erreur sur la position initiale des objets est propagée par le code SGP4. Par conséquent, les positions prédites $p(s, t)$ et $p(d, t)$ pour $t \in [t_0, t_0 + T]$ sont aussi des variables aléatoires. La probabilité à estimer sur ces variables aléatoires est définie par $P_{col}(s, d) = P(\exists t \in [t_0, t_0 + T], d(p(s, t), p(d, t)) \leq d_{lim})$.

Pour estimer la valeur de $P_{col}(s, d)$, des techniques sophistiquées telles que les méthodes de splitting décrites dans l'article de Morio et al. [55] ont été mises en place. Ces méthodes sont plus efficaces en terme de temps de calcul que les méthodes de type Monte Carlo rappelées dans le livre de Sobol [71], qui sont certes plus simples à mettre en place mais moins précises pour l'estimation d'un événement rare, telle que peut l'être la collision de deux objets spatiaux. Le principe de la technique de splitting est d'exprimer la probabilité de collision $P_{col}(s, d)$ sous forme de probabilités conditionnelles, plus simples à estimer que $P_{col}(s, d)$ par simulation car moins rares. Soit $S_1 > S_2, \dots, > S_n = d_{lim}$ une suite décroissante de seuils sur la distance entre les objets. Définissons l'événement \mathbf{A}_k : " $\exists t \in [t_0, t_0 + T], d(p(s, t), p(d, t)) \leq S_k$ ". La probabilité $P_{col}(s, d)$ peut se réécrire de la manière suivante :

$$P_{col}(s, d) = P(\mathbf{A}_n) = \prod_{k=1}^{n-1} P(\mathbf{A}_{k+1} | \mathbf{A}_k) \times P(\mathbf{A}_1)$$

où $P(\mathbf{A}_{k+1} | \mathbf{A}_k)$ est la probabilité de \mathbf{A}_{k+1} sachant \mathbf{A}_k . Chaque probabilité conditionnelle est ensuite estimée de proche en proche à l'aide d'un algorithme particulière, comportant une étape de sélection et une étape de mutation des particules. Le comportement de cet algorithme particulière est en fait très proche d'un algorithme génétique. Pour plus de détails sur l'algorithme d'estimation de probabilité utilisé, le lecteur peut se référer à Liang et al. [39]. Pour chaque estimation de probabilité $P_{col}(s, d)$, un budget d'environ 20000 échantillons a été utilisé au lieu des 10^6 à 10^7 échantillons nécessaires aux techniques Monte Carlo. Cela est suffisant pour estimer des probabilités de l'ordre de 10^{-6} . Le calcul nous permet d'évaluer les performances du débris sur le premier critère. $g_1(d)$ est la probabilité que le débris entre en collision avec au moins un satellite de la constellation, il a pour expression :

$$g_1(d) = 1 - \prod_{s \in S} (1 - P_{col}(s, d))$$

En pratique, la durée T doit être un majorant d'une période commune au satellite et aux débris. Les périodes étant exprimées en nombres entiers de minutes, il suffit de prendre $T = \max_{i, j \in D \cup S} (P_i \times P_j)$ où P_i dénote la période de l'orbite d'un objet i . Les périodes des objets LEO étant de l'ordre de quelques heures, la durée T est discrétisée par un pas de temps de 5 minutes. La distance d_{lim} doit être largement supérieure à la somme des rayons des deux objets et prendre en considération les erreurs commises lors de la prédiction des positions des objets. Nous avons choisi $d'_{lim} = 10km$, $d_{lim} = 10m$ et $T = 100h$.

2.3.3 Evaluation des positions des débris par rapport aux zones encombrées

Les trois zones encombrées identifiées dans la littérature sont notées z_1, z_2 et z_3 . Pour $j \in \{1, 2, 3\}$ la zone z_j est décrite par une altitude $a(z_j)$ et une inclinaison $i(z_j)$. On considère qu'un débris appartient à une zone si la différence entre son altitude et celle de la zone est inférieure à un seuil k_a et la différence entre son inclinaison et l'inclinaison de la zone est inférieure à un seuil k_i . Le débris sera considéré très éloigné d'une zone z_j si la différence entre son altitude et celle de la zone $a(z_j)$ est supérieure à un seuil l_a et si la différence entre son inclinaison et l'inclinaison de la zone $i(z_j)$ dépasse un seuil l_i .

On note pour $j \in \{1, 2, 3\}$ $A_j = |a(d) - a(z_j)|$ et $B_j = |i(d) - i(z_j)|$. Nous établissons deux mesures pour chaque débris et chaque zone :

$$\alpha_j = \frac{l_a - \max\{\min\{A_j, l_a\}, k_a\}}{l_a - k_a}$$

$$\beta_j = \frac{l_i - \max\{\min\{B_j, l_i\}, k_i\}}{l_i - k_i}$$

Nous souhaitons agréger ces deux mesures afin d'évaluer la performance des débris sur le deuxième critère. L'agrégation doit se faire de telle sorte que le résultat de l'agrégation est d'autant plus élevé que les deux mesures sont élevées. Ceci peut être réalisé en choisissant un opérateur d'agrégation conjonctif. Notons T cet opérateur, une propriété naturelle est d'alors d'imposer que :

$$\forall a \in [0, 1] T(1, a) = a$$

Si on rajoute à cette condition des exigences telles que la non-décroissance qui signifie que pour une même valeur de l'une des deux mesures, la performance globale du débris le plus performant sur la deuxième mesure doit être plus grande, c'est-à-dire, pour x, x' et y dans l'intervalle $[0, 1]$:

$$x \leq x' \Rightarrow T(x, y) \leq T(x', y)$$

Si on rajoute de plus, les propriétés d'associativité et de commutativité, il advient que T est une norme triangulaire. Ces opérateurs ont été largement étudiés et leurs caractéristiques sont bien connues [23, 54, 19].

On peut citer, sans prétendre à l'exhaustivité, l'opérateur minimum, le produit algébrique, le produit de Hamacher défini par

$$\frac{xy}{x + y - xy}$$

ou encore le produit d'Einstein défini par

$$\frac{xy}{1 + (1 - x)(1 - y)}$$

Si l'on impose de plus que deux débris ayant des valeurs voisines sur les deux mesures aient des valeurs agrégées voisines, il advient que la norme triangulaire utilisée doit être continue, ce qui est vérifié par les trois produits. Pour des raisons de simplicité, nous avons choisi le produit algébrique pour agréger les deux mesures en un même critère. Les trois degrés d'appartenance aux trois zones encombrées doivent être ensuite agrégés. La fonction d'agrégation devrait permettre que 0 soit l'élément neutre ce qui veut dire que la performance globale d'un débris n'appartenant pas à une des trois zones doit être égale à l'agrégation des appartenances aux deux autres zones. De plus, 1 doit l'élément absorbant, c'est-à-dire que dès lors qu'un débris appartient très fortement à une des trois zones, sa performance sur le deuxième critère doit être égale à 1. Si l'on veut de plus que la fonction d'agrégation soit monotone, commutative et associative. Nous définissons les propriétés d'une T-conorme. Les T-conormes ont été également étudiées dans la littérature [54], on peut citer, à titre d'exemple, l'opérateur max, la somme d'Hamacher

$$\frac{x + y - 2xy}{1 - xy}$$

ou encore la somme d'Einstein définie par

$$\frac{x + y}{1 + xy}$$

A ce niveau d'agrégation, il suffit qu'un débris ait une forte appartenance à une zone encombrée pour que sa performance globale augmente, et ce quels que soient ses degrés d'appartenance aux deux autres zones. L'opérateur max, le plus simple, paraît ainsi le plus adéquat à la situation. Nous définissons donc la performance du débris sur le deuxième critère par :

$$g_2(d) = \max_{j \in \{1,2,3\}} (\alpha_j \times \beta_j)$$

On vérifie bien que, lorsqu'un débris appartient à une zone encombrée, alors il existe $j_0 \in \{1,2,3\}$ tel que

$$A_{j_0} \leq k_a \leq l_a$$

et

$$B_{j_0} \leq k_i \leq l_i$$

Ainsi :

$$\alpha_{j_0} = \frac{l_a - k_a}{l_a - k_a} = 1$$

et

$$\beta_{j_0} = \frac{l_i - k_i}{l_i - k_i} = 1$$

Par conséquent, on a $g_2(d) = 1$ dans ce cas. Lorsque le débris est très éloigné des zones encombrées cela signifie que :

$$\forall j \in \{1,2,3\} \quad A_j \geq l_a \quad \text{ou} \quad B_j \geq l_i$$

Ainsi

$$\forall j \in \{1,2,3\} \quad \alpha_j = 0 \quad \text{ou} \quad \beta_j = 0$$

Par conséquent, on a $g_2(d) = 0$ dans ce cas.

Dans les autres cas,

$$\exists j_0 \in \{1,2,3\} \quad A_{j_0} \leq l_a \quad \text{et} \quad B_{j_0} \leq l_i$$

et

$$\forall j \in \{1,2,3\} \quad A_j \geq k_a \quad \text{ou} \quad B_j \geq k_i$$

alors

$$\forall j \in \{1,2,3\} \quad 0 < \alpha_j \times \beta_j < 1$$

Ainsi, on a alors $0 < g_2(d) < 1$ dans les autres cas.

2.4 Procédure de catégorisation des débris

L'objectif de cette partie est concevoir une méthode d'affectation des débris aux différentes catégories. Nous définissons deux seuils sur le premier critère. Le premier seuil p_1 représente la valeur à partir de laquelle la performance du débris sur le premier critère est jugée suffisamment élevée pour qu'il menace directement la constellation. Le seuil $q_1 \leq p_1$ est le seuil en deçà duquel le débris ne présente pas directement un danger pour la constellation.

Un premier seuil p_2 est défini sur le deuxième critère. C'est le seuil à partir duquel l'appartenance des débris aux zones encombrées est considérée assez élevée pour dire qu'il présente un danger indirect pour la constellation. Tandis que $q_2 \leq p_2$ est le seuil en deçà duquel l'appartenance aux zones encombrées est tellement faible que le débris ne représente pas de danger indirect pour la constellation.

Ces quatre seuils permettent de partitionner l'ensemble des débris potentiels en neuf classes :

$$X = \{(i,j), i \in \llbracket 1,3 \rrbracket, j \in \llbracket 1,3 \rrbracket\}$$

tels que pour un débris d appartenant un sous-ensemble de débris (i,j)

$$\begin{cases} i = 1 & \text{si } g_1(d) \geq p_1 \\ i = 2 & \text{si } q_1 \leq g_1(d) < p_1 \\ i = 3 & \text{si } g_1(d) < q_1 \end{cases}$$

$$\begin{cases} j = 1 & \text{si } g_2(d) \geq p_2 \\ j = 2 & \text{si } q_2 \leq g_2(d) < p_2 \\ j = 3 & \text{si } g_2(d) < q_2 \end{cases}$$

Etant donnée la définition des catégories présentée en section 2.1, on peut répartir les classes comme suit. Un débris d sera affecté à la première catégorie C_1 s'il présente un danger direct pour la constellation ou s'il a une forte appartenance aux zones encombrées et présente un danger significatif pour la constellation, c'est-à-dire :

$$\begin{cases} g_1(d) \geq p_1 \text{ ou} \\ g_2(d) \geq p_2 \text{ et } g_1(d) \geq q_1, \end{cases}$$

Ainsi

$$C_1 = \left(\bigcup_{j=1}^3 (1,j) \right) \cup (2,1)$$

Le débris sera affecté à C_3 s'il ne présente pas de danger direct ou indirect à la constellation, soit

$$\begin{cases} g_1(d) < q_1 \text{ et} \\ g_2(d) < q_2 \end{cases}$$

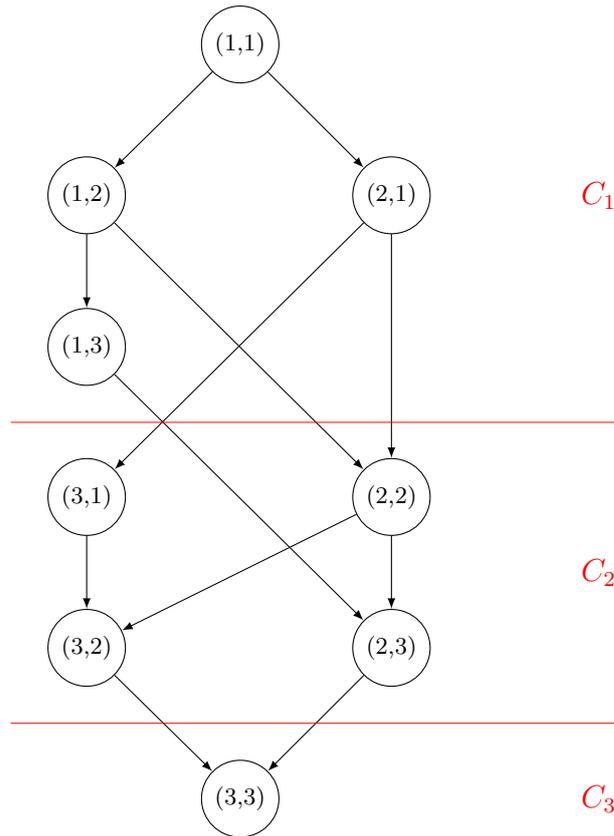
Ainsi

$$C_3 = (3,3)$$

La catégorie C_2 contient tous les autres débris.

Pour affiner davantage le classement au sein des catégories, nous définissons sur l'ensemble X , la relation R permettant de traduire des règles de priorité. Ainsi, pour des classes x et y de X , $x R y$ si et seulement si les débris qui sont dans x sont au moins aussi prioritaires à être enlevés que ceux qui sont dans y .

Ainsi, pour deux éléments $x = (x_1, x_2)$ et $y = (y_1, y_2)$ Il paraît naturel de dire que $x R y$ si $x_1 \leq y_1$ et $x_2 \leq y_2$ ce qui signifie qu'un débris au moins aussi dangereux sur les deux critères est au moins aussi prioritaire à être enlevé. Notons R_1 ce premier cas où la relation R s'applique. Nous avons $R_1 \subset R$, la relation R_1 induit un ordre


 FIGURE 2.1 – Graphe correspondant à l'ordre induit par la relation R_1

partiel sur les classes des débris, cet ordre est présenté dans le graphe de la figure 2.1.

Afin d'être cohérent avec la catégorisation en trois classes C_1 , C_2 et C_3 , il paraît naturel de chercher à enrichir R_1 . Ainsi entre C_1 et C_2 , il conviendrait d'ajouter les arcs $((1,2); (3,1))$ et $((1,3); (3,1))$. Ces arcs traduisent l'idée que les débris présentant un danger direct très fort sont plus prioritaires que les autres ce qui peut être fait en définissant la relation $R_2 \subset R$ telle que :

$x R_2 y$ si et seulement si $x_1 = 1$ et $y_1 \geq 2$.

Proposition 1. La relation $R = \bigcup_{i=1}^2 R_i$ définit un ordre sur l'ensemble X .

Preuve 1. R est réflexive via R_1 . Montrons que R est antisymétrique et transitive : soit $x = (x_1, x_2)$, $y = (y_1, y_2)$, deux éléments de X tels que, $x R y$ et $y R x$ montrons que $x = y$. Le seul cas où on peut avoir $x R y$ et $y R x$ est $x R_1 y$ et $y R_1 x$ ce qui implique que $x_1 \leq y_1$, $x_2 \leq y_2$, $y_1 \leq x_1$ et $y_2 \leq x_2$. Il en découle que $x = y$. Montrons que R est transitive :

soit $x = (x_1, x_2)$, $y = (y_1, y_2)$, et $z = (z_1, z_2)$ trois éléments de X tels que, $x R y$ et $y R z$ montrons que $x R z$.

Si $x R_1 y$ et $y R_1 z$, il advient que $x R_1 z$.

Dans le cas où $x R_1 y$ et $y R_2 z$ nous avons $x_1 \leq y_1$, $x_2 \leq y_2$ et $y_1 = 1$, $z_1 \geq 2$, ce qui permet de dire $x_1 = 1$, $z_1 \geq 2$, alors $x R_2 z$.

Dans le cas où $x R_2 y$ et $y R_1 z$, nous avons $x_1 = 1$ et $y_1 \geq 2$ et $y_1 \leq z_1$ et $y_2 \leq z_2$ alors

$x_1 = 1$ et $z_1 \geq 2$, ce qui permet de conclure que xR_2z .

Le cas où xR_2y et yR_2z est impossible car il implique que $y_1 \geq 2$ et $y_1 = 1$ ce qui est impossible.

Ce qui achève la démonstration.

□

R définit un ordre sur l'ensemble X . Nous représentons les deux relations $\{R_i, i \in \llbracket 1,2 \rrbracket\}$ par les graphes de la figure 2.2.

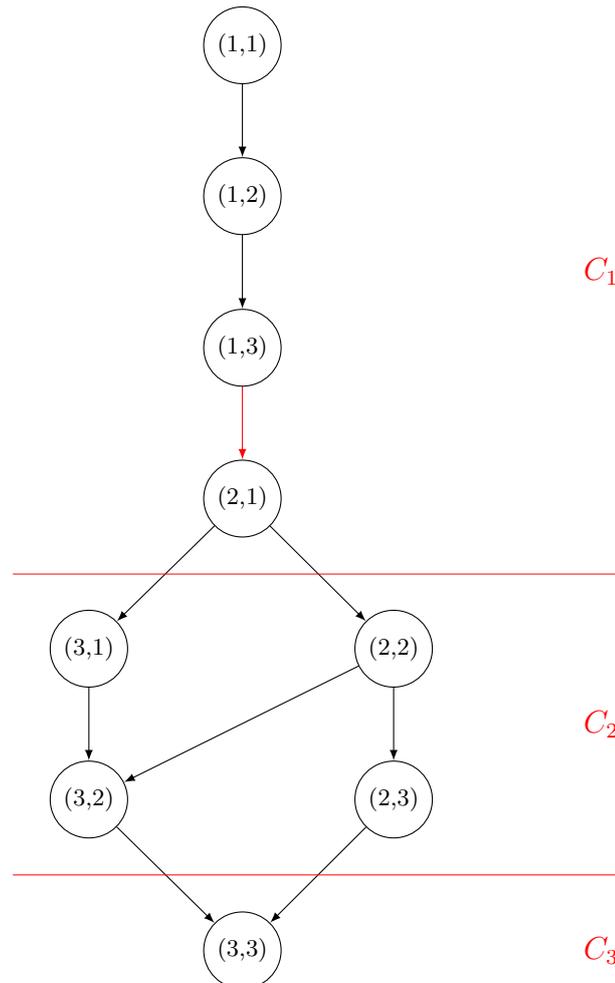


FIGURE 2.2 – Graphe correspondant à l'ordre induit par la relation $R_1 \cup R_2$

Il est intéressant de remarquer que la relation $R_1 \cup R_2$ permet d'établir un ordre total sur les éléments de la première catégorie. Elle autorise toutefois trois incomparabilités entre des éléments de la deuxième catégories. Ainsi la classe (2,3) est incomparable avec les classes (3,2) et (3,1) et les classes (3,1) et (2,2) sont incomparables aussi. Si on souhaite affiner davantage l'ordre, on pourrait considérer que les débris de la classe (2,3) sont au moins aussi dangereux que ceux de la classe (3,2), cela revient à considérer qu'une différence de classement sur le deuxième critère peut être compensée par une différence au moins aussi importante sur le premier. Cette règle peut être formalisée de la façon suivante :

pour $x = (x_1, x_2)$ et $y = (y_1, y_2)$, $x R_3 y$ si et seulement si $x_1 + x_2 \leq y_1 + y_2$ et $x_1 < y_1$. Par le biais de cette règle, les débris de la classe (2,3) sont au moins aussi prioritaires à être enlevés que ceux de la classe (3,2). Ce qui réduit le nombre d'incomparabilités entre les neuf classes à un.

Proposition 2. *La relation $R = \bigcup_{i=1}^3 R_i$ définit un ordre sur l'ensemble X .*

Preuve 2. *Il suffit de montrer que R_3 combinée avec R_1 et R_2 préserve l'antisymétrie et la transitivité. En effet, pour tout $x = (x_1, x_2)$ et $y = (y_1, y_2)$ de X , si $x R_1 y$ alors $x_1 \leq y_1$ et par conséquent, on ne peut avoir $y R_3 x$.*

Inversement si $x R_3 y$ alors $x_1 < y_1$ et par conséquent, on ne peut avoir $y R_1 x$.

De même si $x R_2 y$ alors $x_1 = 1$ et par conséquent, on ne peut avoir $y_1 < x_1$ alors on ne peut avoir $y R_3 x$.

Si $x R_3 y$ alors $x_1 < y_1$ et par conséquent, on ne peut avoir $y_1 = 1$ alors on ne peut avoir $y R_2 x$.

Le cas où $x R_3 y$ et $y R_3 x$ ne peut pas non plus se présenter car cela implique que $x_1 < y_1$ et $y_1 < x_1$.

Ainsi l'antisymétrie est préservée. La transitivité est également préservée, en effet, Soit $x = (x_1, x_2)$, $y = (y_1, y_2)$ et $z = (z_1, z_2)$ des éléments de X :

Dans le cas où $x R_1 y$ et $y R_3 z$ nous avons $x_1 \leq y_1$, $x_2 \leq y_2$ et $y_1 + y_2 \leq z_1 + z_2$ et $y_1 < z_1$, ce qui permet de dire $x_1 + x_2 \leq z_1 + z_2$ et $x_1 < z_1$, alors $x R_3 z$.

Dans le cas où $x R_2 y$ et $y R_3 z$, il advient que $x_1 = 1$, $y_1 \geq 2$, $y_1 + y_2 \leq z_1 + z_2$ et $y_1 < z_1$, ce qui permet de dire que $x_1 = 1$ et $z_1 \geq 2$ alors $x R_2 z$.

Dans le cas où $x R_3 y$ et $y R_1 z$, il advient que $x_1 + x_2 \leq y_1 + y_2$, $x_1 < y_1$, $y_1 \leq z_1$ et $y_2 \leq z_2$, ce qui permet de dire que $x_1 + x_2 \leq z_1 + z_2$ et $x_1 < z_1$ alors $x R_3 z$.

Dans le cas où $x R_3 y$ et $y R_2 z$, il advient que $x_1 + x_2 \leq y_1 + y_2$, $x_1 < y_1$, $y_1 = 1$ et $z_1 \geq 2$, ce qui permet de dire que $x_1 = 1$ et $z_1 \geq 2$ alors $x R_2 z$.

Dans le cas où $x R_3 y$ et $y R_3 z$, il advient que $x_1 + x_2 \leq y_1 + y_2$, $x_1 < y_1$, $y_1 + y_2 \leq z_1 + z_2$ et $y_1 < z_1$, ce qui permet de dire que $x_1 + x_2 \leq z_1 + z_2$ et $x_1 < z_1$, alors $x R_3 z$.

Ainsi $R = \bigcup_{i=1}^3 R_i$ définit bien un ordre sur l'ensemble X

□

L'ordre induit par R sur X réduit permet de comparer la quasi-totalité des classes de X comme montré dans la figure 2.3.

Nous appliquons la procédure de catégorisation sur l'ensemble des débris LEO, nous calculons la cardinalité de chaque sommet du graphe de la figure 2.3. Les seuils choisis sont $p_1 = 10^{-2}$ et $q_1 = 10^{-4}$ pour le premier critère et $p_2 = 0.9$ et $q_2 = 0.3$ pour le deuxième.

les résultats sont présentés dans la Table 2.1.

D'après le tableau, la première catégorie comporte 4209 débris, dont 123 plus urgents à éliminer que le reste des débris de la catégorie. Parmi tous les éléments de la première catégorie, seulement, 20 débris peuvent être traités dans l'immédiat. Les autres débris nécessitent des études plus approfondies afin développer des

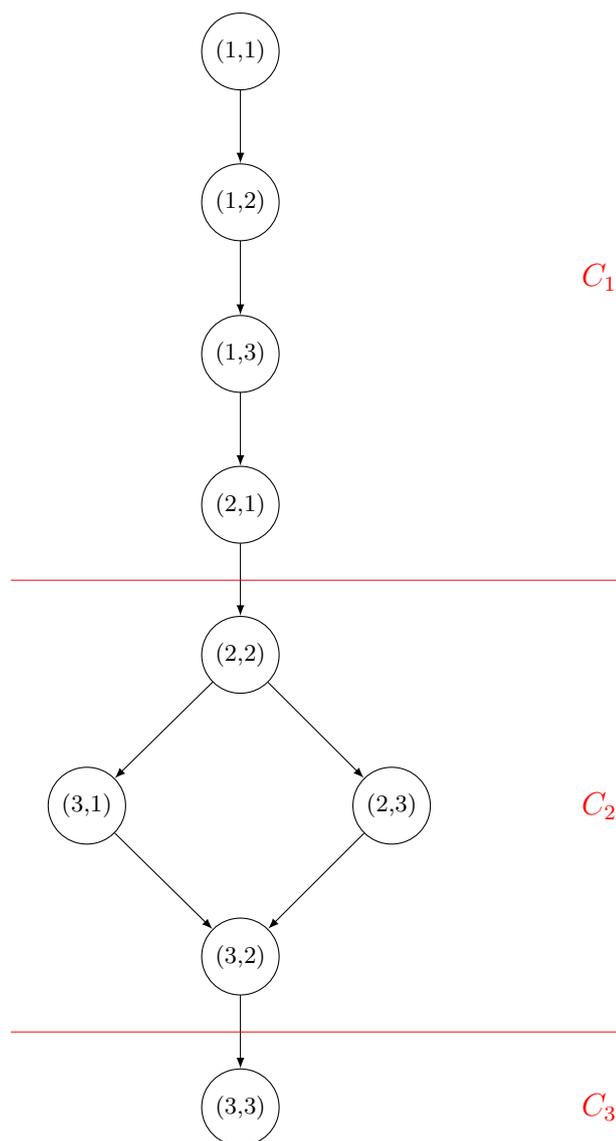


FIGURE 2.3 – Graphe correspondant à l'ordre induit par la relation $R = \bigcup_{i=1}^3 R_i$

Catégorie	Ordre	Classes	Nombre débris	Européens et facilement manipulables	Européens et non facilement manipulables	Non européens et facilement manipulables	Non européens et non facilement manipulables
C_1	1	(1,1)	123	0	0	19	104
	2	(1,2)	1118	0	0	103	1015
	3	(1,3)	2850	17	53	393	2387
	4	(2,1)	118	3	0	37	78
C_2	5	(2,2)	899	0	0	103	796
	6	(2,3) et (3,1)	3005	16	51	569	2369
	7	(3,2)	551	0	1	102	448
C_3	8	(3,3)	6345	180	259	2539	3367

TABLE 2.1 – Nombre de débris dans chaque classe

méthodes de désorientation des débris dont la forme est inconnue ou nécessitent de plus un accord avec les pays qui sont à l'origine des débris. Il est remarquable que les débris les plus dangereux de la catégorie C_1 (débris appartenant à (1,1) et (1,2)) ne sont pas européens et ne peuvent être traités dans l'immédiat.

La deuxième catégorie comporte 4555 débris, seuls 16 parmi ces débris peuvent être traités immédiatement. La deuxième catégorie contient deux sous-catégories incomparables à savoir les ensembles de débris (2,3) et (3,1).

La dernière catégorie contient 6345 débris dont 180 sont éliminables dans l'immédiat.

Dans la section suivante, nous présentons une méthode pour effectuer un choix des débris à éliminer, nous y distinguons deux cas, le cas où une seule navette est utilisée et le cas où plusieurs navettes sont disponibles.

2.5 Sélection des débris selon la capacité de traitement

Les débris appartenant à la première catégorie sont les plus urgents à éliminer. Leur nombre peut cependant dépasser la capacité de traitement d'une navette spatiale. Plusieurs navettes peuvent être utilisées pour traiter l'ensemble des débris. Nous distinguons deux cas, le cas où une seule navette est utilisée et le cas où l'on dispose de plusieurs navettes.

Dans le cas où l'on dispose d'une seule navette, il conviendrait de faire faire à cette même navette plusieurs missions afin de traiter tous les débris dont le retrait est urgent. L'idée serait de traiter les débris dans l'ordre de l'urgence.

Si le nombre des navettes est suffisant pour traiter les débris urgent simultanément, la question serait comment répartir les débris entre les navettes.

Notons que le cas intermédiaire où l'on dispose de plusieurs navettes mais qui ne peuvent désorbiter tous les débris dangereux est une combinaison de ces deux cas. En effet, tout se passe comme si l'on disposait d'une seule navette dont la capacité est la somme des capacités des navettes disponibles dans un premier temps. Il convient ensuite de répartir les débris de cette première mission fictive entre les différentes navettes.

Nous proposons dans ce paragraphe des modélisations possibles du problème

2.5.1 Utilisation d'une seule navette de désorbitation

Si l'on dispose d'une seule navette, on traiterait les débris dans l'ordre de priorité. On pourrait ainsi traiter la classe (1,1) puis (1,2) et ainsi de suite jusqu'à ce qu'on arrive à la première classe, ou famille de classes incomparables qu'on ne peut traiter en entier. Il convient ainsi de choisir parmi les débris de cette ou ces classes ceux qui seront traités. Une idée consiste à définir une priorité en considérant d'autres aspects tels que la proximité.

Sachant que l'on dispose déjà d'une liste de débris à traiter, il paraît raisonnable de la compléter avec les débris les plus proches dont la désorbitation coûterait le moins possible. L'étude réalisée dans la deuxième partie de cette thèse montre que le changement d'inclinaison est le facteur qui contribue le plus au coût d'un

transfert interorbital. Il serait ainsi souhaitable d'éliminer d'abord les débris dont le traitement n'engendre pas de changements d'inclinaison importants.

L'inclinaison est l'angle que fait le plan orbital avec un plan de référence. Ce dernier étant en général le plan de l'écliptique dans le cas d'orbites planétaires (plan contenant la trajectoire de la Terre , voir Figure 2.4).

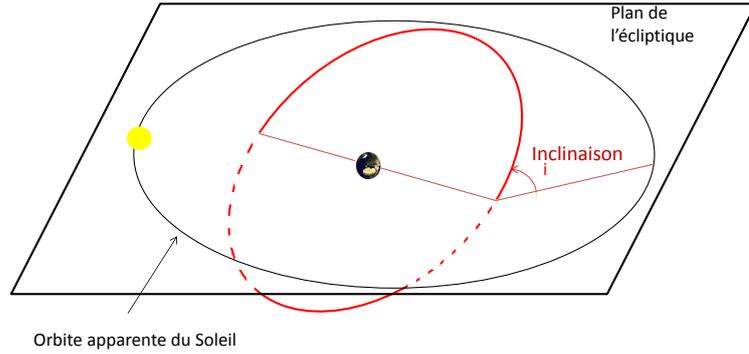


FIGURE 2.4 – Inclinaison d'une orbite par rapport au plan de l'écliptique

Ainsi, soit C la capacité de traitement de la navette, \bar{X} l'ensemble de débris sélectionné et \tilde{X} la première classe ou union de classes que la navette ne peut traiter en entier. L'idée est de choisir $C - |\bar{X}|$ débris parmi les débris de \tilde{X} . Si l'on note par I_j l'inclinaison d'une orbite j , le changement maximal qu'une navette sera amenée à effectuer pour traiter les débris de \bar{X} est :

$$\max_{i,j \in \tilde{X}} (I_j - I_i) = \max_{j \in \tilde{X}} I_j - \min_{j \in \tilde{X}} I_j$$

Un débris $i \in \tilde{X}$, tel que $\min_{j \in \tilde{X}} I_j \leq I_i \leq \max_{j \in \tilde{X}} I_j$ n'augmente pas l'étendue de l'inclinaison et devrait être plus prioritaire qu'un autre tel que $I_i < \min_{j \in \tilde{X}} I_j$ ou $I_i > \max_{j \in \tilde{X}} I_j$. Si le nombre de débris qui n'augmentent pas l'étendue de l'inclinaison est exactement $C - |\bar{X}|$, il convient de les traiter dans leur intégralité. Si ce nombre est différent de $C - |\bar{X}|$, il faudra considérer, en plus de l'étendue de l'inclinaison, d'autres aspects pour compléter le choix des débris.

Si ce nombre est inférieur, il conviendrait de compléter la liste par les débris les plus proches de ceux sélectionnés, la proximité d'un débris peut être définie par :

$$\forall i \in \tilde{X} \text{ } prox(i) = \min_{j \in \tilde{X}} |I_i - I_j| = \min(|I_i - \max_{j \in \tilde{X}} I_j|, |I_i - \min_{j \in \tilde{X}} I_j|)$$

Cela revient à compléter la liste par les débris qui augmentent le moins possible l'étendue de l'inclinaison. Pour ce faire, nous définissons des variables binaires x_i telles que :

$$x_i = \begin{cases} 1 & \text{si le débris } i \text{ est sélectionné,} \\ 0 & \text{sinon} \end{cases} \quad i \in \bar{X} \cup \tilde{X}$$

Ce problème peut être modélisé comme suit :

$$\begin{cases} \min \max_{i,j \in \bar{X} \cup \tilde{X}} |I_i - I_j| x_i x_j \\ \sum_{i=1}^{|\bar{X}|+|\tilde{X}|} x_i = C \\ \forall i \in \bar{X} \quad x_i = 1 \\ \forall i \in \bar{X} \cup \tilde{X} \quad x_i \in \{0,1\} \end{cases}$$

Afin de linéariser le produit apparaissant dans la fonction objectif, il convient de définir des variables binaires y_{ij} représentant le produit $x_i x_j$. La fonction objectif devient $\min \max_{i,j \in \bar{X} \cup \tilde{X}} |I_i - I_j| y_{ij}$. Comme la fonction objectif est à minimiser, les variables y_{ij} prennent naturellement la valeur 0. Afin de les forcer à 1 lorsque x_i et x_j sont égaux à 1, il convient de rajouter les deux contraintes suivantes :

$$\begin{cases} \forall i,j \in \tilde{X} \cup \bar{X} \quad y_{ij} \geq x_i + x_j - 1 \\ \forall i,j \in \tilde{X} \cup \bar{X} \quad y_{ij} \in \{0,1\} \end{cases}$$

La fonction objectif peut être linéarisée comme suit :

$$\begin{cases} \min \mu \\ \forall i,j \in \tilde{X} \cup \bar{X} \quad \mu \geq |I_i - I_j| y_{ij} \end{cases}$$

Si, par contre, le nombre de débris qui n'augmentent pas l'étendue des inclinaisons est supérieur à $C - |\bar{X}|$, il conviendrait de choisir les débris de telle sorte que les changements d'inclinaison effectués par la navette soient le plus homogène possible. Il convient ainsi de minimiser le plus grand écart d'inclinaison entre deux orbites successives choisies. Numérotons les débris appartenant dans l'ordre croissant des inclinaisons de leurs orbites et notons m le nombre de débris qui appartiennent à \bar{X} ou qui n'augmentent pas l'étendue d'inclinaison de \bar{X} . L'écart maximal à minimiser est ainsi :

$$\begin{cases} \min \max_{i < j} \left((I_j - I_i) x_i x_j \prod_{k=i+1}^{j-1} (1 - x_k) \right) \\ \sum_{i=1}^m x_i = C \\ \forall i \in \bar{X} \quad x_i = 1 \\ \forall i \in \llbracket 1, m \rrbracket \quad x_i \in \{0,1\} \end{cases}$$

Le produit apparaissant dans la fonction objectif peut être linéarisé en introduisant des variables binaires y_{ij} . La fonction objectif devient :

$$\min \max_{i < j} (I_j - I_i) y_{ij}$$

Ces variables prennent naturellement la valeur 0 comme la fonction objectif est à minimiser, il convient de les forcer à 1 lorsque le produit est non nul. Cela peut être effectué en introduisant deux contraintes :

$$\begin{cases} y_{ij} \geq x_i + x_j - \sum_{k=i+1}^{j-1} x_k - 1 \\ \forall i, j \in \llbracket 1, m \rrbracket \quad y_{i,j} \in \{0,1\} \end{cases}$$

Enfin la fonction objectif peut être linéarisée comme suit :

$$\begin{cases} \min \mu \\ \forall i < j \quad \mu \geq (I_j - I_i)y_{ij} \end{cases}$$

2.5.2 Utilisation de plusieurs navettes de désorbitation

Supposons que plusieurs navettes sont disponibles afin de désorbiter l'ensemble des débris. Notons par n le nombre des débris à désorbiter, nous supposons que chaque navette i a une capacité de traitement de C_i débris.

Soit m le nombre de ces navettes, le nombre total de débris qui peut être traité est $\sum_{i=1}^m C_i$.

Le problème à résoudre serait de savoir quels débris seront traités par quelle navette et dans quel ordre, ce qui peut être modélisé comme un problème de tournées de véhicules (VRP) largement étudié dans la littérature [17, 36, 37].

Le problème étant NP-complet [60] sa résolution s'avère extrêmement coûteuse. Une simplification possible consiste à déterminer, a priori, pour chaque navette l'ensemble des débris qu'elle aura à traiter et de planifier une mission de retrait par navette.

Naturellement, les meilleurs partitionnements sont ceux qui minimisent le coût global des missions et qui, de plus, affectent aux mêmes navettes des débris nécessitant un même traitement.

Nous nous intéressons dans cette section à la modélisation du problème de partitionnement de n débris entre m navettes, avec $n \leq \sum_{i=1}^m C_i$. La modélisation doit prendre en compte les deux contraintes suivantes :

- un débris doit être affecté à une seule navette.
- chaque navette j doit traiter au plus C_j débris.

Nous définissons des variables binaires x_{ij} telles que :

$$x_{ij} = \begin{cases} 1 & \text{si le débris } i \text{ est affecté à la navette } j, i = 1, \dots, n, j = 1, \dots, m \\ 0 & \text{sinon} \end{cases}$$

La première contrainte peut être respectée en imposant que :

$$\forall i \in \llbracket 1, n \rrbracket \sum_{j=1}^m x_{ij} = 1$$

La deuxième revient à imposer :

$$\forall j \in \llbracket 1, m \rrbracket \sum_{i=1}^n x_{ij} \leq C_j$$

L'objectif du partitionnement peut être de minimiser le coût global du traitement de l'ensemble des débris. En considérant que les transferts qui coûtent le plus sont ceux qui nécessitent d'effectuer de grands changements d'inclinaison d'orbite, il convient de minimiser les différences d'inclinaison entre les orbites des débris appartenant à la même classe. Notons par I_i l'inclinaison d'une orbite i . L'objectif serait donc de minimiser le maximum des différences d'inclinaison entre les débris traités par chaque navette. Pour une navette k parmi les m navettes, le maximum des changements d'inclinaisons qui pourrait être effectué est :

$$\max_{i,j} |I_i - I_j| x_{ik} x_{jk}$$

où I_j dénote l'inclinaison de l'orbite j .

Le problème peut être modélisé comme suit :

$$\begin{cases} \min \max_{i,j,k} |I_i - I_j| x_{ik} x_{jk} \\ \forall i \in \llbracket 1, n \rrbracket \quad \sum_{j=1}^m x_{ij} = 1 \\ \forall j \in \llbracket 1, m \rrbracket \quad \sum_{i=1}^n x_{ij} \leq C_j \\ \forall i \in \llbracket 1, n \rrbracket \quad \forall j \in \llbracket 1, m \rrbracket \quad x_{ij} \in \{0,1\} \end{cases}$$

Pour linéariser la fonction objectif, on peut introduire des variables binaires z_{ijk} représentant les produits $x_{ik}x_{jk}$. Comme la fonction objectif est à minimiser, les variables z_{ijk} prennent naturellement la valeur 0. Il convient donc seulement de forcer les variables z_{ijk} à 1 lorsque $x_{ik} = 1$ et $x_{jk} = 1$. Pour ce faire il suffit d'introduire les contraintes suivantes :

$$\begin{cases} \forall i, j \in \llbracket 1, n \rrbracket \quad \forall k \in \llbracket 1, m \rrbracket \quad z_{ijk} \geq x_{ik} + x_{jk} - 1 \\ \forall i, j \in \llbracket 1, n \rrbracket \quad \forall k \in \llbracket 1, m \rrbracket \quad z_{ijk} \in \{0,1\} \end{cases}$$

Le programme linéaire devient :

$$\left\{ \begin{array}{l} \min \max_{i,j,k} |I_i - I_j| z_{ijk} \\ \forall i \in \llbracket 1, n \rrbracket \quad \sum_{j=1}^m x_{ij} = 1 \\ \forall j \in \llbracket 1, m \rrbracket \quad \sum_{i=1}^n x_{ij} \leq C_j \\ \forall i, j \in \llbracket 1, n \rrbracket \quad \forall k \in \llbracket 1, m \rrbracket \quad z_{ijk} \geq x_{ik} + x_{jk} - 1 \\ \forall i, j \in \llbracket 1, n \rrbracket \quad \forall k \in \llbracket 1, m \rrbracket \quad z_{ijk} \in \{0, 1\} \\ \forall i \in \llbracket 1, n \rrbracket \quad \forall j \in \llbracket 1, m \rrbracket \quad x_{ij} \in \{0, 1\} \end{array} \right.$$

La fonction objectif peut être linéarisée de la façon suivante :

$$\left\{ \begin{array}{l} \min \mu \\ \forall i, j \in \llbracket 1, n \rrbracket \quad \forall k \in \llbracket 1, m \rrbracket \quad \mu \geq |I_i - I_j| z_{ijk} \end{array} \right.$$

Alternativement, on pourrait souhaiter minimiser, au sein de chaque navette, le nombre de débris nécessitant un traitement différent, c'est-à-dire, le nombre de débris de types différents. On note par s le nombre de types de débris. Pour ce faire, il convient de définir les paramètres τ_{ik} tels que :

$$\tau_{ik} = \begin{cases} 1 & \text{si le débris } i \text{ est de type } k, i = 1, \dots, n, k = 1, \dots, s \\ 0 & \text{sinon} \end{cases}$$

Ainsi le nombre n_{kj} d'objets de type k affectés à la navette j est :

$$n_{kj} = \sum_{i=1}^n \tau_{ik} \times x_{ij}$$

Définissons les variables y_{kj} binaires indiquant si la navette j traite un débris de type k ou non.

$$y_{kj} = \begin{cases} 1 & \text{si la navette } j \text{ traite au moins un débris de type } k \\ \text{c'est-à-dire si } n_{kj} \geq 1, k = 1, \dots, s, j = 1, \dots, m \\ 0 & \text{sinon} \end{cases}$$

Le nombre de types de débris traités par la navette j est $\sum_{k=1}^s y_{kj}$. L'objectif serait ainsi de minimiser la diversité au sein des débris traités par une même navette.

$$\min \max_{j \in \llbracket 1, m \rrbracket} \left\{ \sum_{k=1}^s y_{kj} \right\}$$

Comme la fonction objectif est à minimiser, les variables y_{kj} prennent naturellement la valeur 0, afin de les forcer à 1 lorsque $n_{kj} \geq 1$, on impose la contrainte

$$\forall k \in \llbracket 1, s \rrbracket \quad \forall j \in \llbracket 1, m \rrbracket \quad n_{kj} \leq y_{kj} \times C_j$$

Finalement la fonction objectif est linéarisée en introduisant une variable μ telle que :

$$\begin{cases} \min \mu \\ \forall j \in \llbracket 1, m \rrbracket \quad \mu \geq \sum_{k=1}^s y_{kj} \end{cases}$$

Conclusion du chapitre

Nous avons proposé une nouvelle approche pour déterminer les débris qui semblent prioritaires à enlever dans le cadre d'une prochaine mission de retrait. La méthode proposée ne fournit pas un ordre total sur l'ensemble des débris selon leur priorité mais permet de catégoriser ces débris en se basant sur plusieurs aspects.

Les débris sont ainsi ordonnés en plusieurs classes, ce qui nous permet de choisir ceux qui feront l'objet d'une première mission. Après avoir sélectionné les débris dont le retrait serait pertinent, il convient de planifier une mission spatiale pour le retrait de ces débris.

Selon le nombre de navettes disponibles pour effectuer le traitement, nous avons proposé des modélisations possibles du problème de traitement de débris par des navettes spatiales. En choisissant de partitionner les débris de telle sorte que chaque ensemble de débris soit traité par une navette, le problème qui se pose est de planifier chaque mission.

La planification d'une mission spatiale doit être réalisée de telle sorte que le coût total de la mission soit minimal. Le temps est également un facteur à prendre en considération lors de la planification. En effet, plus le temps s'écoule plus de nouveaux débris seront générés. On se propose dans la seconde partie de ce travail d'étudier le problème de planification d'une mission de retrait d'un ensemble de débris par une navette.

Chapitre 3

Présentation du problème de retrait des débris spatiaux

Sommaire

3.1	Planification d'une mission de retrait de débris spatiaux	50
3.1.1	Réalisation d'un rendez-vous spatial entre la navette et un débris	51
3.1.2	Calcul du coût d'un transfert de Lambert	51
3.2	Notations	52
3.3	Formulation du problème	53
3.4	État de l'art sur le problème de voyageur de commerce biobjectif et dépendant du temps	55
	Conclusion	59

La population des objets spatiaux étudiés est constituée de l'ensemble des débris à éliminer et de la navette spatiale. Sous l'effet de l'attraction terrestre, chaque objet de la population est en révolution autour de la Terre occupant ainsi une orbite que l'on suppose elliptique. La navette doit effectuer un tour afin de visiter les orbites l'une après l'autre, traiter un à un les débris puis retourner à l'orbite initiale. Plusieurs décisions sont à prendre afin de réaliser le tour complet. Ainsi, la navette doit choisir la prochaine orbite à visiter à chaque fois. Dès lors que ce dernier choix est fait, il convient de choisir l'instant de départ de l'orbite courante et la durée du transfert de telle sorte qu'un rendez-vous soit réalisé avec le prochain débris à traiter. Ce chapitre décrit comment la planification d'une mission de retrait des débris spatiaux est effectuée et présente une modélisation du problème associé.

3.1 Planification d'une mission de retrait de débris spatiaux

Très peu de travaux traitent la planification de missions visant le retrait des débris spatiaux. Cerf [16] a abordé le problème de stabilisation de la taille de la population des objets spatiaux en éliminant 5 débris annuellement comme le suggéraient Liou et Johnson [43]. Il propose donc une approche pour sélectionner et éliminer exactement 5 débris parmi n'importe quel nombre de débris. Dans son approche, le temps est considéré comme une contrainte et le coût de la mission est le seul critère à minimiser. Les débris sélectionnés sont les moins coûteux à traiter et le problème est modélisé comme la recherche des k plus courts chemins de longueur 5 dans un graphe complet. Un algorithme de Branch and Bound a été utilisé dans cette approche.

Des missions analogues ont été abordées. Ainsi Wadsley et al. [81] ont proposé une approche permettant de planifier une mission spatiale pour la maintenance des satellites. Ils ont étudié la version mono-objectif du problème visant la minimisation du coût de la tournée sans se soucier de sa durée. Le problème est ainsi formulé comme un problème de voyageur de commerce. La méthode de résolution proposée repose sur plusieurs hypothèses. Ainsi, on se donne une durée fixe pour effectuer n'importe quel transfert. On suppose de plus que le temps de service est le même quel que soit le satellite servi. On suppose en outre que les satellites sont en mouvement circulaire autour de la Terre. Cette dernière hypothèse est légitimée par les faibles excentricités des orbites des satellites considérés. En faisant cette hypothèse, il n'est plus besoin de déterminer la valeur optimale de la vitesse de la navette au moment où la manœuvre est effectuée. La dernière hypothèse consiste à supposer que les demi-grands axes des orbites sont tous égaux. Pour appréhender le problème, une approche exacte basée sur la programmation dynamique a été utilisée. Des instances allant jusqu'à 20 satellites sont alors résolues. Stodgell [73] abordent le problème de visite de satellites en minimisant le coût et la durée de la mission. Le problème est donc formulé comme un problème de voyageur de commerce biobjectif. Dans leur étude, les orbites sont supposées elliptiques. Entre chaque paire de débris, les coûts correspondants à différentes possibilités de transferts sont calculés et seul le transfert minimisant le coût est retenu ce qui constitue une restriction importante, d'autres transferts de coûts plus élevés, mais plus rapides pouvant être pertinents. Une instance comportant 6 satellites a été résolue en utilisant un algorithme génétique multiobjectif.

Dans le cadre de notre étude, on souhaite proposer une méthode exacte pour trouver l'ensemble des tours efficaces qui permettront d'optimiser les deux critères coût et durée de la mission. Pour ce faire, il convient de décider de la manière de réaliser un rendez-vous spatial avec les débris et du traitement à effectuer une fois le rendez-vous réalisé.

3.1.1 Réalisation d'un rendez-vous spatial entre la navette et un débris

L'arrivée de la navette sur une orbite doit s'effectuer de manière à réaliser un rendez-vous avec le débris occupant cette orbite. Si la navette arrive sur un point de l'orbite autre que celui qu'occupe le débris, le rendez-vous spatial ne pourra s'effectuer sans que la navette change d'altitude. La navette et le débris se suivront sans jamais se rencontrer. Différentes techniques de transfert sont possibles afin de permettre à la navette de se déplacer d'une orbite à une autre. Les manœuvres orbitales sont caractérisées par un changement de la vitesse orbitale ΔV , c'est une grandeur vectorielle qui vient s'ajouter au vecteur vitesse initiale de l'objet V_1 résultant en un nouveau vecteur vitesse V_2 . Quand le changement de vitesse est fait instantanément la manœuvre est dite impulsive.

Les manières d'effectuer le transfert diffèrent selon le nombre d'impulsions requises, leurs intensités et les points où elles sont appliquées. Considérons par exemple le transfert de Hohmann. Celui-ci s'effectue entre deux orbites circulaires, coplanaires et décrites dans le même sens, c'est le transfert bi-impulsionnel qui nécessite le moins de carburant [65]. Pour les transferts entre deux orbites pas forcément décrites dans le même sens, le transfert peut s'effectuer suivant la méthode bi-elliptique. Le transfert bi-elliptique est effectué en trois impulsions entre deux orbites coplanaires et circulaires [30]. Ces deux premières techniques ne seront pas utilisées dans notre cas étant donné qu'elles supposent que les orbites des objets sont circulaires. Nous opterons pour une méthode qui permet de prendre en considération le caractère elliptique des orbites des débris. Nous nous baserons pour ce faire, sur le théorème de Lambert. Lambert a démontré qu'étant donné un point de départ sur une première orbite et un point d'arrivée sur une deuxième orbite, il existe un seul arc de conique permettant de joindre les deux points dans une durée de temps fixée [32]. Des algorithmes permettant de calculer le coût de ce transfert existent dans la littérature [6]. Ces algorithmes s'avèrent néanmoins très gourmands en terme de temps de calcul.

3.1.2 Calcul du coût d'un transfert de Lambert

Afin de calculer le coût d'un transfert elliptique effectué selon la méthode de Lambert, l'algorithme commence par calculer les paramètres orbitaux de l'ellipse de transfert étant donné le point de départ et d'arrivée et la durée de transfert. La première impulsion sert à mettre l'objet sur l'orbite de transfert dont les paramètres ont été calculés à l'étape précédente. Connaissant l'ellipse de transfert, l'algorithme calcule le vecteur vitesse tangent à l'ellipse de départ et à celle du transfert ce qui permet de calculer le coût de la première impulsion (voir la figure 3.1). La deuxième impulsion sert à freiner la navette afin qu'elle passe à l'orbite du débris à désorbiter. De même, l'algorithme calcule le coût de la deuxième impulsion et retourne la somme des coûts. Le calcul s'avère coûteux en terme de temps. Dans notre cas, le calcul du coût d'un transfert ne se fait qu'après que l'instant de départ

et la durée de transfert sont fixés. En effet, dès lors que ces deux paramètres sont fixés, la position d'arrivée est connue puisque c'est la position qui permet de réaliser un rendez-vous avec le débris sur l'orbite de destination. Notre algorithme principal fait donc appel à l'algorithme de Lambert à chaque fois qu'il est besoin de déterminer le coût d'un transfert pour une durée choisie en entrée.

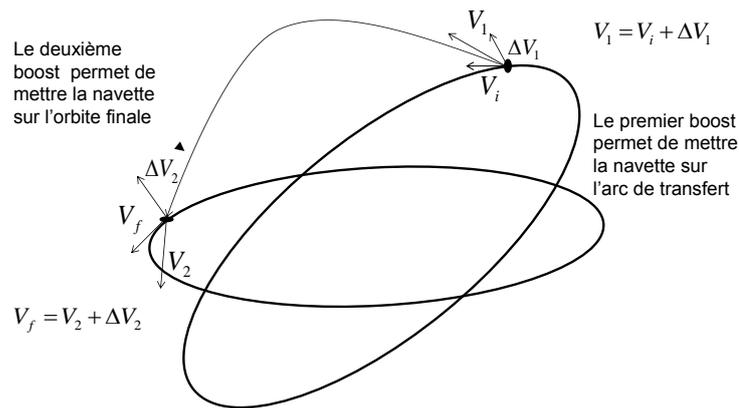


FIGURE 3.1 – Transfert entre orbite 1 et orbite 2 effectué selon la méthode de Lambert.

3.2 Notations

Les objets spatiaux étudiés sont numérotés de 0 à n . L'objet 0 correspond à l'orbite initiale de la navette tandis que les objets de 1 à n font référence aux débris à éliminer. La quantité de carburant brûlée afin d'achever le transfert constitue le coût du transfert. Le coût d'un transfert entre deux débris i et j est noté c_{ij} . La durée de ce transfert est notée d_{ij} .

La quantité de carburant brûlée pendant la mission ne peut dépasser la capacité du réservoir de la navette. Cette capacité est notée c_{max} . De plus, une durée maximale de la mission est fixée et notée d_{max} . Etant donné que la navette réalise un rendez-vous avec chaque débris sur son orbite, on confond lors de cette étude chaque débris avec son orbite. Pour un débris i donné, t_i dénote l'instant auquel la navette atteint l'objet spatial. Lorsque la navette atteint l'orbite d'un débris elle procède à son traitement. Le temps de traitement du débris occupant une orbite i est noté δt_{serv_i} . Une fois le traitement effectué, la navette choisit un instant pour entamer son prochain voyage vers un débris qui n'est pas encore visité. Plusieurs instants de départ sont possibles, l'instant de départ au plus tôt d'une orbite i est noté $t_{i_f} = t_i + \delta t_{serv_i}$. L'instant de départ au plus tard de la même orbite t_{i_l} correspond au dernier instant permettant à la navette de terminer sa mission sans dépasser la durée maximale d_{max} . La modélisation adoptée doit prendre également

en considération plusieurs possibilités de transfert. Ainsi, la navette peut attendre une certaine durée avant d'entamer le transfert ou commencer tout de suite le voyage vers le prochain débris. La durée du transfert est la somme de la durée d'attente et la durée du voyage. La durée minimale d'un voyage interorbital est notée δt_{trans} et son coût est au moins égal à une quantité δc_{min} .

3.3 Formulation du problème

L'ensemble des objets spatiaux étudiés V et l'ensemble des transferts pouvant être effectués entre chaque paire d'orbites dans E sont modélisés par un graphe orienté $G = (V, E)$. Plusieurs arcs peuvent lier un couple de sommets, un arc correspondant à une durée d'attente et une durée de transfert données. L'ensemble des arcs liant un couple de sommets dépend de l'instant d'arrivée à l'orbite initiale. Ainsi à chaque triplet (i, j, t_i) correspond un ensemble d'arcs qu'on note $A_{ij}(t_i)$. Dans l'ensemble $A_{ij}(t_i)$, seuls les transferts efficaces sont retenus. C'est-à-dire que ne sont retenus que les transferts tels que toute amélioration sur l'aspect coût ou durée entraîne nécessairement une perte sur le deuxième aspect. Comme l'ensemble des arcs liant deux sommets dépend de l'instant d'arrivée au sommet de départ, le graphe G est dynamique.

La figure 3.2 montre comment le graphe se construit au fur et à mesure que la navette progresse dans son tour.

Le transfert entre un nœud i et un nœud non encore visité j après avoir visité un sous ensemble d'orbites $V' \subset V$ peut démarrer à tout instant $t_{i_f} \leq t \leq t_{i_i}$, la date t_{i_f} est le premier moment auquel la navette peut entamer son voyage, qui correspond à la date d'arrivée à l'orbite à laquelle est ajouté le temps de service du débris dans l'orbite de départ $t_{i_f} = t_i + \delta t_{serv_i}$. La date t_{i_i} correspond au dernier instant auquel la navette peut quitter l'orbite de départ i .

Cette date est liée à la durée maximale de mission ainsi qu'à la durée minimale de transfert vers les orbites non encore visités. La date de départ doit être choisie de telle sorte que le temps résiduel soit suffisant pour compléter la mission.

Ainsi, $t_{i_i} = t_0 + d_{max} - (n + 1 - |V'|) \times \delta t_{trans} - \sum_{i \in V \setminus V'} \delta t_{serv_i} - \delta t_{trans}$ où $t_0 + d_{max}$ est le dernier délai pour finir la mission, $(n + 1 - |V'|) \times \delta t_{trans}$ est le temps minimal pour se déplacer vers les débris qui ne sont pas encore visités, $\sum_{i \in V \setminus V'} \delta t_{serv_i}$ est le temps de les traiter et δt_{trans} est le temps minimal pour effectuer un retour vers l'orbite de départ de la navette. La durée d'attente est discrétisée en utilisant un pas de temps δt . Le transfert peut démarrer à n'importe quel instant dépassant t_{i_f} par un multiple de δt pourvu que cet instant reste antérieur à t_{i_i} . Soit : $t_{i_f} + w \times \delta t$ où $w \in I(t_i)$ et $I(t_i) = \{0, 1, \dots, \lfloor \frac{t_{i_i} - t_{i_f}}{\delta t} \rfloor\}$.

Chaque instant de départ possible peut donner lieu à plusieurs durées de transfert. L'ensemble des durées de transfert possibles est noté T . A un instant d'arrivée t_i , $w \in I(t_i)$ et une durée de transfert $p \in T$, correspond un arc permettant à la navette d'atteindre l'orbite d'arrivée après une durée de transfert $d_{ij}(t_i, w, p)$ ce

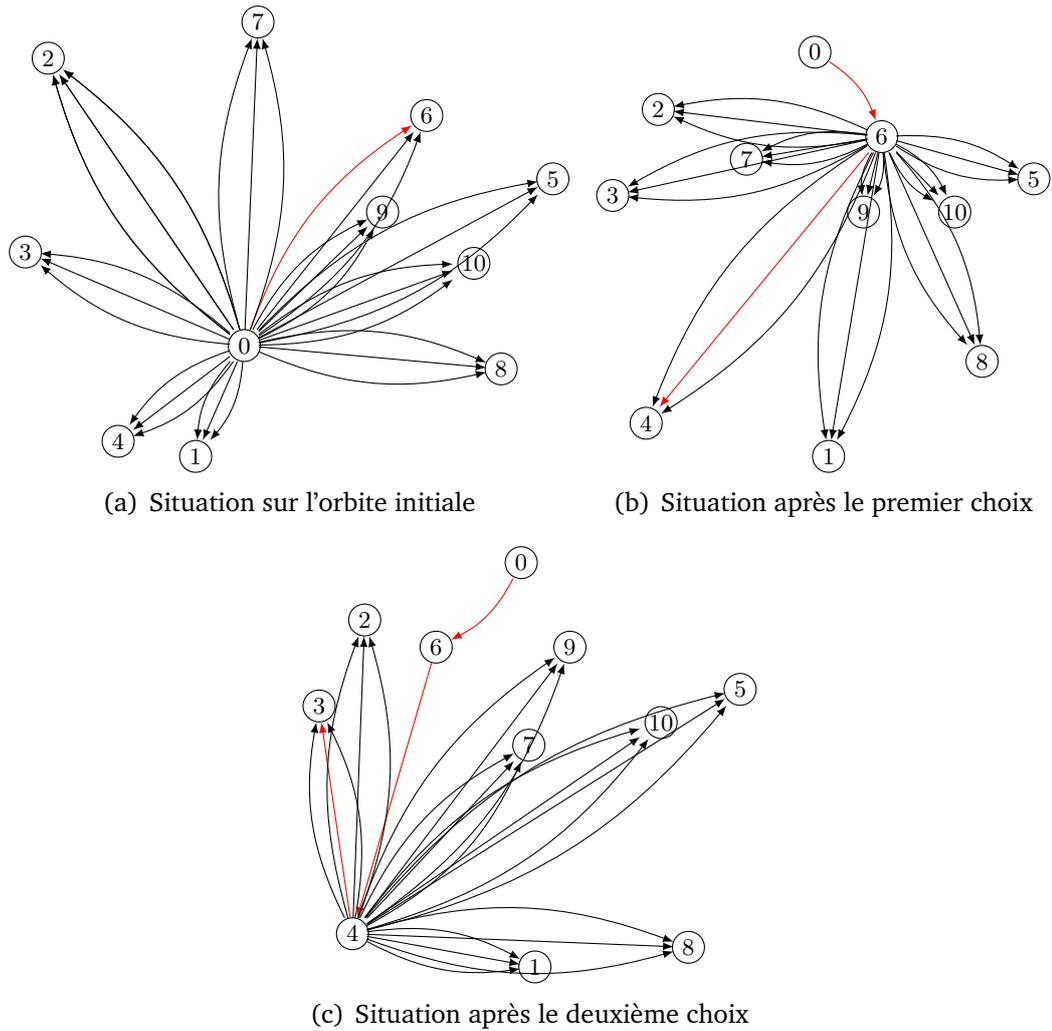


FIGURE 3.2 – Progression du graphe de la population des objets étudiés

qui permet d'atteindre l'orbite j à $t_j = t_{i_f} + d_{ij}(t_i, w, p)$. La valuation de cet arc est $(c_{ij}(t_i, w, p), d_{ij}(t_i, w, p))$, le coût $c_{ij}(t_i, w, p)$ étant calculé par l'algorithme de Lambert, les possibilités de transferts sont présentées sur la figure 3.3.

La mission que la navette doit effectuer correspond à un tour dans un multigraphe orienté et dynamique. L'objectif est d'effectuer ce tour en minimisant le coût total et la durée totale de la mission. Cela revient à résoudre un problème de voyageur de commerce biobjectif sur le graphe précédemment décrit. Le caractère dynamique du graphe rend le problème très dépendant du temps car la valuation d'un arc dépend de l'instant auquel l'extrémité initiale de l'arc a été atteinte et des durées d'attente et de voyage correspondant à cet arc. Le problème à résoudre est ainsi un problème de voyageur de commerce biobjectif et dépendant du temps.

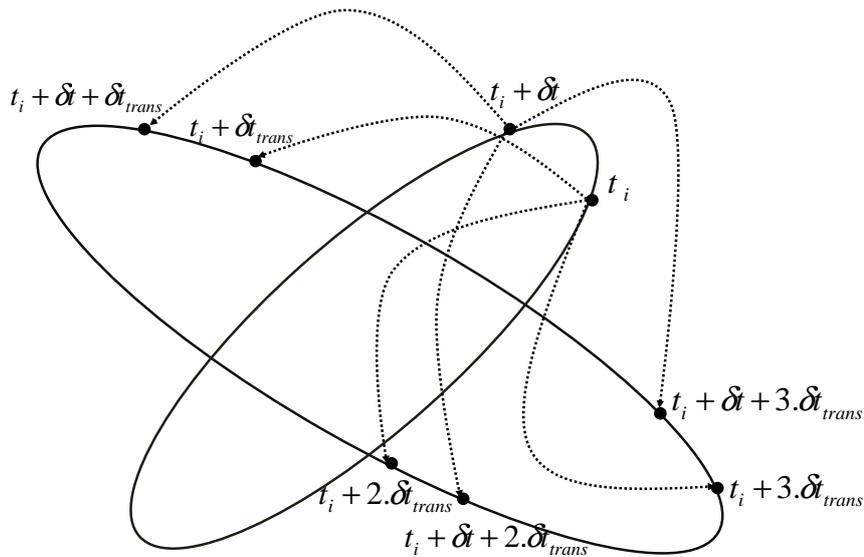


FIGURE 3.3 – Les possibilités de transfert si la navette atteint l'orbite i à t_i .

3.4 État de l'art sur le problème de voyageur de commerce biobjectif et dépendant du temps

Le problème du voyageur de commerce biobjectif et dépendant du temps n'a pas été traité dans la littérature à notre connaissance. Cependant, plusieurs articles traitent de la version monocritère du problème de voyageur de commerce dépendant du temps (TDTSP) et du problème du voyageur de commerce biobjectif (BiTSP).

TDTSP est une variante du problème de voyageur de commerce où la valuation sur chaque arc dépend de l'instant auquel le sommet de départ est atteint. Il peut avoir plusieurs application dans la vie réelle. Ceci peut se produire, par exemple, lorsque les durées de parcours dépendent de conditions extérieures (heures de pointe, heures creuses, . . .). Le problème de voyageur de commerce dépendant du temps permet de modéliser ce genre de situation en utilisant des fonctions de coût approprié à chaque arc.

Plusieurs formulations ont été fournies pour le TDTSP [27]. Dans [24] Fox propose une formulation en TDTSP pour un problème de gestion de production dans lequel la demande en personnel est dépendante au temps. Dans [62], Picard et Queyranne ont fourni trois formulations en programme en nombres entiers du TDTSP afin de modéliser le programme d'ordonnancement des tâches dont la durée d'exécution dépend de leurs positions dans la séquence. Ces formulations

sont utilisées afin de minimiser le coût du retard en utilisant un algorithme par séparation et évaluation. Des instances de 20 sommets ont été résolues de manière exacte.

Dans [25] Fox et al. ont présenté une nouvelle formulation en programmation linéaire du problème qui permet d'utiliser moins de contraintes mais plus de variables. Gouveia et Voss [28] ont présenté deux nouvelles formulations du TDTSP issues du problème d'affectation quadratique. Presque en même temps Vander Wiel et Sahinidis [75], ont proposé une modélisation semblable à celle vue dans [28] à la seule différence que dans [75], on démontre que la contrainte d'intégrité de certaines variables de décision peut être relâchée. Une heuristique de recherche locale qui étend l'algorithme de Lin-Kernighan a été développé pour le TDTSP, et les résultats expérimentaux présentés permettent de résoudre des instances de 100 sommets et assure un bon rapport à l'optimalité avec un meilleur coût de calcul.

Dans [76] ils ont développé un algorithme exact pour le TDTSP en utilisant la décomposition de Benders à une la formulation établie dans [75], ce qui leur a permis de résoudre des instances de 20 sommets de façon exacte.

Une approche intéressante est proposée par Bigras et al. dans [10]. Ils ont étudié la formulation proposée par Picard et Queyranne [62] et ont amélioré le modèle en rajoutant des coupes valides appliquées à la racine du problème. Ils ont développé un algorithme de Branch and Cut and Price qui permet de résoudre des instances allant jusqu'à 50 sommets du TDTSP. Dans deux papiers récents, Abeledo et al. [1] ont fourni de bons résultats expérimentaux en se basant sur les travaux de Bigras et al. [10] et dans [27], ils ont proposé une nouvelle formulation du TDTSP en se basant sur la formulation de fournie par Picard and Queyranne [62]. Leur principale contribution est d'avoir utilisé plus de variables et plus de contraintes pour définir les sous problèmes que dans le modèle d'origine. Les résultats numériques montrent que la solution des problèmes relâchés pour plusieurs instances testées est une solution pour le problème d'origine.

Dans [70], Shneider a adapté la métaheuristique de recuit simulé pour proposer une résolution efficace au TDTSP. Malandraki et al. [47] ont étendu les algorithmes de programmation dynamique proposé par Bellman [8] et par Karp et al. [29] pour prendre en compte la dépendance au temps. Ils ont ainsi proposé une heuristique basée sur la programmation dynamique pour le problème de voyageur de commerce incluant de nouvelles considérations telles que la dépendance au temps. L'algorithme se base sur l'hypothèse qu'il est impossible pour le voyageur de commerce de partir d'un sommet i à un instant $t_i^b > t_i^a$ et d'atteindre un autre sommet j à un instant plus tôt que s'il était parti à t_i^a . Cette hypothèse peut ne pas être vérifiée dans plusieurs cas de la vie réelle, citons par exemple le cas des vols entre deux villes, un avion partant plus tard pouvant arriver plus tôt qu'un autre qui fait une escale ou qui est moins rapide. L'optimalité n'est pas garantie du fait que seulement un certain nombre de tours partiels est retenu dans chaque

étape d'algorithme. Néanmoins, il est possible pour l'utilisateur de générer une solution correspondant à un compromis entre le temps de calcul et la qualité de la solution en choisissant le nombre de solutions partielles à retenir à chaque étape de l'algorithme. Dans un article apparu très récemment Miranda-Bront et al. [52] proposent une nouvelle formulation, en analysant les formulations proposées par Picard et al. [62] et Vander Wiel et Sahinidis [75]. Ils définissent plusieurs familles d'inégalités valides, qui leur ont permis d'améliorer les performances des algorithmes de Branch and Cut disponibles dans la littérature.

Plusieurs papiers traitent du problème du voyageur de commerce biobjectif. Dans [83] Yan et al. ont étudié l'étape de sélection dans les algorithmes évolutionnaires multiobjectifs, la sélection d'une famille de solutions qu'on va croiser est faite de façon à maximiser la probabilité de trouver une solution non dominée. Dans toutes simulations l'algorithme développé trouve des fronts de Pareto de meilleure qualité que ceux trouvés par l'algorithme de SPEA (Strength Pareto Evolutionary Algorithm).

Dans [2], Zhou et al. se sont intéressés à la question de génération d'instances pour le problème de MOTSP. Ils présentent ainsi une méthode pour fabriquer des instances du problème de voyageur de commerce multiobjectif (MOTSP) pour lesquelles le degré de conflit entre les objectifs est mesurable. De plus, ils ont proposé une approche dite d'estimation multiobjective des algorithmes de distribution basée sur la décomposition afin de résoudre le problème, cette méthode consiste à décomposer le problème en plusieurs sous problèmes mono-objectifs et de les résoudre de façon efficace en utilisant à la fois les informations de départ et les informations d'apprentissage. La résolution simultanée des sous-problèmes permet de fournir une approximation de la solution du problème d'origine.

Une comparaison entre l'algorithme génétique et MOEA/D est fournie dans [61]. Peng et al. ont montré lorsqu'on applique MOEA/D avec l'algorithme de recherche locale, les résultats trouvés sont généralement meilleurs qu'avec un algorithme génétique.

Dans [58], Paquete et al. propose un algorithme de recherche locale en deux phase pour approcher l'ensemble des solutions non dominées. La première étape de l'algorithme consiste à résoudre une version mono-objectif du problème puis démarrer de cette solution et chercher l'ensemble des solutions non dominées en résolvant une séquence de versions mono-objectifs du problème en utilisant à chaque fois un algorithme de recherche locale. Chaque fonction objectif des problème résolu est une agrégation des critères à optimiser.

Dans [57], Paquete et al. se sont intéressé au BiTSP, ils ont ainsi étendu les algorithmes de recherche locale qui existent pour le TSP au TSP multiobjectif. Une première façon d'étendre consiste à garder la même stratégie de recherche appliquée pour le TSP monocritère et la seule différence porte sur le critère d'acceptation d'une solution, une solution dans le voisinage est acceptée si elle n'est

dominée par aucune des solutions déjà choisies. Ainsi, partant d'une solution candidate, le voisinage de cette solution est défini en changeant 3 composantes de la solution et l'algorithme s'arrête lorsqu'on n'améliore pas la qualité de la frontière de Pareto obtenue au bout d'une certaine durée de recherche.

Une nouvelle procédure de recherche a été développée pour le MOTSP par Li dans [38], elle consiste à résoudre chaque version monocritère du problème par un algorithme de recherche locale et à repérer les arêtes les plus utilisées dans les solutions localement optimales. Ces arêtes sont dites attrayantes pour le critère optimisé. L'ensemble de ces arêtes pour tous les critères est ensuite utilisé pour construire un ensemble de solutions Pareto-optimales pour le problème de voyageur de commerce multiobjectif. L'approche de résolution a été appliquée sur une version bi-objectif du problème. La comparaison avec un algorithme de recherche locale appliquée à une version mono-objectif du problème agrégant les deux critères par une somme pondérée avec plusieurs jeux de poids montre que l'algorithme développé fournit une frontière de Pareto de très bonne qualité.

Gendreau et al. [9] ont notamment utilisé la méthode ϵ -contrainte afin de générer efficacement la frontière Pareto du problème de voyageur de commerce avec profit. Cette variante du problème de voyageur de commerce n'impose pas la visite de tous les sommets du graphe. Des profits sont associés aux sommets du graphe ce qui définit une version bi-objectif du problème de voyageur de commerce dans laquelle deux critères conflictuels sont à optimiser. En pratique, il s'agit de trouver un circuit hamiltonien défini sur un sous graphe tel que le coût de ce circuit soit minimal tandis que le profit est maximal. L'objectif de maximisation du profit fait qu'on cherche à maximiser le nombre de sommets retenus tandis que l'objectif de minimisation de coût fait que l'on cherche à visiter moins de sommets. L'approche de résolution consiste à considérer le premier critère (resp le deuxième) et d'imposer que la performance sur le deuxième ne dépasse pas un certain seuil ϵ_2 (resp ϵ_1). Les rôles des deux critères sont échangeables. Ainsi, il convient de résoudre des programmes linéaires mono-objectifs obtenus, la résolution est effectuée à l'aide d'une procédure de Branch and Cut. L'algorithme de résolution commence par calculer la performance des points idéal et nadir sur les deux critères. Partant de la performance du point Nadir sur le critère transformé en contrainte, la valeur de ϵ est diminuée, ce qui définit une suite de programmes linéaires à résoudre fournissant chacun au moins une solution efficace au problème.

Cette procédure étant susceptible de générer des points dominés, la frontière efficace est nettoyée à chaque fois qu'une nouvelle solution est calculée.

Schmitz et Niemann [69] se sont intéressés aux problèmes associés aux services de livraison pour lesquels les endroits à visiter n'ont pas tous la même priorité. Ainsi, à la distance séparant ces endroits, s'ajoute un deuxième critère qui traduit la priorité de visite. Après avoir précisé comment la priorité est déterminée ils ont présenté un algorithme de recherche locale afin d'approcher la frontière efficace.

Paquete et al. [59] ont développé un algorithme basé sur la recherche locale pour résoudre le problème de voyageur de commerce multiobjectif. L'algorithme

s'effectue en deux étapes. La première consiste à appliquer un algorithme de recherche locale à une solution réalisable générée aléatoirement afin de trouver une bonne solution du problème multicritère. La deuxième étape démarre de la solution trouvée à la fin de la première étape et résout une suite de programmes linéaires correspondant chacun à une scalarisation de l'ensemble des objectifs de départ. La résolution est effectuée à l'aide d'un algorithme de recherche locale. Pour chaque jeu de poids utilisé, l'algorithme de recherche locale démarre de la solution obtenue lors de l'étape précédente et l'ensemble des solutions non dominées obtenues est mis à jour à la fin de chaque algorithme.

Dans [45] Lust et Jaszkiwicz ont également proposé une heuristique de résolution basée sur la méthode de recherche locale à deux phases avec des techniques d'accélération pour le BiTSP.

La méthode de résolution s'effectue également en deux phases, la première étapes consiste à générer l'ensemble des solutions efficaces supportées en résolvant plusieurs versions monocritère du problème correspondant chacune à une agrégation des critères à optimiser.

La deuxième étape consiste à chercher l'ensemble des solutions efficaces non supportées situées entre deux solutions supportées, en utilisant une approche de recherche locale permettant d'explorer le voisinage des solutions supportées et d'inclure les solutions voisines, si celles-ci s'avèrent non faiblement dominées par celles déjà prises.

L'algorithme s'avérant coûteux en terme de temps de calcul, des techniques d'accélération doivent être utilisées afin de résoudre des instances de plus que 500 sommets. Le voisinage d'une solution étant défini par le changement de deux arêtes, une technique d'accélération consiste à vérifier qu'au moins une des arêtes entrantes n'est pas dominée par une arête sortante. Une deuxième technique consiste à considérer en priorité les sommets qui, dans les recherches précédentes, ont fait améliorer l'ensemble des solutions efficaces.

Conclusion du chapitre

Trouver une politique de visite optimale revient à résoudre un problème de voyageur de commerce biobjectif et dépendant du temps dans un graphe orienté dynamique. Peu de travaux ont été effectués pour résoudre ce problème. Dans notre étude, on se propose de résoudre des instances du problème en utilisant un algorithme basé sur une procédure par séparation et évaluation. Le degré de complexité du problème étant très élevé, il importe d'optimiser l'algorithme afin d'obtenir des solutions efficaces pour des instances de taille relativement grande.

Chapitre 4

Développement d'une procédure de résolution pour le problème BiTDTSP

Sommaire

4.1	Optimisation multiobjectifs	62
4.1.1	Introduction	62
4.1.2	Procédures par séparation et évaluation multiobjectifs . .	64
4.2	Description de la méthode	69
4.2.1	Arborescence de recherche	70
4.2.2	Adaptation de la méthode	70
	Conclusion	78

Nous proposons dans ce chapitre une approche basée sur une procédure par séparation et évaluation (Branch and Bound) afin de déterminer l'ensemble des tours non dominés dans le graphe défini dans la partie précédente. La première partie de ce chapitre est dédiée à un état de l'art des algorithmes de résolution des problèmes multiobjectifs se basant sur la technique de séparation et évaluation. Nous présentons ensuite l'arborescence de recherche de solutions dans le graphe précédemment défini. De par la complexité du problème, la taille de l'arborescence de recherche est assez grande. Il importe donc de réduire la taille de cette arborescence en coupant les branches qui ne mèneront pas à des solutions complètes efficaces. Nous présentons dans ce chapitre des techniques permettant de limiter la taille de l'arborescence de recherche. Les travaux de cette partie ont fait l'objet d'un premier article à Acta Astronautica [46] et de plusieurs conférences.

4.1 Optimisation multiobjectifs

4.1.1 Introduction

Les problèmes d'optimisation multiobjectif peuvent être formulés comme suit :

$$\min Cx : Ax \geq b, x \in \mathbb{R}^n$$

où C est une matrice $p \times n$ de coût, p étant le nombre d'objectifs et n la taille du problème, c^k est la $k^{\text{ième}}$ ligne dans C qui correspond au $k^{\text{ième}}$ objectif. Si m est le nombre de contraintes dans le problème, A est une matrice $m \times n$ de coefficients des contraintes et b un vecteur de \mathbb{R}^m . L'ensemble des solutions réalisables $X = \{x \in \mathbb{R}^n, Ax \geq b\}$, peut décrire des structures combinatoires, un arbre couvrant, une affectation, un tour dans un graphe... On supposera que X est fini, et on notera $Y = CX$ son image par C dans \mathbb{R}^p . X est défini dans \mathbb{R}^n dit *espace des décisions* tandis que Y est son image dans \mathbb{R}^p , *l'espace des objectifs*. Une solution réalisable \hat{x} est dite efficace ou Pareto optimale s'il n'existe pas d'autre solution $x \in X$ telle que $c^k x \leq c^k \hat{x}$ pour $k = 1, \dots, p$ avec au moins une inégalité stricte. Autrement dit, aucune autre solution ne fait au moins aussi bien que \hat{x} sur tous les critères tout en étant strictement meilleure sur un critère en particulier.

La notion d'efficacité est relative à l'espace des décisions. Dans l'espace des objectifs \mathbb{R}^p , l'image d'une solution efficace est appelée point *non dominé*. On notera X_E l'ensemble des solutions efficaces et Y_N l'ensemble des points non dominés. X_E peut être partitionné en deux sous-ensembles :

- Une solution efficace \hat{x} est dite *supportée* s'il existe un vecteur scalaire λ positif de \mathbb{R}^p tel que $\lambda^T C \hat{x} \leq \lambda^T C x$ pour tout $x \in X$. On notera X_{ES} l'ensemble des solutions efficaces supportées et par analogie, Y_{NS} l'ensemble des points non dominés supportés. Les points *non dominés supportés*, se trouvent sur la frontière non dominée de l'enveloppe convexe de Y ,
- Si un tel scalaire n'existe pas, la solution sera dite *non supportée*, et le point correspondant est dit point non dominé non supporté.

On peut encore distinguer deux types de solutions efficaces supportées :

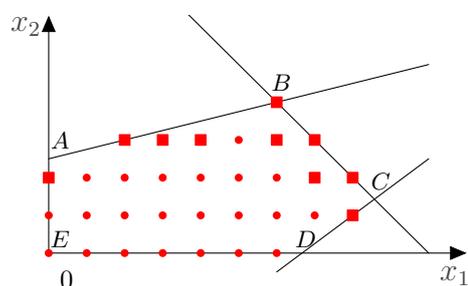
- Les solutions efficaces supportées *extrêmes* noté X_{ESE} dont l'image se situe sur un sommet de l'enveloppe convexe de Y . Leurs images dans l'espace des critères forment l'ensemble des points non dominés extrêmes Y_{NSE} .
- Les solutions efficaces supportées *non extrêmes* dont l'image n'est pas sur un sommet de l'enveloppe convexe de Y .

Afin d'illustrer ces différentes notions considérons l'exemple suivant.

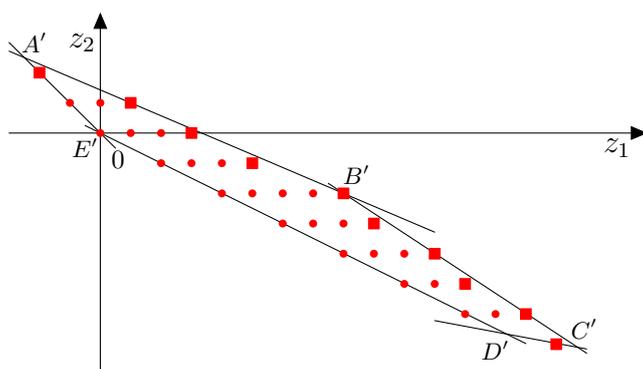
Exemple 1. Soit le programme linéaire biobjectif suivant :

$$\left\{ \begin{array}{l} \max z_1 = 2x_1 - x_2 \\ \max z_2 = -x_1 + x_2 \\ \text{s.c.} \\ \quad -x_1 + 4x_2 \leq 10 \\ \quad 3x_1 - 4x_2 \leq 20 \\ \quad x_1 + x_2 \leq 10 \\ \quad x_i \in \mathbb{N} \quad i = 1,2 \end{array} \right.$$

L'ensemble des solutions réalisables dans les espaces des décisions et l'espace des critères est présenté dans la figure 4.1



(a) Ensemble réalisable dans l'espace des décisions



(b) Son image dans l'espace des critères

FIGURE 4.1 – Ensemble des solutions réalisables dans l'espace des décisions et celui des critères

Dans cet exemple :

- les points en rouge correspondent aux solutions réalisables dans l'espace des décisions et à leurs performances dans l'espace des critères.
- $X_E = \{(0,2), (2,3), (3,3), (4,3), (6,4), (6,3), (7,3), (7,2), (8,2), (8,1)\}$.
- $Y_N = \{(-2,2), (1,1), (3,0), (5, - 1), (8, - 2), (9, - 3), (11, - 4), (12, - 5), (14, - 6), (15, - 7)\}$.
- $X_{ES} = \{(0,2), (2,3), (6,4), (7,3), (8,2), (8,1)\}$
- $Y_{NS} = \{(-2,2), (1,1), (8, - 2), (11, - 4), (14, - 6), (15, - 7)\}$
- parmi les solutions de X_{ES} , seule la solution $(7,3)$ n'est pas extrême, les images des autres solutions supportées font partie des sommets définissant le plus petit convexe contenant Y . Le vecteur correspondant $(11, - 4)$ est le seul vecteur non extrême.

Comme son nom l'indique, l'objet de l'optimisation multiobjectifs est de traiter des problèmes d'optimisation en présence de points de vue multiples. Deux types d'approches peuvent être envisagées : soit on cherche à générer l'ensemble des points non dominés (ainsi qu'une solution efficace associée), soit on cherche à déterminer une solution de compromis optimisant une fonction d'agrégation des critères. Chercher à calculer le front Pareto est plus naturel lorsqu'on ne connaît pas les préférences du décideur.

Le nombre de points non dominés peut toutefois augmenter exponentiellement en fonction de la taille de l'instance et du nombre d'objectifs. Cela justifie l'effort fourni en développement d'algorithmes d'optimisation multiobjectifs. La plupart des méthodes classiques exactes ou approchées pour les problèmes d'optimisation combinatoires ont été revisitées afin de proposer des versions multiobjectifs : programmation dynamique [35, 7, 22], les méthodes par séparation et évaluation [11, 34, 72]. Nous nous intéresserons à ce dernier type de méthodes en particulier.

4.1.2 Procédures par séparation et évaluation multiobjectifs

Un algorithme par séparation et évaluation, également appelé selon le terme anglophone *branch and bound*, est une méthode générique de résolution de problèmes d'optimisation, et plus particulièrement d'optimisation combinatoire ou discrète. C'est une méthode d'énumération implicite : toutes les solutions possibles du problème peuvent être énumérées au pire des cas mais, l'utilisation de bornes permet d'éviter l'énumération de sous-ensembles de mauvaises solutions. Dans les méthodes par séparation et évaluation, la séparation subdivise un ensemble de solutions en sous-ensembles tandis que l'évaluation sonde ces sous-ensembles afin d'éviter l'énumération exhaustive des solutions.

La phase de séparation consiste à diviser le problème en un certain nombre de sous-problèmes de telle sorte que les ensembles de solutions associées aux sous problèmes forment un recouvrement (idéalement une partition de l'ensemble de solutions associées au problème initial). Ainsi, en résolvant tous les sous-problèmes et en prenant la meilleure solution trouvée, on est assuré d'avoir résolu le problème initial. Ce principe de séparation peut être appliqué de manière récursive à chacun

des sous-ensembles de solutions obtenus, et ceci tant qu'il existe des ensembles contenant potentiellement des solutions intéressantes. Les ensembles de solutions (et leurs sous-problèmes associés) ainsi construits définissent une hiérarchie naturelle en arbre, dite arbre de recherche.

L'évaluation d'un nœud de l'arbre de recherche a pour but de déterminer une nouvelle solution intéressante ou, au contraire, de prouver que le sous-problème associé ne contient pas de solution intéressante pour la résolution du problème. Pour déterminer qu'un ensemble de solutions réalisables ne contient pas de solution optimale, la méthode la plus générale consiste à déterminer une borne inférieure pour le coût des solutions contenues dans l'ensemble, en cas de minimisation, et à éliminer cet ensemble dès que la borne est supérieure ou égale à la valeur de la meilleure solution trouvée.

Un algorithme par séparation et évaluation biobjectif diffère du cas mono-objectif lors de l'étape d'évaluation. Cette différence vient de la nature des bornes utilisées. La performance des algorithmes d'énumération implicite dépend de la qualité des bornes utilisées.

D'après [77], un ensemble BS bornant supérieurement l'ensemble des points non dominés Y_N est un ensemble tel que chaque élément de BS est non dominé ou est dominé par au moins un point non dominé du problème.

Afin de borner supérieurement l'ensemble des points non dominés, il suffit de considérer un ensemble de points réalisables. En effet, ces points réalisables permettent d'exclure de l'espace des critères les points qu'ils dominent, définissant ainsi une zone contenant les points non dominés (cf. zones hachurées des figures 4.2 et 4.3).

Plus ces points réalisables sont proches de la frontière Pareto, plus l'algorithme par séparation et évaluation sera performant. Une borne supérieure de bonne qualité peut être obtenue en résolvant de façon approchée le problème. Cette borne peut être améliorée au fur et à mesure que de nouvelles solutions sont contruites. Ainsi lorsqu'une solution réalisable est construite, il convient d'insérer son image dans l'ensemble bornant supérieurement à moins que le point correspondant ne soit dominé par un point déjà inséré et d'éliminer de cet ensemble les points dominés par ce nouveau point généré.

Construire une borne inférieure de l'ensemble image Y' d'un sous-problème X' est plus difficile. La première idée qui vient à l'esprit pour borner inférieurement un ensemble de points $Y' \subset Y$ (représentant l'image d'un sous ensemble de solutions $X' \subset X$ dans un branch and bound multiobjectif) est de définir un unique vecteur y^I tel que

$$y_k^I = \min_{x \in X'} c^k x \quad k = 1, \dots, p$$

Le point y^I est dit point *idéal*. Lorsque le point idéal correspondant à un sous-problème X' est dominé par un point bornant supérieurement l'ensemble des points non dominés du problème X , le sous-problème X' peut être écarté (voir premier cas dans la Figure 4.2).

Si par contre, ce point n'est pas dominé, on ne peut être affirmé que l'ensemble X' ne contient pas de points efficace sans l'explorer. Dans la figure 4.2, dans le premier sous-cas, le point idéal est dominé par le point A ce qui signifie que le sous-problème X' peut être écarté. Dans le deuxième sous-cas, le point idéal n'est pas dominé et l'exploration s'avère nécessaire.

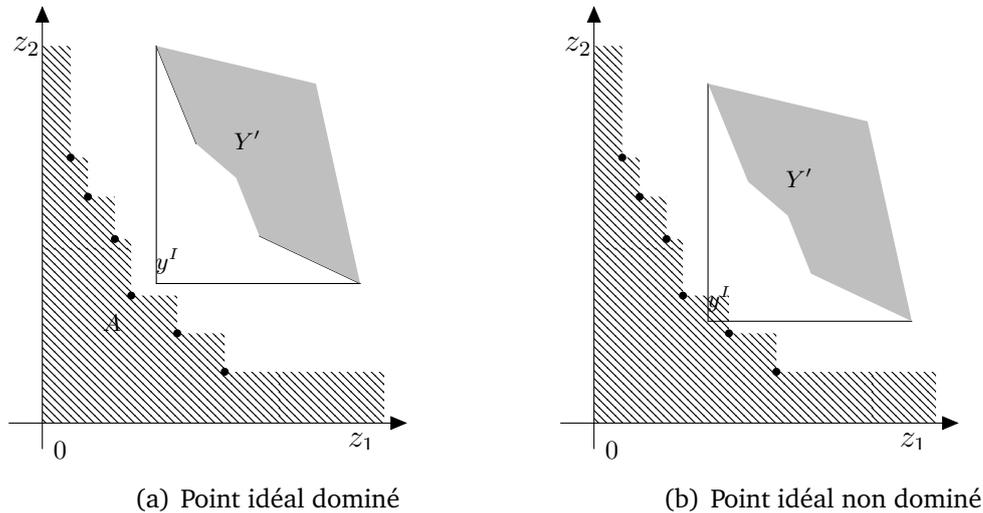


FIGURE 4.2 – Utilisation du point idéal

L'utilisation du point idéal pour borner inférieurement l'ensemble des vecteurs non dominés d'un sous-problème peut être difficile. C'est le cas lorsque la version mono-objectif du problème est NP-complet. Remarquons que tout point dominant le point idéal correspondant au sous-problème X' constitue également une borne inférieure pour l'ensemble Y' . Lorsque le problème à résoudre est complexe de telle sorte que le point idéal est difficile à obtenir, il convient d'utiliser une telle borne. Une telle borne présente l'avantage d'être plus facile à calculer mais elle coupe moins de sous-problèmes.

A l'inverse, et lorsque le problème mono-objectif est facile, on peut approcher plus finement l'image des solutions potentielles dans l'espace des objectifs en faisant appel à un ensemble de points plutôt qu'à un unique point idéal. La première définition d'une borne multiobjectif a été donnée par Villarreal et Karwan [77] pour les problèmes linéaires multiobjectifs à valeurs entières. Un ensemble BI de points constituera une borne inférieure de Y si :

- les points de BI sont non dominés ou dominent des points non dominés.
- chaque point non dominé est un élément de BI ou est dominé par un élément de BI .

Cette définition a été généralisée par Ehrgott et Gandibleux [20] en étendant ces notions aux sous ensembles $Y' \subset Y$ ce qui permet d'appliquer les bornes à un sous-problème du problème initial. La partie non dominée de l'enveloppe convexe de Y' notée $conv(Y')_N = conv(Y'_{NS})_N$ forme une borne inférieure de l'ensemble des vecteurs non dominés [21]. Le calcul de ces bornes nécessite la génération de l'ensemble des points extrêmes supportés, un algorithme permettant de calculer

ces points dans le cas biobjectif a été proposé par Ajena et Nair [3]. Cet algorithme nécessite d'effectuer $2 \times |Y'_{NSE}|$ optimisations d'une combinaison linéaire des deux objectifs, ce qui peut s'avérer très coûteux en terme de temps de calcul.

Une alternative consiste à minorer l'ensemble des vecteurs non dominés correspondant à un sous-problème X' par des droites correspondant à des combinaisons linéaires des deux objectifs ce qui revient à calculer quelques solutions supportées du sous-problèmes Y' . Cela permet de minorer plus finement l'ensemble Y' que lorsqu'on utilise le point idéal mais sans atteindre la qualité de la frontière obtenue avec l'enveloppe convexe des points supportés.

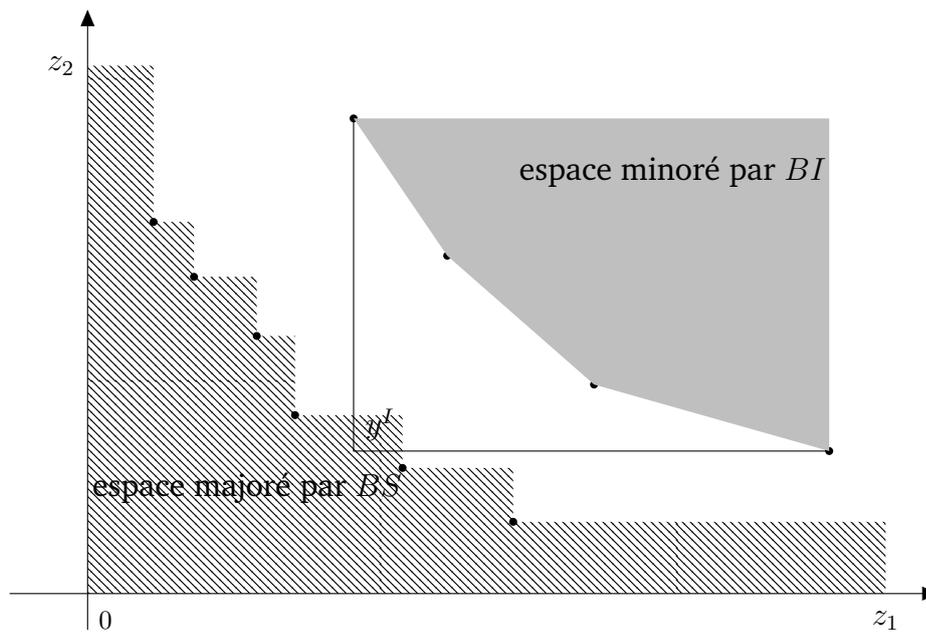


FIGURE 4.3 – Bornes multicritères.

Ces bornes permettent d'écarter des sous ensembles de l'espace réalisable en certifiant qu'il ne peut contenir des solutions efficaces. Ainsi, pour un sous-ensemble $X' \subset X$, supposons qu'on arrive à border inférieurement les éléments $Y' = CX'$ par un ensemble BI et à border supérieurement l'ensemble des solutions efficaces pour le problème principal par un ensemble BS comme montré dans la figure 4.3. Si l'ensemble majoré par BS (espace hachuré dans la figure) et l'ensemble minoré par BI (espace en gris dans la figure) sont disjoints, on peut écarter les solutions contenues dans X' . Si par contre l'intersection est non vide, on ne peut être sûr que le sous problème X' ne contient pas des solutions efficaces. Plus les bornes sont fines, plus nous avons des chances d'écarter des sous-ensembles. L'exemple de la figure 4.3 montre un cas où l'ensemble BI permet d'écarter le sous-problème X' tandis que le point idéal correspondant au même sous problème ne permet pas de conclure. Cependant la détermination de ces bornes peut s'avérer coûteuse en terme de temps de calcul. Il convient donc de réaliser un compromis entre le temps passé à déterminer les bornes et le temps économisé en évitant l'exploration de certaines branches stériles.

Une autre façon d'améliorer les performances consiste à comparer des sous-problèmes. Si X' et X'' sont deux sous-problèmes de X tels que tout élément de Y'' est dominé par au moins un élément de Y' , l'ensemble X'' peut être écarté comme dans la figure 4.4. Cela se produit en particulier lorsque $Y'' \subset Y'$.

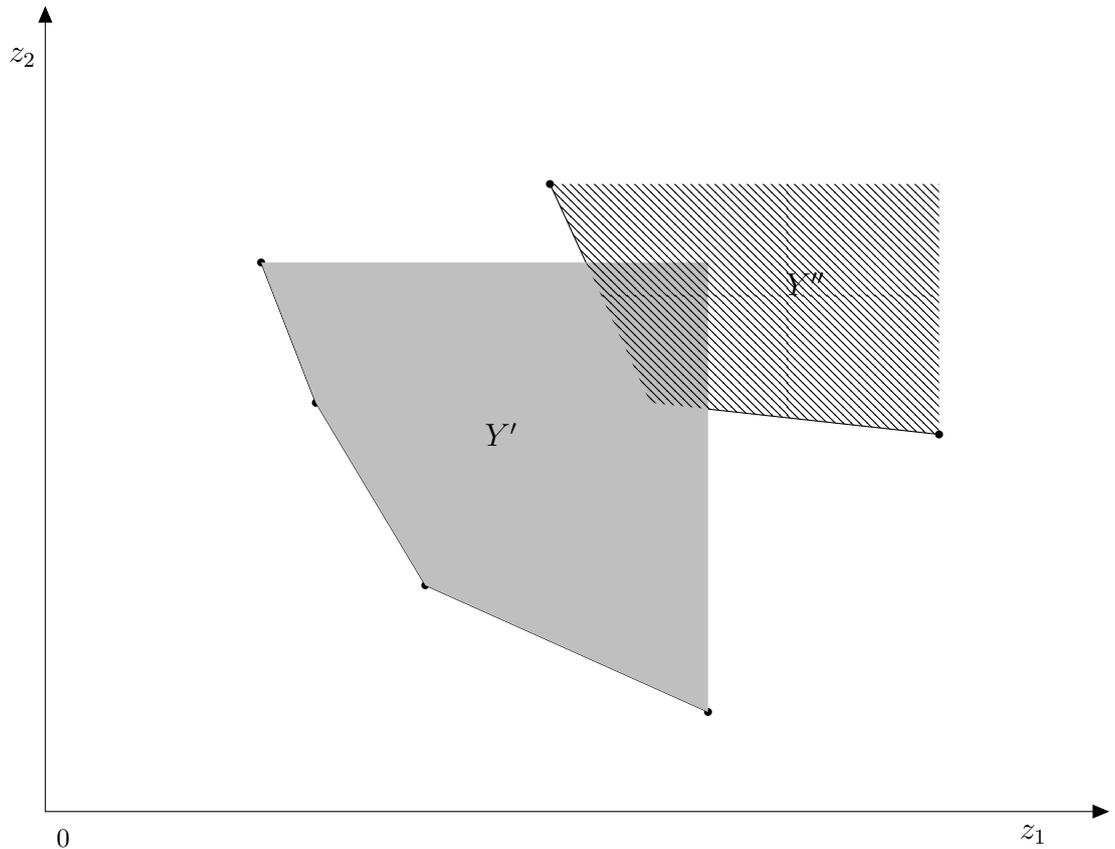


FIGURE 4.4 – Comparaison entre sous-problèmes.

L'utilisation des bornes multiobjectifs a permis de développer des algorithmes par séparation et évaluation performants : pour des problèmes en variables binaires dans [34, 11], des règles de dominance sont appliquées en haut de l'arborescence de recherche ce qui a permis d'accélérer la procédure de recherche.

Ces techniques d'optimisation ont également permis de développer des algorithmes performants pour plusieurs familles de problèmes combinatoires : un algorithme est proposé pour le problème de l'arbre couvrant minimum multi-objectifs dans [72], où à l'issue d'une phase d'initialisation, un algorithme par séparation et évaluation est appliqué afin de certifier l'exactitude de la frontière retournée. Visée et al. [80] ont proposé une méthode en deux phases pour résoudre le problème du sac à dos biobjectif. En seconde phase, les solutions non supportées sont calculées en lançant un branch and bound par zone de recherche susceptible d'en comporter.

Dans [49, 50, 79], des algorithmes par séparation et évaluation ont été proposés pour les problèmes linéaires mixtes multiobjectifs. Des algorithmes par séparation

et évaluation interactifs pour des problèmes linéaires multiobjectifs ont également été proposés dans [78, 48]

4.2 Description de la méthode

L'algorithme que nous proposons se base sur la technique du Branch and Bound. Il est à noter que le calcul d'une valuation d'un arc ne peut être effectué en amont de l'algorithme principal, comme il est besoin de connaître l'instant de départ d'un sommet avant d'évaluer l'ensemble de possibilités de transfert.

Afin d'améliorer la performance de l'algorithme, il convient de limiter le nombre d'appels à la fonction de calcul des coûts de transfert en montrant deux résultats :

- l'amélioration du coût de transfert suite à l'attente sur une orbite n'est plus assurée à partir d'une certaine durée d'attente que l'on déterminera.
- le coût de transfert diminue quand sa durée augmente. Néanmoins à partir d'une certaine durée, on ne gagne pas beaucoup en coût en tolérant de longues durées de transfert.

D'autre part, afin de limiter l'exploration des branches de l'arborescence de recherche qui ne mèneront qu'à des solutions dominées, des règles de dominance ont été conçues. La première règle consiste à ne pas développer les solutions partielles dominées sur le critère (coût, durée) par d'autres solutions visitant un ensemble plus grand de sommets. La deuxième règle s'appuie sur la construction d'un ensemble de solutions complètes pouvant être efficaces qui sera mis à jour chaque fois qu'une nouvelle solution est trouvée. Connaissant un minorant du coût et de la durée d'un transfert interorbital, on peut minorer le coût et la durée d'une solution partielle. Toute solution partielle dont l'extension coûtera et durera plus qu'une solution complète déjà trouvée peut ainsi être éliminée.

La manière d'explorer l'arbre des solutions joue un rôle important dans la vitesse d'apparition des solutions complètes non dominées. Nous avons ainsi testé et comparé différentes manières d'explorer l'arbre. L'ordre selon lequel les débris sont examinés influence notablement le temps de calcul. Sachant que le coût de transfert est dû au changement d'altitude et d'inclinaison de l'orbite, nous avons analysé la performance de l'algorithme en numérotant les débris dans l'ordre des inclinaisons des orbites puis dans celui des demi-grands axes. L'optimisation de l'algorithme a été ainsi faite à deux niveaux. D'abord on limite le nombre d'arcs entre un couple de sommets puis on évite l'exploration de certaines branches qui ne mèneront qu'à des solutions dominées. Une description détaillée de la méthode de résolution est fournie dans la section suivante.

4.2.1 Arborescence de recherche

Dans notre cas, un nœud dans le niveau k de l'arborescence de recherche correspond à une solution partielle qui visite k orbites en commençant par l'orbite initiale de la navette.

Les suivants d'un nœud sont les solutions qui visitent une orbite supplémentaire en empruntant le chemin qui mène au nœud père. Les suivants d'un nœud ne peuvent être générés que lorsque l'instant d'arrivée à ce nœud est connu. La taille de l'arbre de recherche peut ainsi s'avérer très grande. Il convient donc de limiter le domaine d'exploration de l'arbre de recherche. La réduction de l'arbre est faite à deux niveaux. Ainsi, nous minimisons le nombre des possibilités de transfert ce qui limite le nombre des suivants immédiats d'un nœud de l'arborescence, de plus, nous n'explorons pas les branches de l'arbre qui ne mènent pas à des solutions complètes efficaces.

Afin d'explorer l'arborescence de recherche, nous proposons une adaptation de la procédure du *Branch and Bound* à notre problème.

La séparation

La procédure de séparation consiste à choisir le nœud qui sera séparé en générant ses fils immédiats. Sachant que dans notre cas il existe différentes manières de se rendre à un nœud donné, il convient de choisir également la manière de se rendre à ce nœud. Ces deux choix définissent entièrement l'état suivant que l'on notera $s = (V_s, l, T(s), C(s))$, où V_s est l'ensemble des orbites visitées dans l'état s , l la dernière orbite visitée et $T(s)$ et $C(s)$ sont respectivement le temps et la durée du voyage jusqu'à l'arrivée à l'orbite l . Une stratégie de sélection indique le prochain nœud à séparer. Différentes stratégies de sélection ont été envisagées. La séparation de ce nœud revient à générer $A_{lj}(t_l)$ pour tout débris $j \notin V_s$.

L'évaluation

La procédure d'évaluation permet de couper quelques solutions partielles qui ne mèneront pas à des solutions complètes efficaces. Des règles de dominance sont utilisées afin de couper certaines branches de l'arborescence de recherche dont l'exploration s'avère sans intérêt ([7]). Etant donné une solution partielle s visitant k débris, une *extension* de s est une solution réalisable dont les k premières décisions sont celles prises dans s . Dans la suite, on notera par $Ext(s)$ l'ensemble des extensions possibles d'une solution partielle s .

4.2.2 Adaptation de la méthode

Dans cette partie, nous détaillons la procédure d'optimisation de la recherche dans l'arborescence. L'optimisation se fait sur deux niveaux. Du fait que l'algorithme d'exploration de l'arborescence des solutions met plus de temps à calculer les suivants d'un nœud qu'à explorer l'arbre de recherche, nous limitons le nombre de

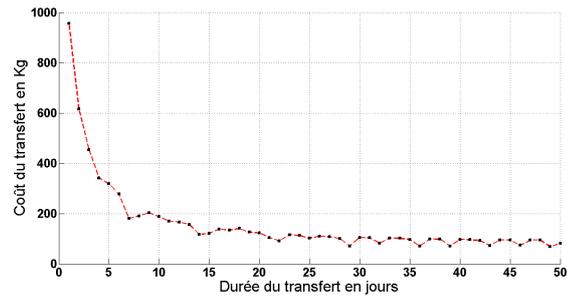
transferts élémentaires entre chaque paires d'orbites en limitant le temps d'attente sur l'orbite et les durées possibles de transfert. On note que, du fait que toutes les possibilités de transfert ne sont pas envisagées, le caractère totalement exact de la méthode n'est plus assuré. Cependant, les transferts qui ont été omis ne mèneront pas à des solutions significativement meilleures que les solutions obtenues par notre algorithme. Le paragraphe suivant compare la qualité des transferts retenus avec celle des transferts omis.

Limitation du nombre des suivants d'un nœud de l'arborescence

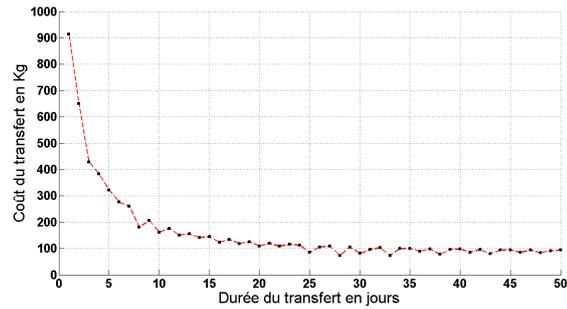
Le nombre des suivants d'un nœud peut être très élevé. Par exemple, si la durée maximale d'attente sur une orbite est de 6 heures et que le pas de discrétisation est fixé à 30 minutes, le véhicule orbital dispose de 13 positions de départ possibles. Si, de plus, on suppose que les durées de voyage possibles sont des multiples de la journée sans dépasser 3 mois, la navette disposera de 90 durées de transfert possibles. Ainsi il existe 1170 transferts possibles. Certes, certains de ces transferts s'avèreront dominés. Néanmoins l'algorithme ne peut détecter les transferts dominés qu'après avoir calculé le coût correspondant à chaque transfert, ce qui est extrêmement coûteux. Il est donc particulièrement intéressant de limiter le nombre de possibilités de transfert.

Dans les figures 4.5, nous avons considéré 5 paires de débris et tracé pour chacune de ces paires la variation du coût de transfert en fonction de sa durée. Dans les 5 instances, le coût diminue considérablement lorsque la durée du transfert augmente. Lorsque la durée de transfert dépasse les 10 jours, le gain sur le critère coût ne justifie plus la perte sur le critère durée. En effet, très peu de transferts de durée supérieure à un mois améliorent le coût, de plus cette amélioration est trop faible pour compenser la perte de temps engendrée. Afin de limiter le nombre des suivants d'un nœud de l'arbre de décision, on limitera donc la durée d'un transfert à dix jours.

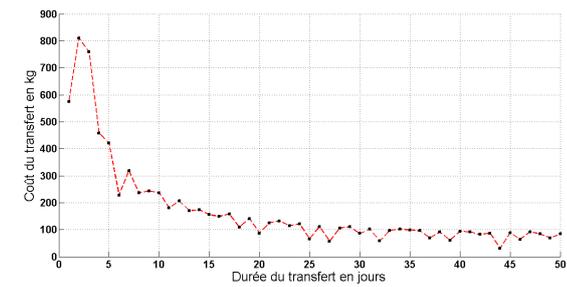
4. DÉVELOPPEMENT D'UNE PROCÉDURE DE RÉOLUTION POUR LE PROBLÈME BITDTSP



(a) Instance 1



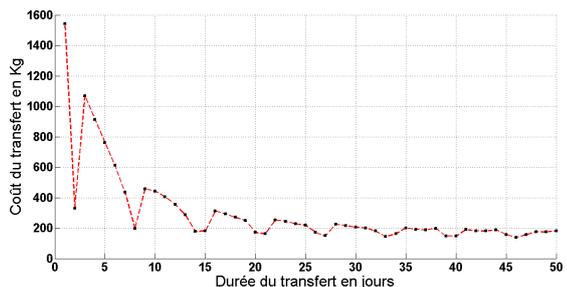
(b) Instance 2



(c) Instance 3



(d) Instance 4



(e) Instance 5

De même, le nombre de possibilités de départ influence considérablement le temps de calcul. Nous tolérons l'attente sur une orbite avant d'entamer un transfert en espérant que certains transferts, parmi ceux qui démarrent plus tard, seront nettement moins coûteux que les transferts qui ne nécessitent pas d'attente, de telle sorte qu'ils s'avèreront non dominés. Si l'attente ne conduit qu'à des transferts dominés, elle est alors sans aucun intérêt.

Nous avons choisi 8 couples d'orbites. Un couple correspond à une orbite de départ et une orbite destination. Nous avons, en outre, associé à chaque orbite de départ un instant de départ. Pour chaque couple, nous avons calculé les coûts et les durées de l'ensemble des transferts efficaces obtenus en tolérant une attente qui va jusqu'à dix fois la période de l'orbite de départ. Les périodes des orbites LEO étant de l'ordre de quelques heures, le temps d'attente maximum est ainsi de quelques dizaines d'heures. On calcule ensuite le pourcentage de ces transferts obtenus si le temps d'attente toléré était moins long. On constate que la totalité des transferts efficaces est obtenue au bout de trois périodes. Dans la table 4.1, on montre la progression de ce pourcentage lorsque la durée d'attente varie entre une période et trois fois la période de l'orbite de départ.

Numéro de paire	Le % des transferts ND obtenu après une attente d'une période	Le % des transferts ND obtenu après une attente de deux périodes	Le % des transferts ND obtenu après une attente de trois périodes
1 ^{ère} paire	100	100	100
2 ^{ème} paire	85.7	100	100
3 ^{ème} paire	87.5	87.5	100
4 ^{ème} paire	92.8	100	100
5 ^{ème} paire	100	100	100
6 ^{ème} paire	83.33	100	100
7 ^{ème} paire	75	100	100
8 ^{ème} paire	100	100	100

TABLE 4.1 – La progression du pourcentage des transferts efficaces.

La Table montre que la totalité des transferts efficaces est obtenue si l'attente est de 3 périodes. Le temps que l'algorithme met pour calculer le coût des transferts démarrant après trois périodes d'attente est donc perdu et n'apporte aucune amélioration de l'ensemble des transferts efficaces retenus.

De plus, pour toutes les instances étudiées, les trois quarts des transferts efficaces sont déjà obtenus après une attente d'une période. Pour comparer la qualité des transferts obtenus au bout d'une période avec ceux qui nécessitent plus d'attente, nous utilisons l'indicateur d'hypervolume (voir [5, 84]). Etant donné deux frontières de Pareto A et B , il s'agit de calculer $I_{HD}(A,B)$ l'hypervolume défini par A et qui n'est pas contenu dans la partie de l'espace dominée par B par rapport à un point de référence Z comme montré dans la figure 4.6.

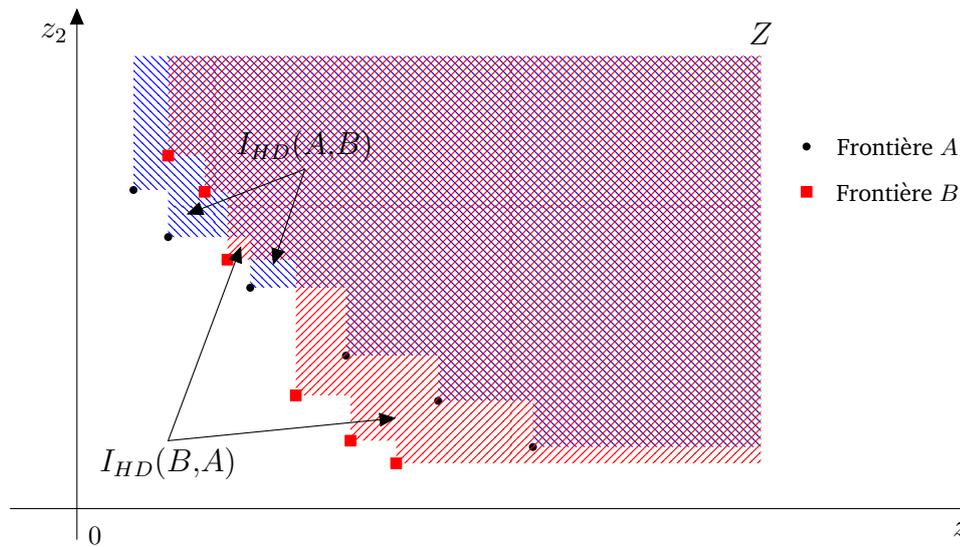


FIGURE 4.6 – Illustration de l'hypervolume associé à une frontière de Pareto.

Dans notre cas, nous comparons l'hypervolume de l'espace des objectifs dominé par les transferts efficaces obtenus après 3 périodes d'attente et celui dominé par les transferts obtenus après une période en choisissant comme référence le point Nadir, qui est le point ayant les pires des performances sur les deux critères. Les rapports entre les deux hypervolumes pour les 8 instances considérées sont présentés dans la Table 4.2.

Numéro de paire	Le rapport entre l'hypervolume obtenu après 1 période et celui obtenu après 3 périodes
1 ^{ere} paire	1
2 ^{eme} paire	0.97
3 ^{eme} paire	0.98
4 ^{eme} paire	0.99
5 ^{eme} paire	1
6 ^{eme} paire	0.97
7 ^{eme} paire	0.96
8 ^{eme} paire	1

TABLE 4.2 – Qualité des transferts obtenus après une période d'attente.

Dans la suite on ne considérera pas les transferts qui démarrent après une attente plus longue que la période de l'orbite de départ.

Utilisation des règles de dominances

Les règles de dominance sont utilisées afin d'épargner à l'algorithme l'exploration de certaines branches de l'arborescence de recherche inutilement. Etant donné une solution partielle s visitant k débris, une *extension* de s est une solution réalisable complète dont les k premières décisions sont celles prises dans la solution partielle s . Dans la suite, on notera $Ext(s)$ l'ensemble des extensions possibles pour une solution partielle s . Nous avons construit deux relations de dominance pour le problème de planification de mission de retrait des débris spatiaux.

Première relation de dominance D_1

La première relation de dominance utilisée compare les solutions partielles avec des solutions complètes préconstruites. Etant donné une solution partielle, l'idée est de minorer le coût et la durée de la meilleure solution issue de cette solution partielle. Si la performance au meilleur des cas d'une complétion de la solution partielle est moins bonne que celle d'une solution complète connue, alors toutes les extensions de cette solution partielle sont dominées par la solution complète connue. Par conséquent, il est inutile d'étendre cette solution partielle.

Pour appliquer la relation, on commence par construire un pool de solutions complètes. Naturellement, meilleure est la qualité des solutions du pool, moins de branches stériles seront explorées. Etant donné une solution partielle $s = (V_s, l, T(s), C(s))$, nous désirons minorer le coût et la durée de la meilleure solution issue de la solution partielle s . Le coût et la durée d'un transfert de Lambert varient conflictuellement. Ainsi un transfert peut durer très longtemps et ne pas nécessiter beaucoup de carburant comme il peut être effectué très rapidement moyennant un coût très élevé.

Dans les exemples numériques que nous avons utilisés, la durée minimale de transfert correspond au pas de discrétisation du temps de transfert qui est de $\delta t_{trans} = 1 \text{ jour}$. On ne considérera pas les transferts qui s'effectueront en une durée plus courte qu'une journée. Le problème de minimisation du coût d'un transfert de Lambert a été traité dans la littérature. [64] présentent une manière de calculer le coût minimal d'un transfert de Lambert étant donné deux orbites et deux positions de départ et d'arrivée bien définies sur ces deux orbites. Dans notre cas, étant donné une instance de plusieurs orbites, on cherche à calculer une valeur qui minore le coût de n'importe quel transfert entre n'importe quelle paire d'orbite. Pour ce faire, nous choisissons pour chaque couple d'orbites, deux positions très favorables pour le transfert de Lambert, soit deux positions peu inclinées par rapport aux plans des orbites.

Pour une instance de plusieurs débris on calcule le coût minimal pour effectuer un transfert entre chaque couple de débris de cette instance puis on prend la plus petite valeur parmi les minimas trouvés.

Ainsi la performance de la solution partielle s est au meilleur des cas sur les critères temps et coûts est

$$T^*(s) = T(s) + (|V| - |V_s| + 1) \times \delta t_{trans} + \sum_{i \in V \setminus V_s} \delta t_{serv_i} + \delta t_{serv_l} \text{ et}$$

$$C^*(s) = C(s) + (|V| - |V_s| + 1) \times \delta c_{min}.$$

Si x est une solution du pool dont le vecteur critère $(C(x), T(x))$ vérifie :

$$T(x) \leq T^*(s)$$

$$C(x) \leq C^*(s)$$

Toute *extension* de s est dominée par x et par conséquent le sommet de l'arborescence de recherche correspondant à la solution partielle s ne doit pas être développé.

Deuxième relation de dominance D_2

La deuxième relation de dominance teste si une solution partielle domine une deuxième. Etant données deux solutions partielles

$s = (V_s, l, T(s), C(s))$ et $s' = (V_{s'}, l', T(s'), C(s'))$ telles que

$$V_s \supseteq V_{s'}, \quad l = l', \quad T(s) \leq T(s') \quad \text{et} \quad C(s) \leq C(s')$$

En supposant qu'il est toujours mieux d'atteindre plus tôt l'orbite destination (hypothèse 1) et que les coûts et les durées des transferts vérifient l'inégalité triangulaire (hypothèse 2), on montre que toute extension de s' est dominée par au moins une extension de s .

Hypothèse 1. Cette hypothèse formule les résultats expérimentaux montrant que l'attente de plus qu'une période sur une orbite n'améliore que très peu la qualité des transferts possibles. Pour tout couple d'orbites i et j , soit t_i et t'_i deux instants possibles d'arrivée sur l'orbite i tels que la durée de temps écoulée entre t_i et t'_i est au moins la période de l'orbite i . Alors toute possibilité de transfert dont on dispose lorsqu'on atteint une orbite à t'_i est dominée par au moins une possibilité de transfert possible si on atteint i à t_i .

L'hypothèse peut être formulée de la manière suivante :

$$\forall w' \in I(t'_i), \quad \forall p' \in T \\ t'_i \geq t_i + P_i \Rightarrow \exists w \in I(t_i) \exists p \in T \text{ tel que}$$

$$\begin{cases} c_{ij}(t_i, w, p) \leq c_{ij}(t'_i, w', p') \\ d_{ij}(t_i, w, p) \leq d_{ij}(t'_i, w', p') \end{cases}$$

Où P_i dénote la période de l'orbite i . Il advient de cette hypothèse que l'attente qui dépasse une période est inutile, puisque tous les transferts qui démarreront après cette attente seront dominés.

Hypothèse 2. La deuxième hypothèse concerne l'inégalité triangulaire des coûts et durées des transferts. Etant donné trois orbites i , j et k , tout transfert permettant d'aller de i à k via j est dominé par au moins un transfert qui démarre avant et qui permet d'aller de i directement vers k .

On ne peut prouver la validité de cette hypothèse dans le cas général, cependant, expérimentalement, l'hypothèse est bien vérifiée.

En effet, considérons cinq triplets d'orbites de débris. Pour chaque triplet (i, j, k) , on calcule l'ensemble $A_{ik}(t_i)$ des transferts permettant à la navette d'aller de i vers k en ayant atteint l'orbite i à un instant t_i . On suppose ensuite qu'une deuxième navette fictive est arrivée à l'orbite i à des instants $\{t_i + k \times P_i/4, k \in \llbracket 0, 4 \rrbracket\}$.

Pour chacun de ces instants, on calcule les coûts et durées des trajectoires efficaces permettant à la deuxième navette d'aller de l'orbite i à l'orbite j puis à l'orbite k .

L'idée consiste à montrer que toute trajectoire effectuée par la deuxième navette est dominée par au moins un transfert de $A_{ik}(t_i)$ effectué par la première navette. Pour ce faire, on calcule le pourcentage de ces trajectoires dominées par au moins un élément de $A_{ik}(t_i)$. Les résultats montrent que 100% de ces transferts sont dominés ce qui légitime la deuxième hypothèse.

Cette hypothèse peut être formulée comme suit :

$$\forall w'_1 \in I(t'_i) \forall (p'_1, p'_2) \in T^2 \forall w'_2 \in I(t'_i + \delta t_{serv_i} + d_{ij}(t'_i, w'_1, p'_1)) \\ t_i \leq t'_i \Rightarrow \exists w \in I(t_i) \exists p \in T$$

$$\begin{cases} t_i + \delta t_{serv_i} + d_{ik}(t_i, w, p) \leq t'_i + \delta t_{serv_i} + d_{ij}(t'_i, w'_1, p'_1) + \delta t_{serv_j} + d_{jk}(t'_j, w'_2, p'_2) \\ c_{ik}(t_i, w, p) \leq c_{ij}(t'_i, w'_1, p'_1) + c_{jk}(t'_j, w'_2, p'_2) \end{cases}$$

Proposition 3. *Les hypothèses 1 et 2 sont suffisantes pour assurer la validité de la relation de dominance D_2 .*

Preuve 3. *Soient $s = (V_s, l, T(s), C(s))$ et $s' = (V_{s'}, l', T(s'), C(s'))$ deux solutions partielles telles que :*

$V_s \supseteq V_{s'}$, $l = l'$, $T(s) \leq T(s')$ et $C(s) \leq C(s')$. Supposons que le trajet est effectué dans la première solution partielle par une navette $N1$ et par une navette $N2$ dans la deuxième.

Cas où : $V_s = V_{s'}$

Soit e' une extension de s' .

Si dans e' , $N2$ quitte l'orbite l à un instant t tel que

$T(s) + \delta t_{serv_l} \leq t \leq T(s) + \delta t_{serv_l} + P_l$ alors $N1$ peut attendre sur l'orbite de l pendant une durée $t - (T(s) + \delta t_{serv_l})$, cette attente est possible par elle reste inférieure à la période de l'orbite l . Sachant que $N1$ et $N2$ ont dû effectuer un rendez-vous avec le débris d'orbite l , et qu'ils avancent à la même vitesse que le débris, à l'instant t elles vont se retrouver au même endroit. Ainsi, $N1$ pour effectuer la même suite de transferts que $N2$ et achever sa mission au même temps que $N1$ tout en conservant le gain qu'elle avait sur le coût. e' est donc une extension possible pour s aussi dans ce premier sous cas.

Si $N2$ part de l'orbite l à une date $t \geq T(s) + \delta t_{serv_l} + P_l$ alors conformément à l'hypothèse 1, il existe un transfert qui part de l à un instant $t' \in [T(s) + \delta t_{serv_l}, T(s) + \delta t_{serv_l} + P_l]$ et qui permet à la navette $N1$ de visiter le débris suivant visité dans e' plus tôt et avec un coût plus faible. Ce transfert nous ramène à la même situation initiale qui visite une orbite en plus. On réitère le même procédé tant que les débris n'ont pas tous été visités. On construit ainsi une extension de s qui domine e' .

Cas où : $V_{s'} \subsetneq V_s$

Soit $A = V_s - V_{s'}$ l'ensemble des débris visités dans s qui ne sont pas visités dans s' et B l'ensemble des débris qui doivent être visités aussi bien dans les extensions de s et dans celles de s' . Ainsi une extension de s' doit visiter l'ensemble des débris que contient $A \cup B$.

Soit e' une complétion de s' visitant l'ensemble $A \cup B$ dans un ordre donné. La complétion e' est de la forme $\{l, A_1, B_1, A_2, B_2, \dots, A_h, B_h\}$ avec $A = \bigcup_{i=1}^h A_i$ et $B = \bigcup_{i=1}^h B_i$. L'idée est de construire à partir de e' une extension de s qui coûte moins cher que e' et qui permet de terminer la mission plus rapidement que e' . La navette $N1$ atteint l à $t_l = t_0 + T(s)$, tandis que $N2$ l'atteint à $t'_l = t_0 + T(s')$. $t_l \leq t'_l$ alors conformément à l'hypothèse d'inégalité triangulaire, il existe une suite de transferts permettant à la navette $N1$ d'atteindre le premier élément visité dans B_1 plus rapidement que $N2$.

Les éléments de B_1 doivent être visités aussi bien dans e que dans e' , par un raisonnement analogue à celui du cas 1, on peut étendre e de telle sorte à ce que $N1$ atteigne le dernier élément de B_1 plus rapidement que $N2$. Ce qui nous ramène à une situation semblable à la situation initiale en ayant visité l'ensemble $A_1 \cup B_1$ en plus. En répétant ce processus, on construit une extension e de s qui permet d'atteindre le dernier élément à visiter dans B_h par $N1$ plus rapidement que par $N2$. Ce qui permet de valider la deuxième relation de dominance.

□

Conclusion du chapitre

Nous avons présenté dans ce chapitre l'arborescence de recherche des solutions. Cette arborescence étant assez riche, son exploration peut s'avérer très coûteuse en terme de temps de calcul. Il paraît indispensable de limiter la taille de cette arborescence afin d'explorer le moins possible de branches stériles. Dans cette perspective, nous avons réduit le nombre des suivants d'un nœud de l'arborescence limitant le temps d'attente sur chaque orbite et les durées de transfert possibles. En outre, les règles de dominance permettent de couper certaines branches de l'arborescence ce qui contribue à réduire sa taille globale. En plus de la taille, la manière d'explorer l'arborescence influence significativement le temps de recherche des solutions efficaces. Dans le chapitre suivant, on implémente différentes manières d'explorer l'arborescence et on présente les résultats obtenus avec chacune de ces manières.

Chapitre 5

Implémentation et expérimentation de la méthode

Sommaire

5.1	L'algorithme par séparation et évaluation	80
5.1.1	L'algorithme principal	80
5.1.2	Stockage des solutions partielles	81
5.2	Expérimentation et résultats	84
5.2.1	L'impact de l'ordre des solutions partielles et du numérotage de débris sur le temps de calcul	85
5.2.2	L'impact des relations de dominance	86
5.2.3	L'impact du pas de temps	86
5.3	Etude de quelques frontières de Pareto	87
5.3.1	Qualité des solutions du pool	87
5.3.2	L'impact de l'ordre de visite	90
	Conclusion	98

Nous présentons dans ce chapitre l'algorithme par séparation et évaluation adapté à notre problème. La manière d'explorer l'arborescence de recherche influence considérablement le temps de calcul. Nous implémenterons dans ce chapitre différentes manières d'explorer l'arborescence et comparons les résultats trouvés. Sont présentés ensuite les frontières de Pareto obtenues avec des instances de différentes tailles.

5.1 L'algorithme par séparation et évaluation

L'algorithme par séparation et évaluation commence par générer un ensemble de solutions complètes afin d'alimenter le pool de solutions et d'utiliser la première relation de dominance le plus tôt possible. Pour que l'utilisation de la première relation de dominance soit pertinente, il faudrait que le pool de solutions soit généré relativement rapidement tout en contenant des solutions de bonne qualité. Pour ce faire, nous construisons les solutions visitant les orbites dans leur ordre d'inclinaison. Cet ensemble de solutions servira à initialiser l'ensemble des solutions de pool. Cet ordre de visite des orbites est choisi parce qu'il épargne à la navette des changements d'inclinaison importants. Les transferts qui nécessitent de grands changements d'inclinaison coûtant plus cher que les autres, cet ordre devrait faire apparaître des solutions de coût relativement faible. En outre, les résultats numériques montrent que numérotter les orbites dans l'ordre de leur inclinaison améliore le temps de calcul de l'algorithme. Ainsi c'est cet ordre qui est utilisé pour numérotter les orbites initialement. L'ensemble de ces solutions est stocké dans une liste ND ordonnée selon l'ordre lexicographique des vecteurs (coût, durée) qui est mise à jour chaque fois qu'une nouvelle solution complète est construite de telle sorte qu'à la fin de l'algorithme elle ne contienne que des solutions non dominées.

5.1.1 L'algorithme principal

Une liste de travail L contient les solutions partielles en cours. L est initialisée à la solution partielle qui ne visite aucun élément $(\emptyset, 0, 0, 0)$. Tant que L est non vide, l'algorithme étend la première solution partielle dans la liste L . La solution partielle qui a été séparée est ensuite éliminée et ses suivants sont mis dans L . L'ordre selon lequel les solutions partielles sont insérées est crucial puisqu'il détermine la stratégie de parcours de l'arborescence des solutions partielles. Différents ordres ont été testés dans notre cas pour la liste L . Lorsqu'une solution est complète, elle est insérée dans ND , ensuite ND est mise à jour en supprimant toutes les solutions dominées par la nouvelle solution insérée. A la fin de l'algorithme, ND contiendra les vecteurs (coût, durée) correspondant à toutes les solutions efficaces. Pour toute solution partielle s , l'ensemble des suivants de s est noté Γ_s . Les différentes étapes de l'algorithme sont décrites dans l'algorithme 1.

Algorithm 1: L'algorithme de séparation et évaluation**Résultat:** L'ensemble des vecteurs non dominés ND **début**

```

 $L \leftarrow \{(\emptyset, 0, 0, 0)\}$ 
Initialiser  $ND$  à un ensemble de solutions candidates
tant que  $L \neq \emptyset$  faire
  /* On suppose que  $L = \{s_1, s_2, \dots, s_m\}$  */
  /* On suppose que  $ND = \{p_1, p_2, \dots, p_r\}$  */
  Sélectionner  $s \in L$ 
   $L \leftarrow L \setminus \{s\}$ 
  si  $s$  est complète alors
    Dominé  $\leftarrow$  faux
     $j \leftarrow 1$ 
    tant que  $j \leq r$  et  $C(p_j) \leq C(s)$  et Dominé = faux faire
       $j \leftarrow j + 1$ 
      si  $T(p_j) \leq T(s)$  alors
        Dominé  $\leftarrow$  Vrai
      si  $C(s) = C(p_j)$  et  $T(s) \leq T(p_j)$  alors
         $ND \leftarrow ND \setminus \{p_j\}$ 
      si Dominé = faux alors
         $ND \leftarrow ND \cup \{s\}$ 
        tant que  $j \leq r$  et  $T(s) \leq T(p_j)$  faire
           $ND \leftarrow ND \setminus \{p_j\}$ 
           $j \leftarrow j + 1$ 
    sinon
      Générer  $\Gamma_s$ 
      Mise à jour( $L, \Gamma_s, ND$ )
  retourner  $ND$ 

```

5.1.2 Stockage des solutions partielles

La fonction 'Mise à jour' insère les nouvelles solutions partielles générées dans la liste L si elles ne sont pas dominées par les solutions partielles déjà insérées dans L . L'insertion se fait en respectant les règles de dominance. Le nombre des tests de dominance à effectuer varie en fonction de l'ordre choisi. L'ordre de stockage doit être fait de telle sorte que le nombre de tests de dominance effectués soit faible tout en permettant l'apparition des meilleures solutions d'abord. Dans notre cas, trois ordres de stockage des solutions partielles ont été testés.

L'ordre lexicographique des vecteurs (coût, durée)

On notera par \preceq_1 l'ordre lexicographique des vecteurs critère. Il est défini pour tout couple de solutions s_1, s_2 comme suit. Les solutions $s_1 = (V_1, l_1, C(s_1), T(s_1))$ et $s_2 = (V_2, l_2, C(s_2), T(s_2))$ sont telles que $s_1 \preceq_1 s_2$ si et seulement si :

$C(s_1) < C(s_2)$ ou $(C(s_1) = C(s_2) \text{ et } T(s_1) \leq T(s_2))$

L'intérêt de cet ordre est que pour tout couple de solutions (s_1, s_2) tel que $s_1 \preceq_1 s_2$, s_2 ne peut dominer s_1 . D'autre part la première solution développée est celle qui coûte le moins. L'ordre favorise ainsi l'apparition des solutions les moins coûteuses d'abord. La manière d'insérer une nouvelle solution dans l'ensemble des solutions du pool est montrée dans la figure 5.1.

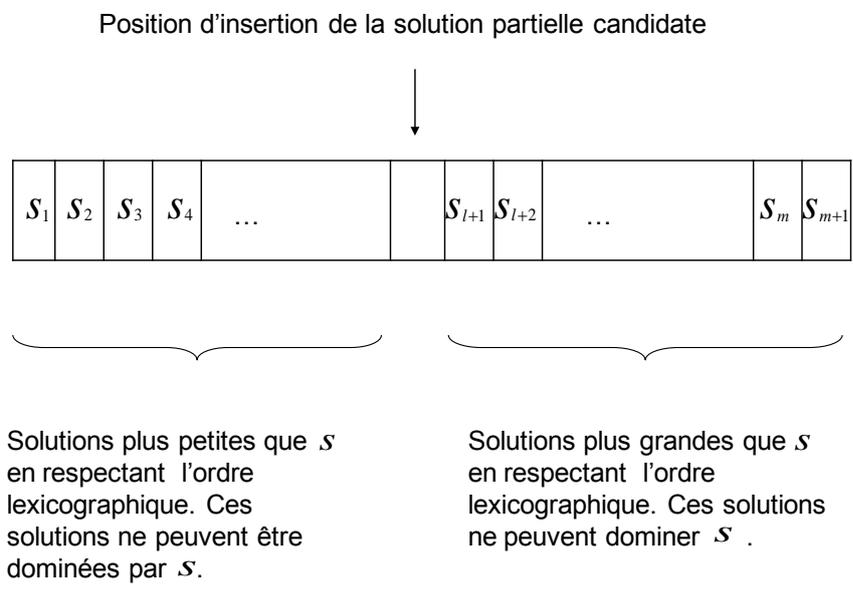


FIGURE 5.1 – L'insertion d'une nouvelle solution dans la liste de travail

Lorsque les solutions partielles sont stockées dans la liste L en respectant cet ordre, la fonction update, qui met à jour la liste des solutions partielles, est utilisée comme indiqué dans l'algorithme 2.

Algorithm 2: Procédure de mise à jour (L, L', ND)

Entrées: Deux listes de solutions partielles L et L' et la liste des solutions du pool ND

Sorties: La liste L mise à jour

début

/* On suppose que $\Gamma_{s_1} = \{s'_1, s'_2, \dots, s'_q\}$ et $ND = \{p_1, p_2, \dots, p_s\}$ */

pour $i \leftarrow 1$ **a** q **faire**

 Dominé \leftarrow *Faux*

$j \leftarrow 1$

tant que Dominé = *Faux* et $j \leq s$ **faire**

si $p_j D_1 s'_i$ **alors**

 Dominé \leftarrow *Vrai*

$j \leftarrow j + 1$

si Dominé = *Faux* **alors**

 /* On suppose que $L = \{s_1, s_2, \dots, s_r\}$ */

$j \leftarrow 1$

tant que Dominé = *Faux* et $C(s_j) \leq C(s'_i)$ et $j \leq r$ **faire**

si $s_j D_2 s'_i$ **alors**

 Dominé \leftarrow *Vrai*

$j \leftarrow j + 1$

si Dominé = *Faux* **alors**

 Insérer s'_i dans L à la position j

tant que $j \leq r$ **faire**

si $s'_i D_2 s_j$ **alors**

 Supprimer s_j de L

$j \leftarrow j + 1$

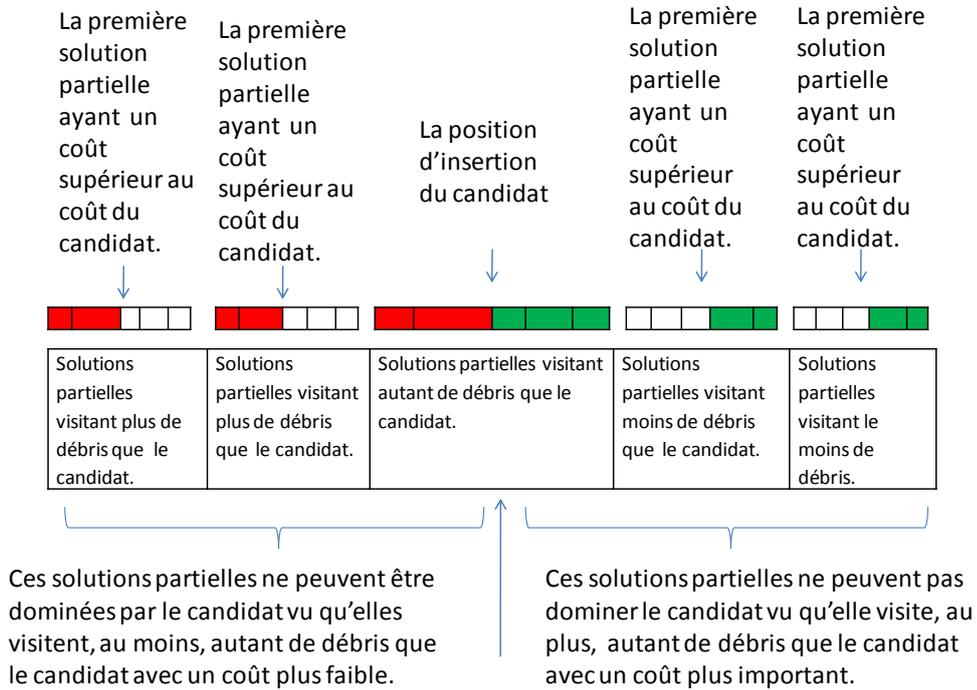
L'ordre lexicographique des vecteurs ($|V|, \text{coût}, \text{durée}$)

On notera \preceq_2 l'ordre lexicographique des vecteurs ($|V|, \text{coût}, \text{durée}$). Il est défini pour tout couple de solutions s_1, s_2 de la manière suivante : Les solutions $s_1 = (V_1, l_1, C(s_1), T(s_1))$ et $s_2 = (V_2, l_2, C(s_2), T(s_2))$ sont telles que $s_1 \preceq_2 s_2$ si et seulement si :

$|V_1| > |V_2|$ ou ($|V_1| = |V_2|$ et $C(s_1) < C(s_2)$) ou ($|V_1| = |V_2|$ et $C(s_1) = C(s_2)$ et $T(s_1) \leq T(s_2)$)

Lorsque les solutions sont stockées en respectant l'ordre \preceq_2 , la première solution partielle développée est celle qui coûte le moins parmi celles qui visitent le plus de débris. Stocker de cette manière les solutions partielles dans cet ordre permet d'explorer l'arbre de recherche en profondeur d'abord. Cet ordre réduit davantage le nombre de tests de dominance lors de la tentative d'insertion d'une nouvelle solution partielle car seules les solutions qui visitent plus de débris et qui coûtent

moins chers peuvent dominer la solution candidate. La figure 5.2 montre comment l'insertion d'un nouvel élément de la liste est effectuée en respectant cet ordre.



Dans la partie rouge de la liste on teste si le candidat est dominé, tandis que dans la partie verte on regarde si les solutions partielles déjà insérées sont dominées.

FIGURE 5.2 – L'insertion d'une nouvelle solution dans la liste de travail

L'ordre croissant des sommes des coûts et durées

On notera \preceq_3 l'ordre qui consiste à stocker les solutions partielles selon la somme des coûts et durées associés. Il est défini pour tout couple de solutions s_1, s_2 comme suit : Les solutions $s_1 = (V_1, l_1, C(s_1), T(s_1))$ et $s_2 = (V_2, l_2, C(s_2), T(s_2))$ sont telles que $s_1 \preceq_3 s_2$ si et seulement si :

$$C(s_1) + T(s_1) \leq C(s_2) + T(s_2)$$

De la même manière que le premier ordre, l'ordre \preceq_3 réduit le nombre de tests de dominance effectués entre les solutions partielles et il permet de stocker les solutions sans accorder à l'un des deux critères une importance supérieure que celle accordée à l'autre. La figure 5.3 montre comment l'insertion d'un nouvel élément de la liste est effectuée en respectant cet ordre.

5.2 Expérimentation et résultats

Dans cette étude, nous utilisons les TLEs décrits dans la paragraphe 2.3.1 et on utilise l'algorithme SGP4 afin de prévoir les positions des débris. Les buts des expérimentations sont donc :

- de déterminer le meilleur ordre pour ordonner les solutions partielles.

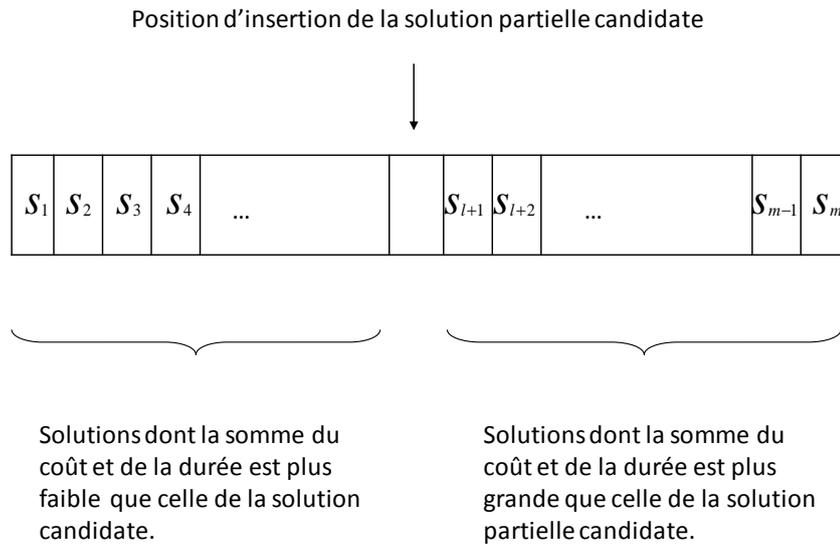


FIGURE 5.3 – L'insertion d'une nouvelle solution dans la liste de travail

- de déterminer le meilleur ordre pour numéroter les débris spatiaux.
- d'analyser l'impact de l'utilisation des relations de dominance.
- de comparer les frontières Pareto obtenues en utilisant plusieurs pas de temps.
- de visualiser certaines frontières Pareto et les comparer avec l'ensemble des solutions du pool initial.

5.2.1 L'impact de l'ordre des solutions partielles et du numérotage de débris sur le temps de calcul

Nous étudions ici la performance de l'algorithme lorsque les solutions partielles sont insérées dans les trois ordres précédemment présentés et nous numérotions les débris dans leur ordre d'inclinaison dans un premier temps puis dans leur ordre d'altitude de leurs orbites. Le pas de temps est fixé à 15 minutes. L'algorithme est testé sur 5 instances de différentes tailles et les résultats présentés correspondent aux moyennes de temps de calcul sur ces instances. Comme montré dans la Table 5.1, le meilleur ordre pour ordonner les solutions partielles est \preceq_2 qui est l'ordre lexicographique ($|V|$, coût, durée). Par ailleurs, le meilleur ordre de numérotation des orbites est l'ordre des inclinaisons des orbites. L'ordre lexicographique \preceq_2 favorise les solutions partielles visitant plus de débris ce qui accélère la construction des solutions complètes permettant ainsi d'améliorer la qualité des solutions de pool et de renforcer la première relation de dominance. Par ailleurs, numérotter les débris dans leur ordre d'inclinaison des orbites favorise les solutions les moins

taille de l'instance	\preceq_1				\preceq_2				\preceq_3			
	Ordre d'altitude		Ordre d'inclinaison		Ordre d'altitude		Ordre d'inclinaison		Ordre d'altitude		Ordre d'inclinaison	
	temps B & B	temps global	temps B & B	temps global	temps B & B	temps global	temps B & B	temps global	temps B & B	temps global	temps B & B	temps global
3	2	34	2	29	2	30	2	26	2	37	2	39
4	3	167	3	145	3	143	2	132	3	174	4	163
5	7	768	6	679	5	645	5	630	8	701	7	689
6	25	1568	23	1481	21	1206	18	1250	28	1591	27	1390
7	67	5423	51	5360	53	4493	52	4199	63	4839	61	5951
8	154	8456	116	7855	135	7813	105	7600	140	8810	154	8721
9	456	9874	449	9324	330	9804	371	9341	510	10549	511	10034
10	680	11569	632	11429	653	10345	658	10321	743	11600	734	11003

TABLE 5.1 – Temps de calcul obtenu en utilisant plusieurs ordres (en secondes).

Taille de l'instance	Avec D_1		Avec D_2		Avec $D = D_1 \cup D_2$	
	temps B & B	temps global	temps B & B	temps global	temps B & B	temps global
3	3	32	3	35	2	26
4	3	145	3	159	2	132
5	5	756	6	680	5	630
6	23	1373	22	1327	18	1250
7	67	5643	62	5472	52	4199
8	156	8340	134	7983	105	7600
9	463	10376	423	10562	371	9341
10	714	11540	702	11582	658	10321

TABLE 5.2 – Impact de l'utilisation des relations de dominance sur le temps de calcul (en secondes).

coûteuses afin qu'elles apparaissent en premier dans l'algorithme.

5.2.2 L'impact des relations de dominance

Nous utilisons ici l'ordre \preceq_2 pour ordonner les solutions partielles, et nous numérotions les débris selon l'ordre des inclinaisons de leurs orbites. Pour voir l'impact des relations de dominance, nous comparons le temps de calcul obtenu avec et sans utilisation des relations de dominance. Comme il est montré dans la Table 5.2, l'utilisation des relations de dominance diminue considérablement le temps de calcul. Le meilleur temps est obtenu lorsque les deux relations de dominance sont exploitées conjointement.

5.2.3 L'impact du pas de temps

Pour toutes les expérimentations faites dans cette partie, nous avons utilisé l'ordre \preceq_2 afin d'ordonner les solutions partielles. Nous avons utilisé également les deux relations de dominance et numéroté les débris dans l'ordre d'inclinaison de leurs orbites. Pour des instances de différentes tailles, nous avons généré l'ensemble des solutions non dominées obtenu et calculé le nombre de vecteurs non dominés et le temps de calcul correspondant. 5 instances par taille ont été considérées.

Nous cherchons à mesurer la qualité des frontières obtenues avec des pas de temps plus grands. Pour ce faire, nous calculons également l'ensemble des vecteurs non dominés obtenu en utilisant des pas de temps plus grands. L'idée est donc de

comparer la qualité de ces ensembles de vecteurs non dominés avec celle de la frontière obtenue avec le pas de temps de 15 minutes. Pour ce faire, nous utilisons l' ϵ -indicateur défini comme suit.

Pour définir cet indicateur établi dans [85], nous rappelons la définition de la relation d' ϵ dominance dans le cas bicritère.

Soit deux vecteurs critères $x = (x_1, x_2)$ et $x' = (x'_1, x'_2)$, deux éléments de l'image de l'ensemble de solutions réalisables d'un problème multicritère considéré, dans l'espace des critères inclus dans \mathbf{R}^2 . Soit s (respectivement s') une solution efficace correspondant au vecteur critère x (respectivement x'). On dit que s ϵ -domine s' qu'on note $s \leq_\epsilon s'$ ssi :

$$\begin{cases} x_1 \leq (1 + \epsilon) \times x'_1 \\ x_2 \leq (1 + \epsilon) \times x'_2 \end{cases}$$

Ainsi, pour comparer deux frontières de Pareto P et P' correspondant respectivement à deux ensembles de solutions S_1 et S_2 , obtenues avec deux pas de temps différents, nous calculons la plus petite valeur de ϵ pour laquelle tout élément de S_2 est ϵ -dominé par au moins un élément S_1 .

$$I_\epsilon(S_1, S_2) = \inf\{\epsilon \in \mathbf{R}^+ / \forall x' \in S_2 \exists x \in S_1 / x \leq_\epsilon x'\}$$

On le calcule de la manière suivante

$$I_\epsilon(S_1, S_2) = \max_{x' \in S_2} \min_{x \in S_1} \max\left(\frac{x_1}{x'_1} - 1, \frac{x_2}{x'_2} - 1\right)$$

Dans notre cas, nous avons utilisé des pas de temps de 25' et de 35'. Pour chaque instance, nous avons calculé la plus petite valeur de ϵ pour laquelle la frontière obtenue avec des pas de temps plus grands est une ϵ approximation. Nous retenons pour chaque taille d'instance les valeurs minimales, maximales et moyennes de ϵ .

Comme il est montré dans la Table 5.3, l'utilisation d'un pas de temps de 25' au lieu de 15' permet d'améliorer considérablement le temps de calcul sans une grande perte sur la qualité de la frontière de Pareto obtenue. Cependant, un pas de temps de 35' détériore la qualité de la frontière obtenue quoiqu'il améliore le temps de calcul.

On en conclut que, malheureusement, on ne peut se permettre d'utiliser des pas de temps plus grands.

5.3 Etude de quelques frontières de Pareto

5.3.1 Qualité des solutions du pool

Dans ce paragraphe, on se propose de visualiser certaines frontières Pareto obtenues avec des instances de différentes tailles. Dans chaque cas, nous comparons la qualité des vecteurs non dominés et du pool de solutions utilisées au départ par l'algorithme. Afin de comparer la qualité de ces deux ensembles, nous utilisons

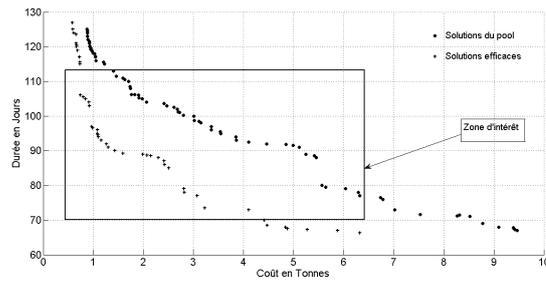
Taille de l'instance	15'		25'					35'				
	$ ND $	Temps de calcul	$ ND $	Temps de calcul	ϵ_{min}	ϵ_{moy}	ϵ_{max}	$ ND $	Temps de calcul	ϵ_{min}	ϵ_{moy}	ϵ_{max}
3	54	26	38	23	0.06	0.06	0.07	33	17	0.15	0.25	0.33
4	63	132	43	103	0.05	0.06	0.07	39	90	0.32	0.4	0.5
5	84	630	63	549	0.04	0.05	0.06	49	465	0.43	0.44	0.52
6	135	1250	127	1278	0.06	0.07	0.09	63	1130	0.33	0.35	0.41
7	169	4199	157	3745	0.03	0.07	0.08	112	3087	0.23	0.27	0.35
8	250	7600	237	7620	0.02	0.04	0.05	183	5145	0.37	0.42	0.45
9	343	9341	365	9273	0.05	0.06	0.06	312	7652	0.32	0.42	0.51
10	483	10321	389	10492	0.08	0.08	0.09	369	8622	0.31	0.34	0.51

TABLE 5.3 – Impact du pas de temps sur le temps de calcul.

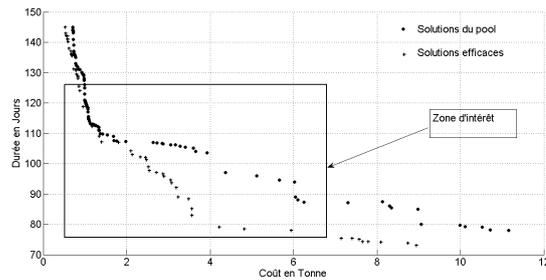
la même critère que dans la section précédente. Ainsi, nous calculons la plus petite valeur de ϵ telle que l'ensemble des solutions de pool est ϵ -approximation de l'ensemble final des vecteurs non dominés.

Dans la figure 5.4, nous présentons l'ensemble des vecteurs non dominés obtenu avec des instances de 6 à 9 débris et les performances des solutions de pool initialement générées pour chaque instance. Les symboles (+) correspondent aux solutions non dominées tandis que les points correspondent aux solutions de pool. La plus petite valeur de ϵ pour laquelle l'ensemble des solutions de pool est une ϵ -approximation de l'ensemble final des vecteurs non dominés est 0.49 pour l'instance de 6 débris, 0.33 pour celle à sept débris, 0.22 pour celle à huit débris 0.13 pour l'instance à neuf débris. L'ensemble des solutions du pool forment ainsi un bon départ pour l'algorithme, cependant on ne peut dire qu'ils constituent une bonne approximation de la frontière obtenue. L'application de l'algorithme s'avère ainsi nécessaire.

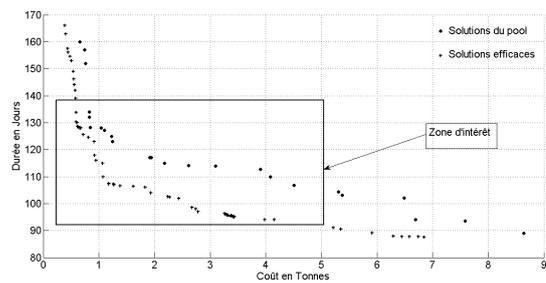
Certaines solutions non dominées obtenues correspondent à des compromis qui présentent une grande perte sur un critère sans assez améliorer le deuxième. Ces solutions sont quasi-dominées par d'autres solutions dans la frontière et sont peu intéressantes. La zone d'intérêt correspond aux solutions pour lesquelles la perte sur un critère est justifiée par le gain sur le deuxième. Cette zone est montrée dans la figure 5.4, elle est constituée de la partie la plus incurvée de la frontière. Il serait intéressant d'explorer plus précisément cette zone en affinant le pas de temps. Ainsi, en utilisant des pas de temps plus petits et en définissant une nouvelle borne supérieure permettant de couper des solutions partielles quasi-dominées, on peut améliorer les performances de l'algorithme et faire apparaître de nouvelles solutions dans la zone d'intérêt.



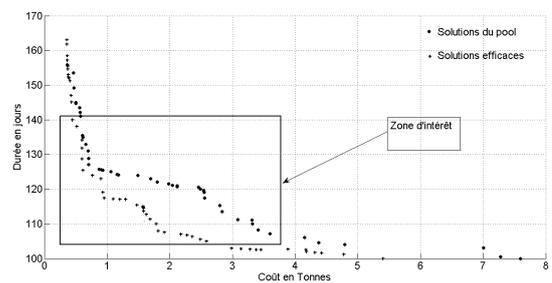
(a) L'ensemble des solution de pool et des vecteurs non dominés pour une instance de 6 débris.



(b) L'ensemble des solution de pool et des vecteurs non dominés pour une instance de 7 débris.



(c) L'ensemble des solution de pool et des vecteurs non dominés pour une instance de 8 débris.



(d) L'ensemble des solution de pool et des vecteurs non dominés pour une instance de 9 débris.

FIGURE 5.4 – Solutions de pool et vecteurs non dominés obtenus avec plusieurs instances

5.3.2 L'impact de l'ordre de visite

Dans cette partie, nous nous intéressons aux ordres qui apparaissent le plus dans les frontières de Pareto. Pour ce faire nous générons les frontières de Pareto obtenues en imposant des ordres de visites et on compare la qualité de ces frontières. Compte tenu de la combinatoire du problème, nous nous restreignons ici à des instances de 4 débris ce qui correspond à 24 ordres de visite possibles. Nous nous intéressons à l'instance suivante dont les caractéristiques sont présentées dans la table 5.4.

Numéro d'orbite	Demi grand axe en Km	Excentricité	Inclinaison (en rad)	RAAN (en rad)
1	7791.74	0.00	0.91	1.46
2	8488.37	0.00	0.91	4.25
3	7009.84	0.00	1.13	4.97
4	6979.66	0.00	1.72	0.06
5	7181.70	0.00	1.88	3.80

TABLE 5.4 – Caractéristiques de l'instance des orbites (RAAN = Right Ascension of the Ascending Node)

L'instance choisie est bien diversifiée comme le montre la figure 5.5

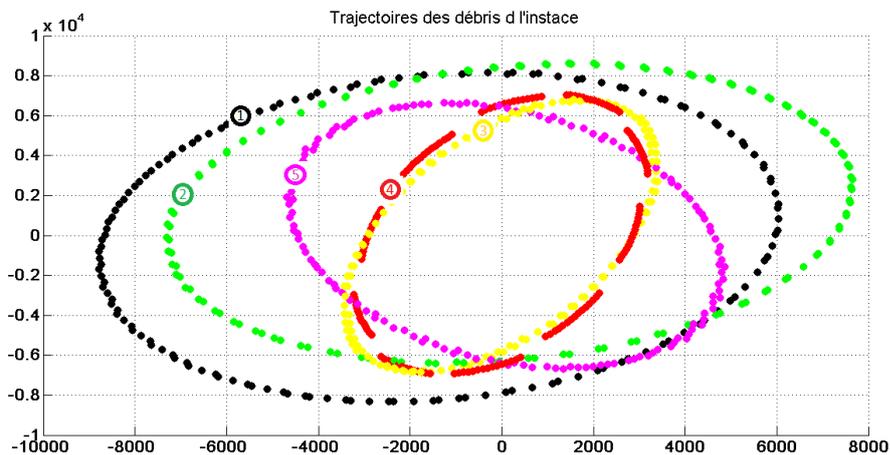


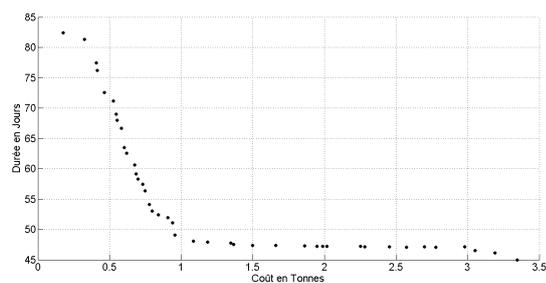
FIGURE 5.5 – Orbites des débris de l'instance

Nous avons calculé la somme des changements d'inclinaison et des changements d'altitude qu'induit chaque ordre de visite et présenté pour chaque ordre la frontière de Pareto correspondante dans la Table 5.5.

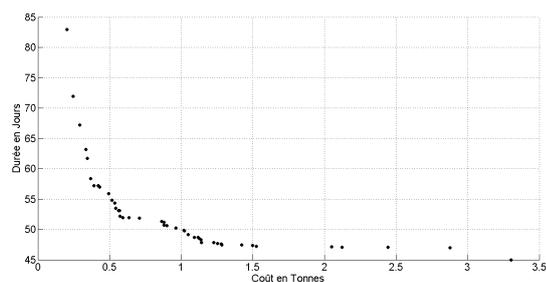
Ordre de visite	Total changements d'inclinaison (en rad)	Total changements d'altitude (en km)
1-2-3-4-5	1.94	3016
1-2-5-4-3	1.94	3016
1-5-4-3-2	1.94	3016
1-3-4-5-2	1.94	3016
1-2-3-5-4	1.94	3360
1-2-4-5-3	1.94	3360
1-4-5-3-2	1.94	3360
1-3-5-4-2	1.94	3360
1-4-5-2-3	2.38	4580
1-5-4-2-3	2.38	4580
1-3-2-4-5	2.38	4580
1-3-2-5-4	2.38	4580
1-5-3-4-2	3.12	3016
1-4-3-5-2	3.12	3016
1-2-4-3-5	3.12	3016
1-2-5-3-4	3.12	3016
1-5-2-3-4	3.56	4236
1-5-2-4-3	3.56	4236
1-4-3-2-5	3.56	4236
1-3-4-2-5	3.56	4236
1-5-3-2-4	3.56	4580
1-4-2-3-5	3.56	4580
1-4-2-5-3	3.56	4580
1-3-5-2-4	3.56	4580

TABLE 5.5 – Total des changements d'ordre et d'inclinaison pour chaque ordre de visite

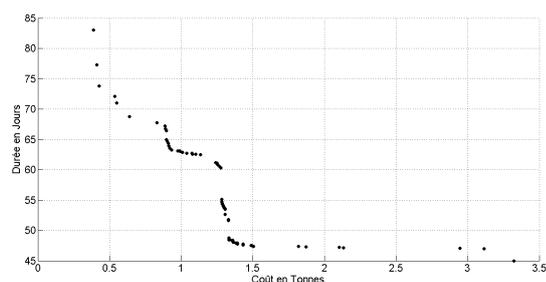
Les meilleures frontières sont obtenues en utilisant les ordres qui minimisent à la fois le total des changements d'inclinaison et celui des altitudes aussi, comme montré dans la figure 5.6.



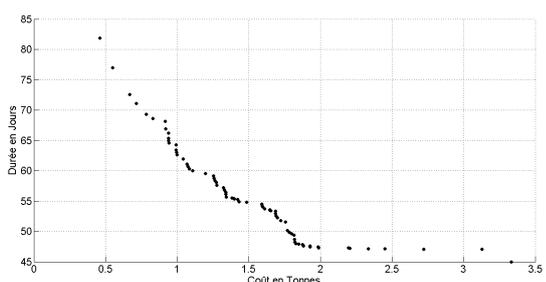
(a) Ordre de visite 1-2-3-4-5



(b) Ordre de visite 1-2-5-4-3



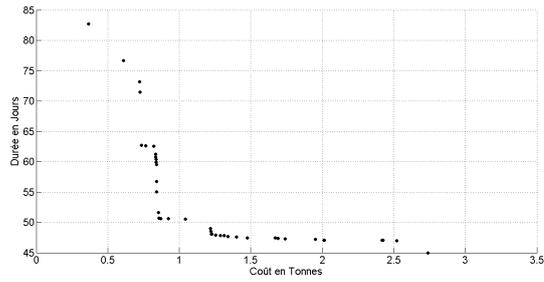
(c) Ordre de visite 1-5-4-3-2



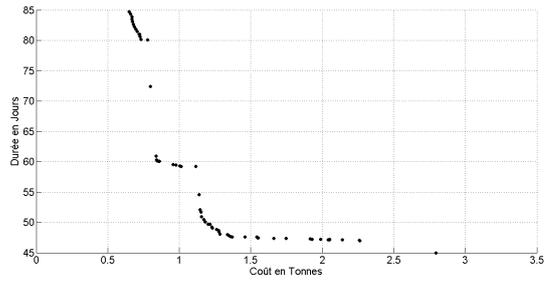
(d) Ordre de visite 1-3-4-5-2

FIGURE 5.6 – Ordres minimisant le changement d'inclinaison et d'altitude

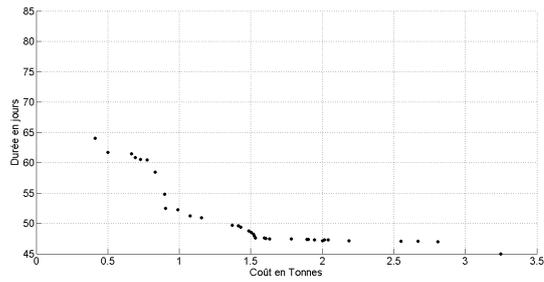
Viennent en deuxième position les ordres minimisant le total des changements d'inclinaison sans minimiser le total des changements d'altitude comme montré dans la figure 5.7, les frontières correspondantes suivantes ne sont pas comparables comme montré dans les figures 5.8 et 5.9.



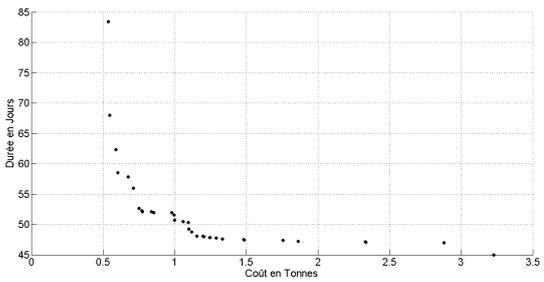
(a) Ordre de visite 1-2-3-5-4



(b) Ordre de visite 1-2-4-5-3



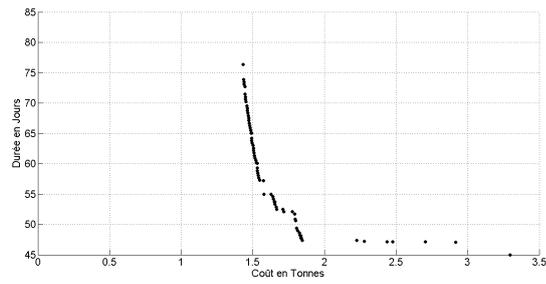
(c) Ordre de visite 1-4-5-3-2



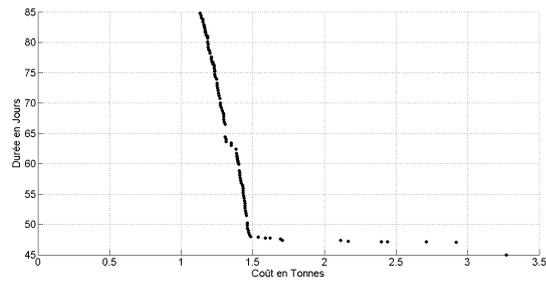
(d) Ordre de visite 1-3-5-4-2

FIGURE 5.7 – Ordres minimisant le changement d'inclinaison

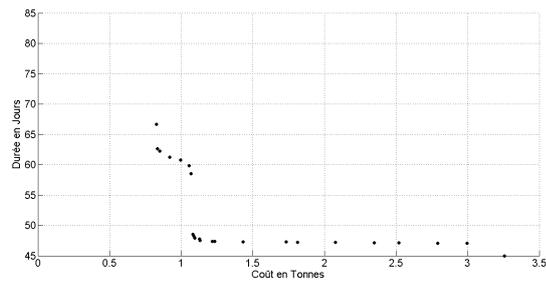
5. IMPLÉMENTATION ET EXPÉRIMENTATION DE LA MÉTHODE



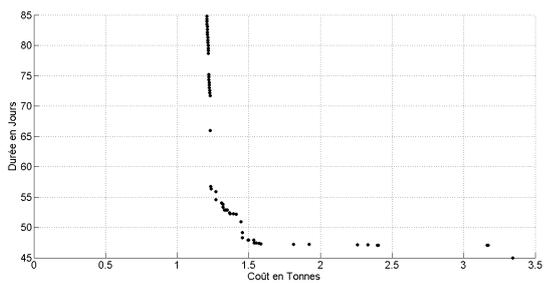
(a) Ordre de visite 1-4-5-2-3



(b) Ordre de visite 1-5-4-2-3

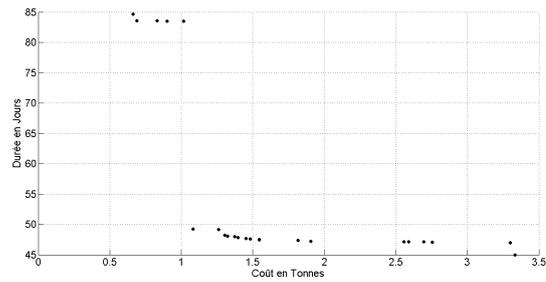


(c) Ordre de visite 1-3-2-4-5

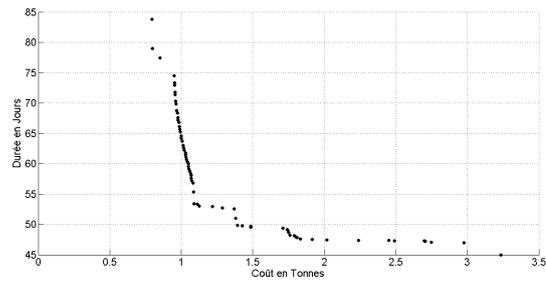


(d) Ordre de visite 1-3-2-5-4

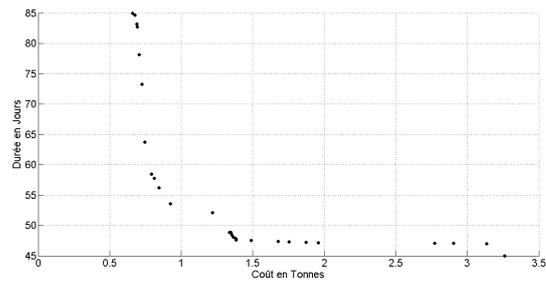
FIGURE 5.8 – Ordres réalisant un compromis entre le changement d'altitude et d'inclinaison



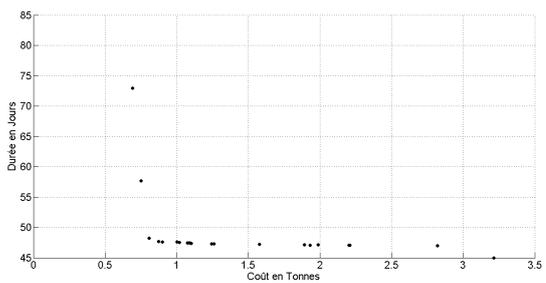
(a) Ordre de visite 1-5-3-4-2



(b) Ordre de visite 1-4-3-5-2



(c) Ordre de visite 1-2-4-3-5

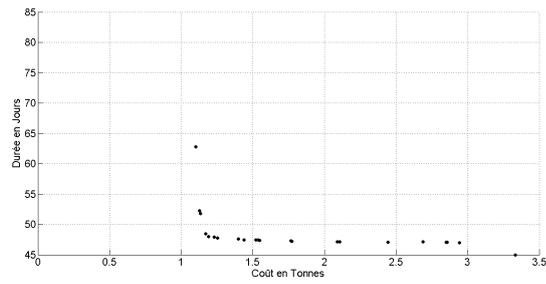


(d) Ordre de visite 1-2-5-3-4

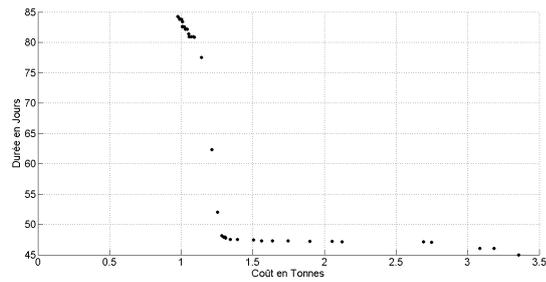
FIGURE 5.9 – Ordres réalisant un compromis entre le changement d'altitude et d'inclinaison

Les pires des frontières efficaces sont obtenues avec les ordres maximisant le total des changements d'inclinaison comme montré dans les figures 5.10 et 5.11.

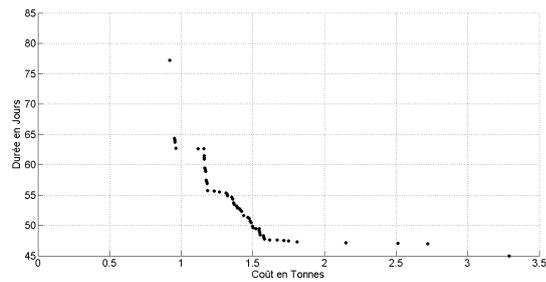
5. IMPLÉMENTATION ET EXPÉRIMENTATION DE LA MÉTHODE



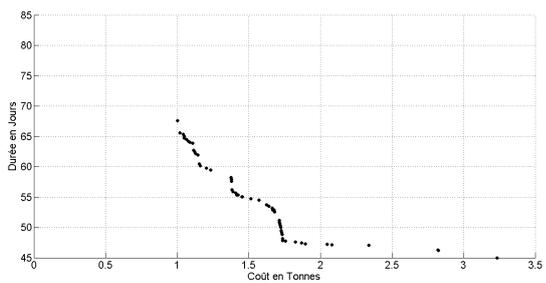
(a) Ordre de visite 1-5-2-3-4



(b) Ordre de visite 1-5-2-4-3

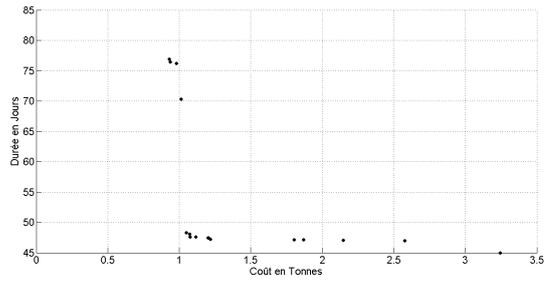


(c) Ordre de visite 1-4-3-2-5

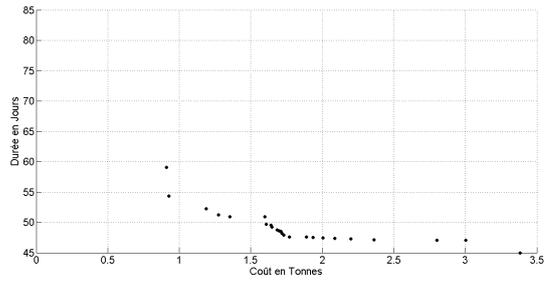


(d) Ordre de visite 1-3-4-2-5

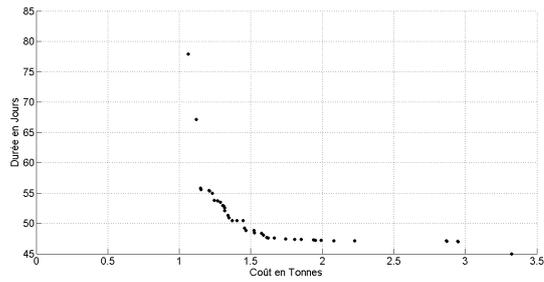
FIGURE 5.10 – Ordres maximisant le changement d'inclinaison



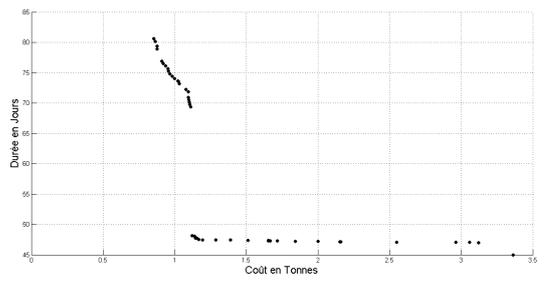
(a) Ordre de visite 1-5-3-2-4



(b) Ordre de visite 1-4-2-3-5



(c) Ordre de visite 1-4-2-5-3



(d) Ordre de visite 1-3-5-2-4

FIGURE 5.11 – Ordres maximisant le changement d'inclinaison

Il advient que toute solution efficace de notre problème est proprement efficace. Cependant, la valeur de M peut être très grande. Il convient de choisir les solutions efficaces pour lesquelles la valeur de M ne dépasse pas une valeur limite choisie par le décideur.

Conclusion du chapitre

L'application de l'algorithme par séparation et évaluation sur des instances de différentes tailles montre que le meilleur ordre pour stocker les solutions partielles parmi les trois ordres considérés est l'ordre lexicographique des vecteurs (nombre de débris visité croissant, coût décroissant, durée décroissante).

On montre aussi que le numérotation des débris influence notablement le temps de calcul, le meilleur ordre parmi les ordres testés est l'ordre des inclinaisons des orbites des débris. On note que l'utilisation des relations de dominance permet de couper certaines branches de l'arborescence de recherche améliorant ainsi la performance de l'algorithme de séparation et évaluation. L'utilisation de pas de temps plus grand détériore la qualité des frontières efficaces à chaque fois quoiqu'elle améliore notablement le temps de calcul. En perspective, il serait intéressant de considérer des pas de temps plus petit afin de zoomer sur les zones intéressantes des frontières efficaces. Nous avons proposé ainsi une méthode pour le calcul de nouvelles bornes sur le coût et la durée qui peuvent être utilisées pour extraire la zone d'intérêt dans l'ensemble des solutions efficaces.

Conclusion générale

Les débris spatiaux constituent une menace pour l'exploitation de l'espace. Leur nombre continuera à croître en raison des explosions et des collisions même si on arrête toute activité spatiale. La probabilité qu'une collision entre débris et satellites opérationnels se produise augmente par conséquent. Les manœuvres qu'effectuent les satellites afin d'éviter des collisions avec les débris ne suffisent plus comme le prouvent les collisions catastrophiques qui ont eu lieu depuis le début des activités spatiales.

Le seul moyen d'affronter cette situation consiste à éliminer ces débris. Le coût de désorbitation étant très élevé, il convient de déterminer les débris les plus dangereux et de les éliminer à moindre coût.

L'objectif de cette étude était de catégoriser les débris des orbites basses : LEO, en plusieurs catégories selon la priorité d'enlèvement et de planifier optimalement une mission spatiale pour le retrait des débris les plus dangereux. Nous avons ainsi proposé une approche multicritère pour affecter les débris à plusieurs catégories prédéfinies. Cette partie englobe le choix des critères caractérisant la priorité d'un débris à être éliminé, la description des différentes catégories, l'implémentation de la méthode de catégorisation et enfin son application sur l'ensemble des débris LEO. La méthode induit un préordre partiel sur l'ensemble des débris. L'idée est de traiter les débris dans l'ordre de priorité. Si le nombre de débris à traiter dépasse la capacité de traitement d'une navette, nous avons proposé des idées afin de traiter prioritairement les débris les plus urgents.

Nous avons développé, dans la deuxième partie de cette thèse, un algorithme pour planifier une mission de désorbitation des débris spatiaux les plus dangereux. Nous avons proposé une modélisation riche dans la mesure où nous prenons en considération le caractère elliptique des orbites des débris, différentes manières d'effectuer le transfert interorbital et les deux critères coût et durée de mission.

Nous avons développé une méthode de résolution basée sur l'approche branch and bound. L'utilisation des relations de dominance permet d'améliorer sensiblement les performances de l'algorithme.

Comme perspectives à ce travail, il serait intéressant d'étudier en profondeur la zone d'intérêt en utilisant des pas de temps plus fins. L'utilisation de plus petits pas augmenterait évidemment le temps de calcul. Une idée pour remédier à ce problème consiste à se focaliser sur une zone d'intérêt en introduisant des bornes sur les critères. Les performances de l'algorithme peuvent aussi être améliorées en connaissant, a priori, les préférences du décideur ce qui permettra de se focaliser

sur une zone de recherche plus restreinte.

Les résultats de l'application sur des instances de débris de différentes tailles confirment que le changement d'inclinaison est ce qui coûte le plus à la navette. En exploitant ce résultat nous avons pu proposer des modélisations pour le problème de traitement de débris par plusieurs navettes. Ces modélisations ont pour objectif de minimiser le coût des missions mais aussi de traiter les débris de même type par une même navette.

Les idées de planification présentées dans cette étude peuvent être appliquées à des problèmes voisins, nous citons par exemple le problème de service de satellites. On pourrait ainsi enrichir les modélisations proposées pour ce problème en proposant des modélisations en graphes dynamiques et de résoudre une version bi-objectif du problème.

On pourrait aussi raffiner la méthode de catégorisation des débris en considérant d'autres aspects qui peuvent s'avérer importants suite à l'évolution des recherches spatiales.

Bibliographie

- [1] H. Abeledo, R. Fukasawa, A. Pessoa, and E. Uchoa. The time dependent traveling salesman problem : Polyhedra and branch-cut-and-price algorithm. In *Experimental Algorithms*, pages 202–213. Springer, 2010.
- [2] Z. Aimin, G. Feng, and Z. Guixu. A decomposition based estimation of distribution algorithm for multiobjective traveling salesman problems. *Computers and Mathematics with Applications*, 66 :1857–1868, 2013.
- [3] Y.P Aneja and K.P.K. Nair. Bicriteria transportation problem. *Management Science*, 25(1) :73–78, 1979.
- [4] L. Anselmo, A. Rossi, and C. Pardini. updated results on the long-term evaluation of the space debris environment. *Advances in Space Research*, 23(1) :201–211, 1999.
- [5] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler. Hypervolume-based multiobjective optimization : Theoretical foundations and practical implications. *Theoretical Computer Science*, 425 :75–103, 2012.
- [6] G. Avanzini. A simple lambert algorithm. *Journal of guidance, control, and dynamics*, 31(6) :1587–1594, 2008.
- [7] C. Bazgan, H. Hugot, and D. Vanderpooten. Solving efficiently the 0-1 multiobjective knapsack problem. *Computers & Operations Research*, 36(1) :260–279, 2009.
- [8] R. Bellman. Dynamic programming treatment of the travelling salesman problem. *Journal of the ACM (JACM)*, pages 61–63, 1962.
- [9] J.F. Bérubé, M. Gendreau, and J.Y. Potvin. An exact-constraint method for bi-objective combinatorial optimization problems : Application to the traveling salesman problem with profits. *European Journal of Operational Research*, pages 39–50, 2009.
- [10] L.P. Bigras, M. Gamache, and G. Savard. The time-dependent traveling salesman problem and single machine scheduling problems with sequence dependent setup times. *Discrete Optimization*, 5(4) :685–699, 2008.
- [11] G.R. Bitran and J.M. Rivera. A combined approach to solve binary multicriteria problems. *Naval Research Logistics Quarterly*, 29(2) :181–201, 1982.

- [12] C. Bonnal, S. Missonnier, B. Malnar, and M. Bullock. Optimization of tethered de-orbitation of spent upper stages : the "mailman" process. In *4th European Conference on Space Debris*, volume 587, pages 361–366, 2005.
- [13] S. Bouveret. *Allocation et partage équitables de ressources indivisibles : modélisation, complexité et algorithmique*. PhD thesis, PhD thesis, Supaéro/University of Toulouse, 2007.
- [14] S. Bouveret, M. Lemaître, H. Fargier, and J. Lang. Allocation of indivisible goods : a general model and some complexity results. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 1309–1310. ACM, 2005.
- [15] M.M. Castronuovo. Active space debris removal a preliminary mission analysis and design. *Acta Astronautica*, 69(9-10) :848–859, 2011.
- [16] M. Cerf. Multiple space debris collecting mission debris selection and trajectory optimization. *Journal of Optimization Theory and Applications*, pages 761–796, 2013.
- [17] J.F. Cordeau and M. Maischberger. A parallel iterated tabu search heuristic for vehicle routing problems. *Computers & Operations Research*, 39(9) :2033–2050, 2012.
- [18] W. Dong and Z. Chang-yin. An accuracy analysis of the sgp4/sdp4 model. *Chinese Astronomy and Astrophysics*, 34(1) :69–76, 2010.
- [19] D. Dubois and H. Prade. Criteria aggregation and ranking of alternatives in the framework of fuzzy set theory. *TIMS Studies in the Management Sciences*, 20 :209–240, 1984.
- [20] M. Ehrgott and X. Gandibleux. Bounds and bound sets for biobjective combinatorial optimization problems. In *Multiple Criteria Decision Making in the New Millennium*, pages 241–253. Springer, 2001.
- [21] M. Ehrgott and X. Gandibleux. Bound sets for biobjective combinatorial optimization problems. *Computers & Operations Research*, 34(9) :2674–2694, 2007.
- [22] J.R. Figueira, L. Paquete, M. Simões, and D. Vanderpooten. Algorithmic improvements on dynamic programming for the bi-objective $\{0, 1\}$ knapsack problem. *Computational Optimization and Applications*, 56(1) :97–111, 2013.
- [23] J.C. Fodor and M Roubens. Fuzzy preference modelling and multi-criteria decision aid. *Kluwer Academic Publisher*, 1994.
- [24] K.R. Fox. *Production Scheduling on Parallel Lines with Dependencies*. Johns Hopkins University, 1973.
- [25] K.R. Fox, B. Gavish, and S.C. Graves. An n-constraint formulation of the (time-dependent) traveling salesman problem. *Operations Research*, pages 1018–1021, 1980.

- [26] V. Gabrel and D. Vanderpooten. Enumeration and interactive selection of efficient paths in a multiple criteria graph for scheduling an earth observing satellite. *European Journal of Operational Research*, 139(3) :533–542, 2002.
- [27] M.T. Godinho, L. Gouveia, and P. Pesneau. Natural and extended formulations for the time-dependent traveling salesman problem. In *Discrete Applied Mathematics*. Elsevier, 2011.
- [28] L. Gouveia and S. Voß. A classification of formulations for the (time-dependent) traveling salesman problem. *European Journal of Operational Research*, 83(1) :69–82, 1995.
- [29] M. Held and R.M. Karp. A dynamic programming approach to sequencing problems. *Journal of the Society for Industrial and Applied Mathematics*, 10(1) :196–210, 1962.
- [30] R.F. Hoelker and R. Silber. The bi-elliptical transfer between co-planar circular orbits. *Planetary and Space Science*, 7 :164–175, 1961.
- [31] F.R. Hoots and R.L. Roehrich. Models for propagation of NORAD element sets., 1980. Space track report no. 3.
- [32] H.C.X. Huawei. Research on algorithm of loopy lambert transfer in space rendezvous [j]. *Chinese Space Science and Technology*, 5 :001, 2004.
- [33] D. Kessler and B. Cour-Palais. Collision frequency of artificial satellites : The creation of a debris belt. *Journal of Geophysical Research : Space Physics (1978–2012)*, 83(A6) :2637–2646, 1978.
- [34] G. Kiziltan and E. Yucaoglu. An algorithm for multiobjective zero-one linear programming. *Management Science*, 29(12) :1444–1453, 1983.
- [35] K. Klamroth and M.M. Wiecek. Dynamic programming approaches to the multiple criteria knapsack problem. *Naval Research Logistics*, 47(1) :57–76, 2000.
- [36] G. Laporte. The vehicle routing problem : An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(3) :345–358, 1992.
- [37] G. Laporte, Y. Nobert, and S. Taillefer. A branch-and-bound algorithm for the asymmetrical distance-constrained vehicle routing problem. *Mathematical Modelling*, 9(12) :857–868, 1987.
- [38] W. Li. Finding pareto-optimal set by merging attractors for a bi-objective traveling salesmen problem. In *Evolutionary Multi-Criterion Optimization*, pages 797–810. Springer, 2005.
- [39] F. Liang, C. Liu, and R.J. Carroll. The metropolis-hastings algorithm. *Advanced Markov Chain Monte Carlo Methods : Learning from Past Samples*, pages 59–84, 2010.

- [40] J.C. Liou. Collision activities in the future orbital debris environment. *Advances in Space Research*, 38(9) :2102–2106, 2006.
- [41] J.C. Liou. A statistical analysis of the future debris environment. *Acta Astronautica*, 62(2-3) :264–271, 2008.
- [42] J.C. Liou. An active debris removal parametric study for leo environment remediation. *Advances in Space Research*, 47 :1865–1876, 2011.
- [43] J.C. Liou and N.L. Johnson. A sensitivity study of the effectiveness of active debris removal in leo. *Acta Astronautica*, 64(2-3) :236–243, 2009.
- [44] J.C. Liou, N.L. Johnson, and N.M. Hill. Controlling the growth of future leo debris populations with active debris removal. *Acta Astronautica*, 66(5-6) :648–653, 2010.
- [45] T. Lust and A. Jaszkievicz. Speed-up techniques for solving large-scale biobjective tsp. *Computers & Operations Research*, 37(3) :521–533, 2010.
- [46] D. Madakat, J.e Morio, and D. Vanderpooten. Biobjective planning of an active debris removal mission. *Acta Astronautica*, 84 :182–188, 2013.
- [47] C. Malandraki and R.B. Dial. A restricted dynamic programming heuristic algorithm for the time dependent traveling salesman problem. *European Journal of Operational Research*, 90(1) :45–55, 1996.
- [48] Odile Marcotte and Richard M Soland. An interactive branch-and-bound algorithm for multiple criteria optimization. *Management Science*, 32(1) :61–75, 1986.
- [49] G. Mavrotas and D. Diakoulaki. A branch and bound algorithm for mixed zero-one multiple objective linear programming. *European Journal of Operational Research*, 107(3) :530–541, 1998.
- [50] G. Mavrotas and D. Diakoulaki. Multi-criteria branch and bound : A vector maximization algorithm for mixed 0-1 multiple objective linear programming. *Applied mathematics and computation*, 171(1) :53–71, 2005.
- [51] P. McKenzie. Advanced 3d flash lidar sensor. Glasgow, 2008.
- [52] J.J. Miranda-Bront, I. Méndez-Díaz, and P. Zabala. Facets and valid inequalities for the time-dependent travelling salesman problem. *European Journal of Operational Research*, 2013.
- [53] N.Z. Miura. Comparison and design of simplified general perturbation models (sgp4) and code for nasa johnson space center, orbital debris program office. 2009.
- [54] M. Mizumoto. Pictorial representations of fuzzy connectives, part i : cases of t-norms, t-conorms and averaging operators. *Fuzzy Sets and Systems*, 31(2) :217–242, 1989.

- [55] J. Morio, R. Pastel, and F. Le Gland. Missile target accuracy estimation with importance splitting. *Aerospace Science and Technology*, pages 40–44, 2011.
- [56] T. Mosher. Spacecraft design using a genetic algorithm optimization approach. In *Aerospace Conference, 1998 IEEE*, volume 3, pages 123–134. IEEE, 1998.
- [57] L. Paquete, M. Chiarandini, and T. Stützle. Pareto local optimum sets in the biobjective traveling salesman problem : An experimental study. In *Metaheuristics for Multiobjective Optimisation*, pages 177–199. Springer, 2004.
- [58] L. Paquete and T. Stützle. A two-phase local search for the biobjective traveling salesman problem. In *Evolutionary Multi-Criterion Optimization*, pages 69–69. Springer, 2003.
- [59] L. Paquete and T. Stützle. Design and analysis of stochastic local search for the multiobjective traveling salesman problem. *Computers & Operations Research*, 36(9) :2619–2631, 2009.
- [60] V. Th Paschos et al. An overview on polynomial approximation of np-hard problems. *The Yugoslav Journal of Operations Research ISSN : 0354-0243 EISSN : 2334-6043*, 19(1), 2009.
- [61] W. Peng, Q. Zhang, and H. Li. Comparison between moea/d and nsga-ii on the multi-objective travelling salesman problem. In *Multi-Objective Memetic Algorithms*, pages 309–324. Springer, 2009.
- [62] J-C. Picard and M. Queyranne. The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling. *Operations Research*, 26(1) :86–110, 1978.
- [63] D. Portree and J. Loftus. Orbital debris and near-earth environmental management. *Chronology. JSC. NASA, Houston, Texas, USA*, 1993.
- [64] A. Prado and R Broucke. The minimum delta-v lambert’s problem. *Controle e Automação*, 7(2) :84–90, 1996.
- [65] John E. Prussing. Simple proof of the global optimality of the hohmann transfer. *Journal of guidance, control, and dynamics*, 15(4) :1037–1038, 1992.
- [66] A. Ravanbakhsh and S. Franchini. Multiobjective optimization applied to structural sizing of low cost university-class microsatellite projects. *Acta Astronautica*, 79 :212–220, 2012.
- [67] E. Riddle. Use of optimization methods in small satellite systems analysis. *Proceedings of 12th AIAA/USU Conference on small Satellites*, 1998.
- [68] W. Schall. Space-based laser debris mitigation. *Acta Astronautica*, 24 :343–351, 1991.
- [69] H. Schmitz and S. Niemann. A bicriteria traveling salesman problem with sequence priorities. *Metaheuristics in the Service Industry*, pages 1–14, 2009.

- [70] J. Schneider. The time-dependent traveling salesman problem. *Physica A : Statistical Mechanics and its Applications*, 314(1) :151–155, 2002.
- [71] I. M. Sobol. *A Primer for the Monte Carlo Method*. CRC Press, Boca Raton, Fl., 1994.
- [72] F. Sourd and O. Spanjaard. A multiobjective branch-and-bound framework : Application to the biobjective spanning tree problem. *INFORMS Journal on Computing*, 20(3) :472–484, 2008.
- [73] T.R. Stodgell and D.B. Spencer. Satellite rendezvous tours using multiobjective evolutionary optimization. *Advances in the astronautical sciences*, 129(3) :2069–2094, 2008.
- [74] E.R. Taylor. Evaluation of multidisciplinary design optimization techniques as applied to spacecraft design. In *Aerospace Conference Proceedings, 2000 IEEE*, volume 1, pages 371–384. IEEE, 2000.
- [75] Russ J. Vander W. and N.V. Sahinidis. Heuristic bounds and test problem generation for the time-dependent traveling salesman problem. *Transportation Science*, 29(2) :167–183, 1995.
- [76] Russ J. Vander W. and N.V. Sahinidis. An exact solution approach for the time-dependent traveling-salesman problem. *Naval Research Logistics (NRL)*, 43(6) :797–820, 1996.
- [77] B. Villarreal and M.H. Karwan. Multicriteria integer programming : A (hybrid) dynamic programming recursive approach. *Mathematical programming*, 21(1) :204–223, 1981.
- [78] Bernardo Villarreal, Mark H Karwan, and Stanley Zionts. An interactive branch and bound procedure for multicriterion integer linear programming. In *Multiple Criteria Decision Making Theory and Application*, pages 448–467. Springer, 1980.
- [79] T. Vincent, F. Seipp, S. Ruzika, A. Przybylski, and X. Gandibleux. Multiple objective branch and bound for mixed 0-1 linear programming : Corrections and improvements for the biobjective case. *Computers & Operations Research*, 40(1) :498–509, 2013.
- [80] M Visée, J Teghem, M Pirlot, and EL Ulungu. Two-phases method and branch and bound procedures to solve the bi-objective knapsack problem. *Journal of Global Optimization*, 12(2) :139–155, 1998.
- [81] B.J. Wadsley and R.G. Melton. Optimal visitation order for spacecraft servicing missions. *Advances in the astronautical sciences*, 129(3) :2705–2720, 2008.
- [82] R. Walker and C. Martin. Cost-effective and robust mitigation of space debris in low earth orbit. *Advances in Space Research*, 34 :1233–1240, 2004.

- [83] Z. Yan, L. Zhang, L. Kang, and G. Lin. A new moea for multi-objective tsp and its convergence property analysis. In *Evolutionary Multi-criterion Optimization*, pages 342–354. Springer, 2003.
- [84] E. Zitzler and S. Künzli. Indicator-based selection in multiobjective search. In *Parallel Problem Solving from Nature-PPSN VIII*, pages 832–842. Springer, 2004.
- [85] E. Zitzler, L. Thiele, M. Laumanns, C. M Fonseca, and V.G. Da Fonseca. Performance assessment of multiobjective optimizers : An analysis and review. *Evolutionary Computation, IEEE Transactions on*, 7(2) :117–132, 2003.

Approches multicritères pour le traitement des débris spatiaux

Les débris spatiaux constituent une menace pour l'exploration et l'exploitation de l'espace. Leur nombre ne cesse d'augmenter et continuera à grandir même si on arrête toute activité spatiale, augmentant ainsi la probabilité d'entrer en collision avec un satellite actif. Le retrait des débris s'avère le seul moyen de protéger ces satellites. Le nombre des débris spatiaux étant très élevé, il convient préalablement de repérer les plus dangereux.

Dans la première partie de la thèse, nous avons élaboré une approche multicritère afin de classer les débris selon leur degré de priorité d'enlèvement. Les débris de la classe la plus prioritaire, feront l'objet d'une mission spatiale de retrait de débris. La planification d'une telle mission est étudiée dans la deuxième partie de la thèse. Elle doit être réalisée en minimisant deux critères : le coût de la mission ainsi que la durée nécessaire pour traiter tous les débris.

La navette se déplace d'une orbite à une autre, traite les débris un par un puis retourne à son orbite initiale. Etant donné que le transfert entre deux orbites de débris peut être effectué de multiples façons, chacune correspondant à un compromis possible entre la durée et le coût de transfert, et que ces coûts et durées dépendent des moments de départ et d'arrivée sur les orbites, l'ensemble des solutions réalisables est défini sur un multigraphe dynamique orienté. Un tour dans un tel graphe définit un scénario de mission possible. Il s'agit de trouver l'ensemble des tours non dominés dans un tel multigraphe. Ceci revient à résoudre un problème de voyageur de commerce biobjectif et dépendant du temps.

Nous avons développé un algorithme basé sur la technique de séparation et évaluation pour restituer l'ensemble de ces tours. L'optimisation de l'algorithme est faite sur deux niveaux :

- On limite le nombre des transferts possibles entre deux orbites en évitant de calculer le coût pour les transferts qui s'avèreraient dominés.
- Des règles de dominance sont utilisées pour couper certaines branches de l'arborescence de recherche qui ne mèneront pas à des solutions efficaces.

Des résultats expérimentaux illustrent l'efficacité de la procédure.

Mots-clés : BRANCH AND BOUND ; DEBRIS SPATIAUX ; ANALYSE MULTICRITERE ; PROGRAMMATION DYNAMIQUE ; CATEGORISATION ; PLANIFICATION

Multicriteria approaches for active debris removal

Space debris are a threat for the space exploitation and exploration. Their number will continue to increase even if we stop all space activities, making collisions between debris and operational satellites more likely to happen. Debris removal proves necessary to protect active satellites. Since the number of space debris is very high, we should first deal with the most dangerous ones.

In the first part of this thesis, we have developed a multicriteria approach to categorize debris depending on their removal priority degree. Debris belonging to the most urgent category will be dealt with during a space mission. The planning of such a space mission is studied in the second part of this thesis. The planning should be designed while optimizing two criteria: mission cost and mission duration.

The space shuttle moves from an orbit to another, deals with the debris one by one and then returns to its initial orbit. There are several ways to perform a transfer between two orbits. Each transfer possibility corresponds to a possible compromise between cost and transfer duration. Transfer costs and durations depend on the moment the shuttle reaches and leaves each orbit. The set of feasible solutions is thus defined on a dynamic digraph. Each tour on such a graph defines a possible mission scenario. The objective is to find the set of non-dominated tours, which can be done by solving a time dependent biobjective traveling salesman problem.

We have developed a branch and bound algorithm to deal with this problem. The algorithm optimization is done on two levels:

- We limit the number of transfer possibilities between a pair of orbits to avoid computing costs of some transfers that will be dominated.
 - Fathoming rules are used to cut some branches that will not lead to efficient solutions.
- Experimental results illustrate the approach efficiency.

Keywords : SPACE DEBRIS ; CATEGORIZATION ; SPACE MISSION PLANNING