

N° d'ordre :0212022-D/MT

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie Houari Boumediène USTHB
Faculté des Mathématiques



Thèse de DOCTORAT EN SCIENCES
présentée pour l'obtention du grade de DOCTEUR
par MAACHOU NACÉRA

THÈME

UNE CONTRIBUTION À LA RÉOLUTION DES PROBLÈMES DE
PROGRAMMATION NON LINÉAIRE À DEUX NIVEAUX

Soutenue publiquement le 24/03/2022, devant le jury composé de :

M. Moncef.ABBAS	Professeur	USTHB	Président
M. Mustapha.MOULAI	Professeur	USTHB	Directeur de thèse
M. Mohamed Saïd.RADJEF	Professeur	U.Bejaïa	Examineur
M. Mouaouia Cherif.BOUZID	Maitre de conférences/A	ENST	Examineur
M. Nedjmeddine.KANTOUR	Maitre de conférences/B	USTHB	Invité

Résumé

Cette thèse porte sur l'optimisation non linéaire à deux niveaux. Les principaux résultats obtenus se scindent en deux parties : Nous avons d'abord développé une méthode pour résoudre un problème fractionnaire quadratique/quadratique à deux niveaux (Maachou N & M. Moulaï M (2015)). La méthode proposée est basée sur les conditions de Karush-Kuhn-Tucker, pour laquelle le problème du second niveau est remplacé par ses conditions d'optimalité de (KKT). Il en résulte un programme mathématique non linéaire à un seul niveau (*PNL*), où la résolution de ce problème non linéaire s'appuie essentiellement sur des résultats trouvés dans un de nos travaux cités dans la thèse (Maachou N & Moulaï M (2008)), le travail consiste à résoudre un problème fractionnaire quadratique discret, la méthode est basée sur la résolution d'un problème fractionnaire linéaire, des conditions d'optimalités sont déterminées afin de résoudre le problème (*PNL*).

La seconde partie de notre contribution consiste à mettre au point une nouvelle méthode exacte pour résoudre le problème quadratique indéfini à deux niveaux en nombres entiers, dit IIQBP. En effet, on résout le problème quadratique indéfini du premier niveau, dont la solution optimale appartient à l'ensemble des solutions efficaces du problème bi-critère associé. L'ensemble des solutions efficaces est déterminé par la méthode Branch & Cut. La méthode est basée sur la procédure Branch & Bound pour trouver une solution entière, une coupe efficace est utilisée afin d'éliminer les solutions dominées du domaine réalisable. La solution optimale entière obtenue du premier niveau est ensuite testée pour l'optimalité du problème principal en résolvant le problème du second niveau, l'algorithme exploité pour la résolution du premier niveau est retenu pour résoudre le second niveau. Si cette solution n'est pas optimale pour le problème principal, on l'élimine du domaine, en ajoutant la coupe de Dantzig au problème du premier niveau et on continue l'exploration. L'algorithme proposé est une méthode exacte, qui fournit la solution optimale si elle existe. La solution optimale entière du problème IIQBP est obtenue en un nombre fini d'itérations. Afin de mesurer l'efficacité de notre algorithme, nous avons construit un tableau de valeurs pour déterminer le temps de calcul pour deux exemples et pour d'autres exemples de plus grandes dimensions. Le temps d'exécution était raisonnable. Un programme informatique a été développé sous environnement Matlab.

Abstract

This thesis deals with bi-level nonlinear optimization. The main results obtained are divided into two parts. We first developed a method for solving a bi-level quadratic/quadratic fractional problem (Maachou N & M. Moulaï M (2015)). The proposed method is based on Karush-Kuhn-Tucker conditions, for which the second level problem is replaced by its (KKT) optimality conditions. The result is a single level nonlinear mathematical program (*NLP*), where the solution of this nonlinear problem is essentially based on results found in one of our works cited in the thesis (Maachou N & Moulaï M (2008)), the work consists in solving a discrete quadratic fractional problem, the method is based on the solution of a linear fractional problem, optimality conditions are determined in order to solve the problem (*NLP*).

The second part of our contribution consists in developing a new exact method for solving the bi-level indefinite quadratic problem in integers, called **IIQBP**. Indeed, we solve the first level indefinite quadratic problem, whose optimal solution belongs to the set of efficient solutions of the associated bicriteria problem. The set of efficient solutions is determined by the Branch & Cut method. The method is based on the Branch & Bound procedure to find an integer solution, an efficient cut is used to eliminate the dominated solutions from the feasible domain. The obtained optimal integer solution of the first level is then tested for optimality of the main problem by solving the second level problem, the algorithm used for solving the first level is retained to solve the second level. If this solution is not optimal for the main problem, it is eliminated from the domain, adding the Danzig cut to the first level problem and continuing the exploration. The proposed algorithm is an exact method, which provides the optimal solution if it exists. The entire optimal solution of the problem **IIQBP** is obtained in a finite number of iterations. In order to measure the efficiency of our algorithm, we constructed a table of values to determine the computation time for two examples and for other examples of larger dimensions. The execution time was reasonable. A computer program was developed under the Matlab environment.

À mes chers parents

À Rayan, mon adorable fils

À mon cher époux pour son soutien quotidien indéfectible et
son enthousiasme contagieux à l'égard de mes travaux comme

de la vie en général

À mon cher frère et mes chères sœurs

Remerciement

Je voudrais tout d'abord exprimer mes plus profonds remerciements au Professeur Mustapha Moulaï qui a assuré l'encadrement scientifique et la direction de cette thèse pour la confiance qu'il m'a accordée et pour les nombreux conseils qu'il m'a apportée dans les moments difficiles pour arriver au bout de ce grand défi. Je dois aussi mentionner sa patience et sa constance tout au long de ces années.

Mon profond respect va à Monsieur le Professeur Moncef ABBAS qui m'a honorée en acceptant de présider mon jury.

Je suis très sensible à l'honneur que m'ont fait Messieurs Mohamed Saïd RADJEF, Professeur à l'Université de Bejaïa , Mouaouia Cherif BOUZID Maître de conférences à ENST/Alger et Nedjmeddine KANTOUR Docteur à l'USTHB, en acceptant d'examiner ce travail.

Je voudrais plus particulièrement remercier Mr CHAIBLAINE Yacine, doctorant à l'USTHB, avec qui des discussions scientifiques ont été établies pour mieux mettre en valeur la rédaction de mon travail.

Je remercie tous mes collègues et amis pour leurs aides et soutiens et tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail en particulier Melle Fali Fatima.

Avant Propos

Ce travail de thèse a été réalisé sous la Direction du Professeur Mustapha Moulai qui a mis à ma disposition gracieusement tous les moyens nécessaires pour la préparation de cette thèse. Par ailleurs, ce travail a donné lieu à trois (03) publications internationales ainsi qu'à onze (11) communications internationales et cinq (05) nationales.

Publications Internationales :

1. Nacéra Maachou and Mustapha Moulai, Branch and cut method for solving integer indefinite quadratic bilevel programs, Janvier 2022, Annals of Operations Research, DOI :10.1007/s10479 – 021 – 04387 – 4.
2. Nacéra Maachou and Mustapha Moulai, Bilevel quadratic fractional/quadratic problem, Modelling, Computation and Optimization in Information Systems and Management Sciences Volume 359 of the series Advances in Intelligent Systems and Computing pp 381-389, 2015. http://link.springer.com/chapter/10.1007%2F978-3-319-18161-5_32.
3. Nacéra Maachou and Mustapha Moulai, An Exact Method for a Discrete Quadratic Fractional Maximum Problem. In : Le Thi H.A., Bouvry P., Pham Dinh T. (eds) Modelling, Computation and Optimization in Information Systems and Management Sciences. Communications in Computer and Information Science, vol 14. Springer, Berlin, Heidelberg, 2008. https://doi.org/10.1007/978-3-540-87477-5_22.

Communication Internationales :

1. Nacéra Maachou and Mustapha Moulai, Branch and cut method for solving integer indefinite quadratic bilevel programs, International Symposium on Combinatorial Optimisation CO2016, University of Kent, Canterbury, UK, 1-3 September 2016.
2. Nacéra Maachou and Mustapha Moulai, Extreme Point Bilevel Quadratic Fractional-Quadratic Problem, 27th European Conference on Operational Research (IFORS'2015), Glasgow, UK, 12-15 Juillet 2015.
3. Nacéra Maachou and Mustapha Moulai, Bilevel quadratic fractional/quadratic problem , Modelling, Computation and Optimization in information Systems and Management Sciences MCO'2015 Metz (France) 11-13 Mai 2015.<http://www.lita.univ-lorraine.fr/iccsama2015/MCO/>.

4. Nacéra Maachou and Mustapha Moulai, Integer bilevel quadratic fractional / quadratic programming problem, The 20th Triennial Conference of the International Federation of Operational Research Societies (IFORS), Barcelona, Spain July 13 – 18, 2014 [http ://ifors2014.org/](http://ifors2014.org/).
5. Nacéra Maachou and Mustapha Moulai, Branch and Bound method for a Discrete Multiobjective Quadratic Fractional Problem”. (ICALT’2013) International Conference On Advanced Logistics and Transport, Sousse, Tunisia, 29-31 Mai 2013.
6. Nacéra Maachou and Mustapha Moulai, Solving Quadratic Fractional / Quadratic Bilevel Problem. (EURO XXVI) The 26th European Conference on Operational Research, Roma, 1-4 Juillet 2013 [http ://ifors.org/web/26th-european-conference-on-operational-research-euro-xxvi/](http://ifors.org/web/26th-european-conference-on-operational-research-euro-xxvi/).
7. Nacéra Maachou and Mustapha Moulai, Integer Bilevel Quadratic Fractional / Linear Fractional Programming Problem, Cinquième Conférence Internationale en Recherche Opérationnelle (CIRO’10), Marrakech (Maroc), 24-26 Mai 2010.
8. Nacéra Maachou and Mustapha Moulai, The Bilevel Quadratic Fractional / Quadratic Programming Problem, Multi-objective programming and goal programming (MOPGP’10 à Sousse (Tunisie), (2010).
9. Nacéra Maachou and Mustapha Moulai, Branch and Bound method for Integer Quadratic Fractional Programming Problem, IFORS’08 en Afrique du Sud, 13-18 juillet 2008.
10. Nacéra Maachou and Mustapha Moulai, Quadratic Fractional Bilevel Programming Problem, NCP07, Rouen, France, Decembre, 17-21, 2007.
11. Nacéra Maachou and Mustapha Moulai, Multicriteria Integer Quadratic Fractional Programming Problem, CIRO’02, Marrakech, Maroc June, 4 – 6, 2002.

Communication nationales :

1. Nacéra Maachou and Mustapha Moulai, Solving a Discrete Multiobjective Quadratic Fractional Problem, Journées graphes et optimisation JGO 2014, USTHB, Novembre 27-28 2014.
2. Nacéra Maachou and Mustapha Moulai, Solving Integer Quadratic Fractional Problem, Acceptation pour le Colloque sur l’Optimisation et les Systèmes d’Information (COSI’2010), Ouargla, Algérie , 18-20 Avril 2010.
3. Nacéra Maachou and Mustapha Moulai, A Discrete Bilevel Linear Fractional Programming Problem, COSI’2009, Annaba - Algérie, 25-27 Mai 2009.

4. Nacéra Maachou and Mustapha Moulai, Non linear Bilevel Programming Problem, RAMA VI , TIZI-OUZOU, Algérie, 26-28 avril 2008.
5. Nacéra Maachou and Mustapha Moulai, Penalties method for integer quadratic programs, MOAD'4, Bejaia, Algérie, Novembre, 18-19, 2007.

Table des matières

Introduction générale	12
1 L'optimisation non linéaire	16
1.1 Introduction	16
1.2 Généralité sur les matrices	17
1.3 La convexité en optimisation	18
1.3.1 Fonctions convexes généralisées	20
1.4 Conditions d'optimalité	21
1.4.1 Conditions de Kuhn-Tucker	23
1.5 La programmation fractionnaire linéaire	24
1.5.1 Résultats théoriques	25
1.5.2 Applications des problèmes fractionnaires	25
1.5.3 Méthodes de résolution	26
1.5.4 La programmation quadratique indéfinie	27
1.5.5 Application des problèmes quadratiques indéfinis	27
1.5.6 Méthode de Résolution	28
1.6 Conclusion	29
2 Généralité sur l'optimisation multiobjectif	30
2.1 Introduction	30
2.2 Notions fondamentales	31
2.2.1 Position du problème	31
2.2.2 La relation de dominance	31
2.2.3 Optimalité de Pareto	32
2.2.4 Détection graphique de l'efficacité	34
2.2.5 Caractérisation des solutions efficaces	34
2.3 Choix de la méthode d'aide à la décision	35
2.4 Programmation linéaire multiobjectif en nombre entiers MOILP	36

2.4.1	Formulation du problème	37
2.4.2	Solutions supportées et non supportées	37
2.4.3	Quelques méthodes de résolution d'un problème <i>MOILP</i>	38
2.5	Programmation non linéaire multiobjectif en nombres entiers MOINLP	40
2.5.1	Programmation fractionnaire linéaire multiobjectif en nombres entiers MOIFLP	40
2.6	Conclusion	41
3	Programmation mathématique à deux niveaux	42
3.1	Introduction	42
3.2	Quelques applications de l'optimisation à deux niveaux	43
3.3	Programmation linéaire à deux niveaux	46
3.4	Programmation non linéaire à deux niveaux	49
3.5	Multiplicité des solutions optimales du second niveau	49
3.5.1	L'approche optimiste	51
3.5.2	L'approche pessimiste	51
3.6	Autres formulations du problème à deux niveaux	52
3.7	Conditions d'optimalité-Approche optimiste	53
3.7.1	Cas linéaire	53
3.7.2	Utilisation des conditions de KKT	54
3.8	Quelques méthodes de résolution pour les problèmes de programmation à deux niveaux	55
3.8.1	Méthode de la fonction de pénalité :	55
3.8.2	La méthode descente	55
3.8.3	La méthode de Branch and Bound :	56
3.9	La programmation à deux niveaux en nombres entiers	56
3.10	La complexité de la programmation à deux niveaux	59
3.11	Conclusion	60
4	Programmation fractionnaire quadratique/quadratique à deux niveaux	61
4.1	Partie 1 : Une méthode exacte pour un problème fractionnaire quadratique discret	61
4.1.1	Introduction	61
4.1.2	Notations et définitions	62
4.1.3	Quelques résultats	63
4.1.4	Notations	66
4.1.5	Conclusion	68

4.2	Partie2 : Résolution d'un porogramme fractionnaire quadratique/quadratique à deux niveaux	68
4.2.1	Introduction	68
4.2.2	Notations et définitions	69
4.2.3	Quelques résultats	71
4.2.4	conclusion	75
5	La méthode Branch & Cut pour la résolution d'un problème quadratique indéfini à deux niveaux discret	77
5.1	Introduction	77
5.2	Définitions et notations	78
5.3	Technique pour résoudre un problème quadratique indéfini en nombres entiers	79
5.3.1	Procédure pour générer les solutions efficaces entières d'un problème linéaire bicritère	81
5.3.2	Résultats théoriques	84
5.3.3	Méthodologie et Algorithme pour la résolution de problème quadratique indéfini en nombres entiers IQP	99
5.4	Méthode "Branch & cut" pour la résolution des problèmes quadratiques indéfinis à deux niveaux en nombres entiers IIQBP	100
5.4.1	Description de la méthode	100
5.5	Un exemple illustratif :	104
5.5.1	Exemple1	104
5.5.2	Exemple2	115
5.6	Résultats de calcul	133
5.7	Conclusion	134
	Conclusion Générale	135
	Bibliographie	137

Introduction générale

La convergence de divers facteurs tels que la concurrence accrue dans une économie mondiale, l'accès en temps réel à une foule d'informations transparentes et l'apparition d'une génération de consommateurs avertis et pragmatiques sont entrainés de modifier le rôle de la nature de l'établissement de prise de décisions. Bien que dans le passé les modèles d'aides à la décision aient principalement fait l'objet de recherches universitaires pures, ces modèles constituent aujourd'hui des outils financiers et opérationnels fondamentaux au sein d'importants secteurs industriels et se rangent maintenant parmi les principaux facteurs de rentabilité. C'est dans cette optique qu'il est observé aussi bien au sein du secteur des affaires que dans la communauté de chercheurs en mathématiques appliquées une volonté croissante de développer des approches de prise de décision optimales.

Les problèmes de décision se modélisent en général par des problèmes d'optimisation sous contraintes. L'optimisation est l'un des domaines des mathématiques appliqués en pleine expansion qui s'est imposée dès ses débuts dans les années 1940 comme un outil essentiel et incontournable dans la prise de décision. Le courant principal de recherche dans ce domaine s'est toutefois beaucoup plus préoccupé des modèles à un seul critère. C'est à dire des modèles mathématiques bien définis, régis par le paradigme classique d'existence de solutions optimales, de caractérisations et des algorithmes convergeant vers celles-ci.

Même si l'utilisation de ces modèles unicritères s'est révélée particulièrement efficace dans de nombreuses situations, il est toutefois apparu qu'il existe une multitude d'applications pour lesquelles ne prendre en compte qu'un seul décideur ou un seul critère de décision ne répond en aucun cas à la réalité. C'est vers le milieu des années 1980 que la nécessité d'une aide multicritère se fit cruellement sentir et se fut le mérite exceptionnel des scientifiques. Depuis lors, l'aide multicritère à la décision connaît une expansion fulgurante avec le développement de deux principaux courants de recherches : l'optimisation multiobjectifs et l'optimisation à deux niveaux.

Un problème d'optimisation à deux niveaux est caractérisé par deux niveaux de prise

de décision classés de manière hiérarchique et où chaque décideur essaie d'optimiser sa propre fonction objectif, sans tenir compte de l'objectif de l'autre partie, mais la décision de chaque partie affectant la valeur optimale de l'autre partie ainsi que l'espace de décision. Ces problèmes apparaissent dans de nombreuses situations réelles dont entre autres le contrôle optimal, l'optimisation des processus, les problèmes de transport, l'agriculture, la finance, la production de biocarburants. Ainsi, de nombreux chercheurs ont effectué des recherches approfondies dans ce domaine. Candler et Townsley [42] ainsi que Fortuni-Amat et McCarl [62] ont présenté la formulation formelle des problèmes de programmation à deux niveaux BLPP. Arora et Gupta [10] ont présenté une approche de programmation par objectifs floue et interactive pour les problèmes de programmation linéaires à deux niveaux avec les caractéristiques de la programmation dynamique.

Les problèmes de la programmation non linéaire à deux niveaux ont été étudiés par plusieurs chercheurs, notamment, le problème de la programmation quadratique à deux niveaux QBLPP a été étudié par Júdice et Faustino [70]; Vicente et al. [136]; Wang S and Wang Q [137]; Calvete and Galé[33].

Vicente et al. [136] ont introduit deux méthodes pour les problèmes QBLPP dans lesquelles la fonction du second niveau est quadratique strictement convexe, la fonction du premier niveau est quadratique, et ils ont prouvé que "la vérification de l'optimalité locale pour la programmation à deux niveaux est un problème NP-hard". Wang S et Wang Q [137] ont présenté les conditions d'optimalité et l'algorithme de résolution du problème de programmation linéaire quadratique. Calvete et Galé [33] ont étudié les conditions d'optimalité pour le problème de programmation linéaire fractionnaire/ quadratique à deux niveaux en se basant sur les conditions de Karush-Kuhn-Tucker et la théorie de la dualité. Maachou et Moulai [88] ont développé un algorithme basé sur les conditions de Karush-Kuhn-Tucker et un problème linéaire fractionnaire/quadratique à deux niveaux pour résoudre un problème fractionnaire quadratique/ quadratique à deux niveaux.

Les problèmes de la programmation à deux niveaux en nombres entiers ont été abordés par Narang et Arora [104] qui ont présenté un algorithme pour résoudre un problème QBLPP entier indéfini avec des variables bornées. Ils l'ont traité en résolvant le problème relaxé. Tandis que, Thirwani et Arora [135] ont développé un algorithme pour résoudre les problèmes QBLPP avec les variables entières. Ils ont résolu le problème en utilisant une technique de linéarisation et ont obtenu une solution entière du problème QBLPP en utilisant la coupe de Gomory et la méthode de dual de simplexe.

Une recherche bibliographique nous a permis de conclure que la résolution du problème non linéaire à deux niveaux n'avait pas reçu autant d'attention jusqu'à ce jour, et beaucoup plus lorsque les deux niveaux sont quadratiques indéfinis dans un domaine réalisable

discret. D'où notre motivation à explorer cette classe de problèmes.

La première partie de notre contribution consiste à élaborer un algorithme pour résoudre un problème fractionnaire quadratique/quadratique à deux niveaux, (Maachou et Moulaï[88]). La méthode proposée est basée sur les conditions de Karush-Kuhn-Tucker, pour lequel le problème du second niveau est remplacé par ses conditions d'optimalité de Karush-Kuhn-Tucker (KKT). Il en résulte un programme mathématique non linéaire à un seul niveau (*PNL*), où la résolution de ce problème non linéaire s'appuie essentiellement sur des résultats trouvés dans notre travail Maachou et Moulaï [89], le travail consiste à résoudre un problème fractionnaire quadratique discret, la méthode est basée sur la résolution d'un problème fractionnaire linéaire, des conditions d'optimalités sont déterminées afin de résoudre le problème (*PNL*).

La deuxième partie de notre contribution consiste à mettre au point une nouvelle méthode exacte pour résoudre le problème quadratique indéfini à deux niveaux en nombres entiers, dit *IIQBP*. En effet, on résout le problème quadratique indéfini du premier niveau, dont la solution optimale appartient à l'ensemble des solutions efficaces du problème bicritère associé. L'ensemble des solutions efficaces est déterminé par la méthode Branch & Cut. Puis la solution optimale entière obtenue est testée pour l'optimalité du problème principal en résolvant le problème du second niveau. Si cette solution n'est pas optimale, on l'élimine du domaine, en ajoutant la coupe de Dantzig au problème du premier niveau et on continue l'exploration.

La thèse est organisée de la façon suivante :

- Le chapitre 1 intitulé "L'optimisation non linéaire", expose l'essentiel de l'optimisation non linéaire nécessaire à la compréhension du reste de cette thèse. On présente en particulier l'optimisation quadratique indéfinie.
- Le chapitre 2 intitulé "L'optimisation multiobjectif", dans ce chapitre nous présentons l'essentiel des définitions et résultats liés à l'optimisation multiobjectif et nous parlerons de quelques travaux récents existant en littérature traitant les problèmes de l'optimisation multiobjectif linéaire et non-linéaire.
- Le chapitre 3 intitulé "Programmation mathématique à deux niveaux" expose le cadre général de notre travail, nous présentons pour cela l'essentiel des définitions et résultats liés à la programmation à deux niveaux et nous parlerons de quelques travaux récents existant en littérature traitant les problèmes de la programmation à deux niveaux.
- Le chapitre 4 intitulé " Programmation fractionnaire quadratique/quadratique à

deux niveaux” met en évidence notre contribution dans la programmation non linéaire à deux niveaux, il a fait également l’objet d’une publication (Maachou et Moulaï [88]), nous avons élaboré une méthode qui résout le problème fractionnaire quadratique/quadratique à deux niveaux où la méthode de résolution s’appuie sur les résultats de notre travail déjà cité (Maachou et Moulaï [89]). Afin, de détailler nos travaux, nous avons subdivisé ce chapitre en deux parties : La première concerne la résolution d’un problème fractionnaire quadratique discret, alors que dans la seconde partie, nous exposons la méthode de résolution d’un problème fractionnaire quadratique/quadratique à deux niveaux.

- Le chapitre 5 intitulé ”La méthode Branch & Cut pour la résolution d’un problème quadratique indéfini à deux niveaux discret” traite une autre classe des problèmes non linéaire à deux niveaux, qui a fait l’objet d’une publication en janvier 2022 dans la revue Annals of Operation Research (ANOR). Les deux niveaux du problème traité sont quadratiques indéfinis en nombres entiers. La méthode de résolution a été programmée sous environnement Matlab, et deux exemples ont été traités afin d’évaluer la performance de la méthode.
- Enfin, la thèse s’achèvera par une conclusion générale et quelques perspectives de recherche.

Chapitre 1

L'optimisation non linéaire

1.1 Introduction

Les problèmes de la programmation linéaire se posent lorsque l'on cherche à optimiser (maximiser ou minimiser) une fonction linéaire de plusieurs variables, étant donnée que celles-ci doivent vérifier un certain nombre d'équations et/ou d'inéquations appelées *contraintes*. La forme exacte de ces contraintes peut différer d'un problème à un autre, cependant, tout programme linéaire peut être transformé sous la forme standard suivante :

$$(LP) \left\{ \begin{array}{l} \text{optimiser } Z(x) = c^t x \\ s.c \\ Ax = b \\ x \geq 0 \end{array} \right.$$

où c et x sont deux n -vecteurs réels, A est une $m \times n$ -matrice réelle et b un m -vecteur réel.

L'ensemble des contraintes $x \in S = \{x \in \mathcal{R}^n | Ax = b, x \geq 0\}$ représente l'ensemble des solutions réalisables de (LP) .

Si les variables sont astreintes à ne prendre que des valeurs entières, on parle de programme linéaire en nombres entiers (ILP), et si les variables ne peuvent prendre que les valeurs 0 ou 1, le programme est linéaire à variables binaires. Il existe aussi des problèmes dans lesquels une partie seulement des variables ne peut prendre que des valeurs entières ; on parle alors, de programme linéaire mixte.

Les éléments de base de la programmation linéaire ont fait l'objet d'une littérature abondante aussi riche que diversifiée. On retiendra essentiellement les ouvrages spécialisés suivants : [86, 106, 112, 123, 133, 128].

Il existe des algorithmes polynômiaux efficaces pour résoudre un programme linéaire comme ceux dits de points intérieurs initiés par Karmarkar [81] en 1984. Néanmoins l'algorithme du simplexe introduit en 1947 par G.B.Dantzig [47] est le plus célèbre (et le plus efficace dans le cas général) des algorithmes de résolution, bien qu'il ne soit pas polynomial !

L'algorithme du simplexe repose sur le fait qu'une solution optimale d'un programme linéaire peut être prise parmi les sommets du polyèdre de $\mathcal{R}^n$ déterminé par $Ax \leq b$.

Un programme non linéaire est beaucoup plus difficile à résoudre, et la théorie du simplexe ne peut être utilisée. Une solution optimale d'un programme linéaire lorsqu'elle existe est un point extrême de l'ensemble des solutions réalisables (polyèdre des solutions réalisables). Par contre une solution optimale d'un problème non linéaire peut être un point intérieur ou sur la frontière de l'ensemble des solutions réalisables mais pas nécessairement un point extrême. La principale difficulté de la programmation non linéaire est que l'on peut avoir des optimaux locaux qui ne sont pas globaux.

Les cas les plus fréquents de la programmation non linéaire sont la programmation fractionnaire et la programmation quadratique dont nous allons détailler dans ce qui suit, les différents outils pour résoudre ces types de programmes.

1.2 Généralité sur les matrices

Définition 1.2.1 Soit A une matrice carrée d'ordre n , symétrique.

- A est dite définie positive, si $x^t Ax > 0, \forall x \in \mathcal{R}^n, x \neq 0$.
- A est dite définie négative, si $x^t Ax < 0, \forall x \in \mathcal{R}^n, x \neq 0$.
- A est dite semi-définie positive, si $x^t Ax \geq 0, \forall x \in \mathcal{R}^n$ et $\exists x \in \mathcal{R}^n$ tel que $x^t Ax = 0, x \neq 0$.
- A est dite semi-définie négative, si $x^t Ax \leq 0, \forall x \in \mathcal{R}^n$ et $\exists x \in \mathcal{R}^n$ tel que $x^t Ax = 0, x \neq 0$.
- A est dite indéfinie, si $x^t Ax$ peut prendre un signe ou l'autre selon la valeur de x .

Notons que $x^t Ax = \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j$ est appelé forme quadratique.

Définition 1.2.2 On appelle mineur principal d'ordre j de A , noté $\nabla_j(A)$, le déterminant de la sous-matrice de A obtenue en supprimant les $(n-j)$ dernières lignes et les $(n-j)$ dernière colonnes de A .

Les mineurs principaux de A sont : $\nabla_1(A) = a_{11}, \nabla_2(A) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}, \dots, \nabla_n(A) = \det(A)$

Définition 1.2.3 Soit $\lambda \in \mathcal{R}$, λ est appelée valeur propre de A si $\det(A - \lambda I) = 0$.

Propriété 1.2.1 Soit A une matrice carrée d'ordre n , symétrique.

- A est définie positive si et seulement si $(-A)$ est définie négative.
- A est définie positive si et seulement si tous les mineurs principaux de A sont strictement positifs. ie, $\nabla_j(A) > 0, \forall j = 1, n$.
- A est définie positive si et seulement si toutes les valeurs propres de A sont strictement positives.
- A est définie négative si et seulement si $\forall j = 1, n, (-1)^j \nabla_j(A) > 0$.
- A est définie négative si et seulement si toutes les valeurs propres de A sont strictement négatives.
- A est semi-définie positive si et seulement si toutes les valeurs propres de A sont positives ou nulles.
- A est semi-définie négative si et seulement si toutes les valeurs propres de A sont négatives ou nulles.

1.3 La convexité en optimisation

Définition 1.3.1 (Ensemble convexe) On dit que l'ensemble S est convexe si et seulement si :

$$\forall x, y \in S \text{ et } \forall \lambda \in [0, 1], \lambda x + (1 - \lambda)y \in S$$

D'un point de vue géométrique, un convexe est donc un ensemble qui, lorsqu'il contient deux points, contient nécessairement le segment les reliant.

Définition 1.3.2 (Combinaison linéaire convexe) Un vecteur $y \in S$ est une combinaison linéaire convexe des points $\{x_1, \dots, x_p\}$ s'il existent des coefficients réels $\lambda_1, \dots, \lambda_p$, tels que :

$$y = \sum_{i=1}^p \lambda_i x_i, \text{ avec } \sum_{i=1}^p \lambda_i = 1 \text{ et } \lambda_i \geq 0$$

Définition 1.3.3 (Enveloppe convexe) L'enveloppe convexe d'un ensemble $S \subset \mathcal{R}^n$; est l'ensemble des points de $\mathcal{R}^n$ qui s'écrivent comme combinaisons convexes des points de S . Elle est notée par :

$$\text{conv}(S) = \{x \in \mathcal{R}^n / x = \sum_{i=1}^p \lambda_i x_i, x_i \in S, \text{ avec } \sum_{i=1}^p \lambda_i = 1 \text{ et } \lambda_i \geq 0\}$$

C'est le plus petit ensemble convexe contenant S .

- L'ensemble $\{x \in \mathcal{R}^n / a'x = b\}$ représente un hyperplan de $\mathcal{R}^n$;
- L'ensemble $\{x \in \mathcal{R}^n / a'x \leq b\}$ représente un demi-espace fermé de $\mathcal{R}^n$ dont l'hyperplan correspondant constitue la frontière.

Définition 1.3.4 (Polyèdre) – *Un polyèdre S est l'intersection d'un nombre fini de demi-espaces fermés et/ou d'hyperplans.*

- *Un polyèdre est un ensemble convexe fermé.*
- *Un polyèdre S est borné s'il existe une valeur β finie et positive telle que :*

$$\forall x \in S, \|x\| \leq \beta$$

- *Un polytope est un polyèdre borné et non vide.*

Définition 1.3.5 (Point extrême) *Soit S un convexe non vide de $\mathcal{R}^n$. x est dit point extrême ou sommet de S si :*

$$x = \lambda x_1 + (1 - \lambda)x_2 \quad \forall x_1, x_2 \in S \text{ et } \lambda \in]0, 1[\text{ alors } x = x_1 = x_2.$$

Définition 1.3.6 (Fonction convexe) *La fonction $f : S \rightarrow \mathcal{R}$ est dite convexe sur l'ensemble S non vide de $\mathcal{R}^n$ si et seulement si :*

$$\forall x, y \in S \text{ et } \forall \lambda \in [0, 1] \text{ on a } f(x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

Définition 1.3.7 (Fonction concave) *La fonction $f : S \rightarrow \mathcal{R}$ est dite concave sur l'ensemble S non vide de $\mathcal{R}^n$ si et seulement si :*

$$\forall x, y \in S \text{ et } \forall \lambda \in [0, 1] \text{ on a } f(\lambda x + (1 - \lambda)y) \geq \lambda f(x) + (1 - \lambda)f(y)$$

- Les cas stricts correspondent aux mêmes définitions avec des inégalités strictes pour $0 < \lambda < 1$ et $x \neq y$.
- La fonction f est convexe (resp. strictement concave) $\Leftrightarrow (-f)$ est concave (resp. strictement convexe).

Définition 1.3.8 (Épigraphe) *On appelle épigraphe de f noté $\text{épi}(f)$ le sous ensemble de $\mathcal{R}^{n+1}$ défini par :*

$$\text{épi}(f) = \{(x, y) \in S \times \mathcal{R} / y \geq f(x)\}$$

Définition 1.3.9 (Hypographe) *On appelle hypographe de f noté $\text{hyp}(f)$ le sous ensemble de $\mathcal{R}^{n+1}$ défini par :*

$$\text{hyp}(f) = \{(x, y) \in S \times \mathcal{R} / y \leq f(x)\}$$

Théorème 1.3.1 1. f est convexe si et seulement si $\text{épi}(f)$ est convexe.

2. f est concave si et seulement si $\text{hyp}(f)$ est convexe.

Lemme 1.3.1 Si f est une fonction convexe alors l'ensemble niveau $L(f, \alpha) = \{x \in S : f(x) \leq \alpha\}$ est convexe pour tout $\alpha \in \mathcal{R}$.

Définition 1.3.10 Soit $S \subseteq \mathcal{R}^n$, $S \neq \emptyset$. L'ensemble $V_\epsilon(\bar{x}) = \{x \in S : \|x - \bar{x}\| < \epsilon\}$ est appelé voisinage de $\bar{x} \in S$, où $\epsilon > 0$ et $\|\cdot\|$ désigne la norme euclidienne de $\mathcal{R}^n$.

Définition 1.3.11 On dit que $\bar{x}$ appartient à l'intérieur de S , noté $\text{int}(S)$, s'il existe $\epsilon > 0$ tel que $V_\epsilon(\bar{x}) \subset S$.

Théorème 1.3.2 Soit $f : S \subset \mathcal{R}^n \rightarrow \mathcal{R}$ une fonction convexe sur S ensemble convexe non vide. Alors f est continue sur l'intérieur de S .

1.3.1 Fonctions convexes généralisées

Les fonctions convexes généralisées jouent un rôle très important dans la théorie de l'optimisation. Soit une fonction $f : S \subset \mathcal{R}^n \rightarrow \mathcal{R}$ avec S un convexe non vide de $\mathcal{R}^n$.

Définition 1.3.12 (Fonction quasiconvexe) La fonction f est dite quasiconvexe sur S si et seulement si :

$$\forall x, y \in S \text{ et } \forall \lambda \in [0, 1] \text{ on a } f(\lambda x + (1 - \lambda)y) \leq \max\{f(x), f(y)\}$$

f est quasiconvexe sur S est ainsi équivalente à :

$$\forall x, y \in S \text{ et } \forall \lambda \in [0, 1], f(x) \leq f(y) \Rightarrow f(\lambda x + (1 - \lambda)y) \leq f(y)$$

Définition 1.3.13 (Fonction quasiconcave) La fonction f est dite quasiconcave sur S si et seulement si :

$$\forall x, y \in S \text{ et } \forall \lambda \in [0, 1] \text{ on a } f(\lambda x + (1 - \lambda)y) \geq \min\{f(x), f(y)\}$$

La fonction f est quasiconcave $\Leftrightarrow (-f)$ est quasiconvexe.

Une fonction qui est à la fois quasiconvexe et quasiconcave est dite *quasimonotone* ou *quasilinéaire*.

Les fonctions quasiconvexes, quasiconcaves et quasilinéaires sont caractérisées par la convexité de leurs ensembles niveaux.

Remarque 1.3.1 Une fonction convexe sur S convexe est quasiconvexe.

Définition 1.3.14 (Fonction strictement quasiconvexe) La fonction f est strictement quasiconvexe sur S si :

$$\forall x, y \in S \text{ avec } f(x) \neq f(y) \forall \lambda \in]0, 1[\text{ on a } f(\lambda x + (1 - \lambda)y) < \max\{f(x), f(y)\}$$

f strictement quasiconcave si $(-f)$ est strictement quasiconvexe.

Définition 1.3.15 (Fonction différentiable) Soit S un ensemble non vide de $\mathcal{R}^n$ et $f : S \rightarrow \mathcal{R}$. f est dite différentiable en $\bar{x} \in \text{int}(S)$ s'il existe un vecteur gradient noté $\nabla f(\bar{x})$ et une fonction $\alpha : \mathcal{R}^n \rightarrow \mathcal{R}$ tels que :

$$f(x) = f(\bar{x}) + \nabla f(\bar{x})(x - \bar{x}) + \|x - \bar{x}\| \alpha(\bar{x}; x - \bar{x}); \forall x \in S$$

$$\text{où } \lim_{x \rightarrow \bar{x}} \alpha(\bar{x}; x - \bar{x}) = 0 \text{ et } \nabla f(\bar{x}) = \left(\frac{\delta f(\bar{x})}{\delta x_1}, \dots, \frac{\delta f(\bar{x})}{\delta x_n} \right)^t$$

Définition 1.3.16 (Fonction pseudoconvexe) Soit f une fonction différentiable sur S un ouvert non vide de $\mathcal{R}^n$. f est pseudoconvexe sur S si :

$$\forall x, y \in S, (y - x)' \nabla f(x) \geq 0 \Rightarrow f(y) \geq f(x)$$

Définition 1.3.17 (Fonction pseudoconcave) Soit f une fonction différentiable sur S un ouvert non vide de $\mathcal{R}^n$. f est pseudoconcave sur S si :

$$\forall x, y \in S, (y - x)' \nabla f(x) \leq 0 \Rightarrow f(y) \leq f(x)$$

Théorème 1.3.3 Soit $f : S \rightarrow \mathcal{R}$ différentiable et pseudoconvexe sur S un convexe ouvert non vide de $\mathcal{R}^n$. Alors, f est à la fois quasiconvexe et strictement quasiconvexe.

Proposition 1.3.1 1. Si f est convexe alors f est quasiconvexe. Si de plus f est différentiable alors elle est pseudoconvexe.

2. Si f est concave alors f est quasiconcave. Si de plus f est différentiable alors elle est pseudoconcave.

1.4 Conditions d'optimalité

Considérons le problème de minimisation (P_m) suivant :

$$(P_m) \begin{cases} \min f(x) \\ \text{sc} \\ x \in S = \{x \in \mathcal{R}^n | Ax \leq b, x \geq 0\} \end{cases}$$

Définition 1.4.1 (Minimum global) Une solution admissible $\bar{x}$ est dite minimum global si et seulement si $f(\bar{x}) \leq f(x)$, $\forall x \in S$.

Définition 1.4.2 (Minimum local) On dit que la fonction $f(x)$ possède un minimum local en $\bar{x}$ s'il existe un ϵ , $\epsilon \geq 0$, tel que pour tout x dans $V_\epsilon(\bar{x})$, $f(\bar{x}) \leq f(x)$.

Tout minimum global est évidemment un minimum local.

Théorème 1.4.1 (Condition nécessaire d'optimalité) Soit $f : S \subset \mathcal{R}^n \rightarrow \mathcal{R}$ est une fonction convexe sur S un ensemble convexe non vide. Si x est un minimum local (maximum local) alors il est un minimum global (maximum global) pour le problème (P_m) .

Théorème 1.4.2 Si $f : S \subset \mathcal{R}^n \rightarrow \mathcal{R}$ est une fonction convexe et différentiable sur S convexe non vide et $\bar{x} \in S$ tel que $\nabla f(x) = 0$, alors x est un minimum global de f sur S .

Théorème 1.4.3 Soit $f : S \rightarrow \mathcal{R}$ une fonction quasiconcave et continue sur S polyèdre borné, alors au moins une solution optimale $\bar{x}$, est un point extrême de S .

Les fonctions strictement quasiconvexes et strictement quasiconcaves, sont particulièrement importantes dans la programmation non linéaire, car elles assurent qu'un minimum local et un maximum local sur un ensemble convexe soit, respectivement, un minimum global et un maximum global.

Théorème 1.4.4 Soit $f : S \rightarrow \mathcal{R}$ strictement quasiconvexe sur S et S un convexe non vide de $\mathcal{R}^n$. Si x est un minimum local alors x est un minimum global.

Les fonctions pseudoconvexes partagent, la propriété importante du théorème 1.4.2, avec les fonctions convexes.

Théorème 1.4.5 Si $f : S \subset \mathcal{R}^n \rightarrow \mathcal{R}$ est une fonction pseudoconvexe et différentiable sur S convexe non vide et $\bar{x} \in S$ tel que $\nabla f(x) = 0$, alors x est un minimum global de f sur S .

1.4.1 Conditions de Kuhn-Tucker

Considérons le problème de minimisation (P) :

$$(P) \begin{cases} \min f(x) \\ sc \\ h_i(x) = 0, \quad i = 1, \dots, m_1; \\ g_j(x) \leq 0, \quad j = 1, \dots, m_2; \\ x \geq 0. \end{cases}$$

On suppose que f et h_i , $i = 1, \dots, m_1$ et g_j , $j = 1, \dots, m_2$, sont différentiables sur un domaine de $\mathbb{R}^n$.

Définition 1.4.3 – Une contrainte $g_j(x) \leq 0$ est dite active (ou saturée) en x^* si $g_j(x^*) = 0$ et elle est dite inactive si $g_j(x^*) < 0$.
 – Une contrainte $h_i(x) = 0$ est dite active (ou saturée) en x^* si elle est vérifiée en x^* , c-à-d $h_i(x^*) = 0$.
 – L'ensemble des indices des contraintes actives en x^* est noté $I(x^*)$.

Définition 1.4.4 On appelle Lagrangien associé au problème (P) la fonction définie par :

$$L(x, \lambda, \mu) = f(x) + \sum_{i=1}^{m_1} \lambda_i h_i(x) + \sum_{j=1}^{m_2} \mu_j g_j(x); \quad \lambda \in \mathcal{R}^{m_1}, \mu \in \mathcal{R}^{m_2}; \mu \geq 0$$

Les λ_i pour $i = 1, \dots, m_1$ et μ_j pour $j = 1, \dots, m_2$ sont appelés multiplicateurs de Lagrange.

Définition 1.4.5 (Contraintes qualifiées) Les contraintes du problème (P) sont dites qualifiées au point x^* si h_i et g_j sont différentiables au point x^* et si les gradients des contraintes actives en x^* sont linéairement indépendants.

Le point x^* est dit dans ce cas point régulier.

Théorème 1.4.6 (CN d'optimalité de Karush-Kuhn-Tucker) Soit $x^* \in S$ un point réalisable du problème (P) . Supposons f , g_j et h_i sont différentiables en x^* et les contraintes sont qualifiées au point x^* . Si x^* est un minimum local de f sur S alors il existe $\lambda^* \in \mathcal{R}^{m_1}$ et $\mu^* \in \mathcal{R}^{m_2}$ tels que :

$$\begin{cases} \nabla L(x^*, \lambda^*, \mu^*) = 0; \\ h_i(x^*) = 0, \quad i = 1, \dots, m_1; \\ \mu_j^* g_j(x^*) = 0, \quad j = 1, \dots, m_2; \\ g_j(x^*) \leq 0, \quad j = 1, \dots, m_2; \\ \mu_j^* \geq 0, \quad j = 1, \dots, m_2, x^* \geq 0. \end{cases} \quad (KT)$$

Si de plus le problème (P) est convexe, alors les conditions de KKT sont suffisantes pour que x^* soit un minimum global de f .

1.5 La programmation fractionnaire linéaire

Les programmes fractionnaires consistent à optimiser un objectif mis sous la forme d'un rapport de deux fonctions linéaires ou non, soumis à un ensemble de contraintes. Différentes versions de ce modèle, en variables entières, en mixtes, en continu ou en variables bivalentes ont plusieurs applications à savoir l'ingénierie, l'optimisation combinatoire, la programmation stochastique, l'informatique et l'économie (voir travaux de Stancu-Minasian [125, 126, 127] et Schaible [114, 115, 116, 117]).

Dans un de nos travaux (voir chapitre 3), nous nous sommes intéressés à la résolution d'un problème de programmation fractionnaire quadratique qui fait appel à un problème fractionnaire linéaire (*LFP*), d'où l'intérêt de la résolution des problèmes (*LFP*). La forme particulière de ce programme a fait que de nombreux auteurs ont élaboré des méthodes de résolution spécifiques qui se sont avérées plus efficaces que l'application directe de méthodes générales de la programmation non linéaire. Trois grandes stratégies de résolution émergent dans la littérature : la résolution directe, la résolution par paramétrisation et la résolution d'un problème équivalent à objectif simplifié. Dans ce chapitre nous présentons la résolution directe.

Le problème de programmation fractionnaire linéaire, ou encore hyperbolique, peut se formuler comme suit :

$$(LFP) \begin{cases} \max f(x) = \frac{p^T x + \alpha}{q^T x + \beta} \\ sc. \\ x \in S. \end{cases} \quad (1.1)$$

où α et β sont des réels, p et q sont des vecteurs de $\mathcal{R}^n$. $S = \{x \in \mathcal{R}^n | Ax \leq b, x \geq 0\}$, avec A une $(m \times n)$ -matrice réelle et b un vecteur de $\mathcal{R}^m$.

On suppose que

- L'ensemble S est borné non vide.
- $q^T x + \beta > 0 \forall x \in S$.
- $\exists x \in S : p^T x + \alpha > 0$.

Si les variables sont astreintes à ne prendre que des valeurs entières, le programme est noté (*ILFP*) et il est appelé programme fractionnaire linéaire en nombres entiers ou encore programme hyperbolique discret.

Lemme 1.5.1 [40] *La fonction $f(x) = \frac{p^T x + \alpha}{q^T x + \beta}$ est à la fois pseudoconvexe et pseudo-concave sur S .*

1.5.1 Résultats théoriques

D'après ce lemme 1.5.1, les sous-sections 1.3.1 et 1.4, on peut citer quelques conséquences pour le problème de programmation linéaire fractionnaire :

- Puisque f est à la fois pseudoconvexe et pseudoconcave sur S donc, d'après le théorème 1.4.2, elle est aussi quasiconvexe, strictement quasiconvexe, quasiconcave et strictement quasiconcave sur S .
- Puisque f est à la fois quasiconvexe et quasiconcave donc elle est quasilinéaire.
- Puisque la fonction objectif f est strictement quasiconvexe et strictement quasiconcave sur S alors, d'après le théorème 1.4.4, un minimum (maximum) local est aussi un minimum (maximum) global sur S respectivement.
- Puisque f est quasiconvexe et quasiconcave et si f est continue sur S alors, d'après le théorème 1.4.3, au moins une solution optimale du problème (PLF) est atteinte en un point extrême de S .
- Puisque f est pseudoconvexe et pseudoconcave sur S donc, d'après le théorème 1.4.5, si $\nabla f(x) = 0$, alors x est un minimum (maximum) global de f sur S .

1.5.2 Applications des problèmes fractionnaires

Plusieurs applications des programmes mathématiques fractionnaires ont déjà été décrites dans la littérature [53], [116], [117]. Comme domaines d'applications des problèmes fractionnaires soumis à un ensemble de contraintes, nous pouvons citer par exemple :

- Le domaine des finances où il est souvent question d'optimiser un rapport de deux fonctions telles que [dette/capitaux propres], [rendement/employé], [coût effectif/coût standard], [bénéfice/coût], [inventaire/ventes], [risque des actifs/capital].
- Le domaine de l'économie offre un large éventail d'applications. En effet, la mesure de l'efficacité des systèmes étudiés s'exprime sous forme de rapports entre les critères techniques et/ou économiques. Par exemple, [rendement/risque], [rendement/investissement], [coût/temps], [dette/capitaux propres], [inventaire/ventes].
- Le domaine de la santé comme la planification dans un hôpital [coût/patient], [infirmière/patient], [docteur/patient], etc.
- Les processus de décision de Markov peuvent également mener à la maximisation du rapport de la moyenne et l'écart type.
- Dans la théorie de l'information, la capacité d'un canal de transmission peut être définie comme étant le taux maximal de transmission sur toutes les probabilités.
- Programmation stochastique : Dans un programme linéaire, en supposant que les composantes de l'objectif sont non constantes, indépendantes et varient suivant une

loi de probabilité donnée. La programmation stochastique se propose de maximiser la probabilité pour que la valeur de l'objectif soit supérieure à une valeur donnée. Lorsque les composantes des fonctions objectifs considérées comme des variables aléatoires, suivant une loi de Gauss. Bereanu [30] a montré que le programme stochastique est équivalent à un programme déterministe fractionnaire linéaire convexe en variables continues.

1.5.3 Méthodes de résolution

Cette section traite des méthodes qui résolvent le programme fractionnaire sous sa forme originale, c'est-à-dire sans modifier ni l'objectif ni l'ensemble des contraintes. Cette approche est utilisée pour résoudre les programmes hyperboliques tant qu'en variables continues qu'en variables entières (en particulier en 0-1).

Résolution d'un programme hyperbolique continu :

Plusieurs approches furent élaborées pour la recherche d'une solution optimale d'un programme hyperbolique en variables continues ([53, 31, 66, 34]). Parmi les plus récentes, celle de A. Cambini et al. [34]. Les grandes lignes de cette méthode sont présentées ci-dessous :

La méthode de Cambini et Martein [34] : Rappelons le programme hyperbolique continu (LFP) :

$$(LFP) \begin{cases} \max f(x) = \frac{p^T x + \alpha}{q^T x + \beta} \\ sc. \\ x \in S = \{x \in \mathcal{R} | Ax \leq b\}. \end{cases} \quad (1.2)$$

où α et β sont des réels, p et x des vecteurs de $\mathcal{R}^n$, q un vecteur non nul de $\mathcal{R}^n$, A une matrice réelle d'ordre $m \times n$ et b un vecteur de $\mathcal{R}^m$. L'ensemble S est supposé borné non vide et $q^T x + \beta > 0$ sur S .

Définition 1.5.1 *Un point réalisable $\bar{x}$ est une solution niveau optimale pour le problème (LFP) si $\bar{x}$ est une solution optimale pour le programme linéaire suivant :*

$$(LFP_{\bar{x}}) \begin{cases} \max(p^T x + \alpha) \\ sc. \\ x \in S \text{ et } qx = q\bar{x}. \end{cases} \quad (1.3)$$

Si de plus $\bar{x}$ est un point extrême de S , $\bar{x}$ est dit solution niveau optimale de base.

L'algorithme de (Cambini et al.) [34], génère une séquence finie de solutions niveau optimales dont la première x^0 est une solution optimale du programme linéaire :

$$\begin{cases} \min(q^T x + \beta) \\ sc. \\ x \in S. \end{cases} \quad (1.4)$$

- Si x^0 est unique, alors elle est aussi une solution niveau optimale de base pour (LFP).
- Sinon, résoudre le programme linéaire (LFP $_{x^0}$) pour obtenir une solution niveau optimale de base.

Théorème 1.5.1 [94] *Le point x^0 de S est une solution optimale du problème linéaire fractionnaire (LFP) si et seulement si le vecteur gradient réduit $\bar{\gamma} = \bar{\beta}.\bar{p} - \bar{\alpha}.\bar{q}$ est tel que $\bar{\gamma}_j \leq 0$ pour tout indice hors base $j \in N$.*

1.5.4 La programmation quadratique indéfinie

Un problème de programmation quadratique indéfinie est un programme dont la fonction objectif quadratique n'est ni convexe ni concave, c'est à dire que la matrice G n'est plus définie positive ou définie négative. Dans ce cas le problème (IQP) devient plus difficile à résoudre puisque l'optimum local ne sera pas forcément un optimum global.

Nous nous intéresserons dans cette thèse à l'optimisation d'un critère quadratique indéfini qui peut être écrit sous forme d'un produit de deux fonctions linéaires positives.

Ainsi, notre problème, noté (IQP), peut être écrit sous la forme suivante :

$$(IQP) \begin{cases} \max f(x) = (p^t x + \alpha)(q^t x + \beta) \\ sc \\ \mathcal{S} = \{x \in \mathcal{R}^n / Ax \leq b, x \geq 0\} \end{cases}$$

On suppose que les facteurs $(p^t x + \alpha) > 0$ et $(q^t x + \beta) > 0 \forall x \in \mathcal{S}$, $p, q \in \mathcal{R}^n$ et $\alpha, \beta \in \mathcal{R}$. L'ensemble $\mathcal{S}$ est supposé borné non vide.

1.5.5 Application des problèmes quadratiques indéfinis

Une application pour ce type de programme peut être trouvée dans la théorie microéconomique Cottle R.W. [45], Henderson J.M. et Quandt R.E. [75]; étant donné un consommateur rationnel souhaitant maximiser son utilité de consommer deux produits

soumis à une contrainte budgétaire, la fonction d'utilité étant le produit des quantités des deux produits consommés, il est clair que la consommation est limitée par les niveaux de production des produits et par conséquent il est utile d'inclure des contraintes linéaires supplémentaires relatives aux ressources et aux capacités.

1.5.6 Méthode de Résolution

De nombreuses méthodes exactes ont été proposées pour la résolution des problèmes modélisés sous forme de (IQP) , nous pouvons citer en particulier les méthodes proposées par K.swarup[131], S.D. Sharma[119], C.R. Bector et M. Dahl [19], et P. Anand [8].

Dans un autre travail, Swarup [131] a montré que la fonction quadratique indéfinie est explicitement quasiconcave, en rajoutant des conditions sur la fonction, par conséquent, un minimum local se situe en un point extrême de l'ensemble réalisable et a fourni une méthode similaire à la méthode du simplexe pour trouver un minimum local. Bector et Dahl [19] ont également présenté une technique similaire à la méthode du simplexe pour résoudre le problème IQP , tout en gardant les mêmes conditions exigées par Swarup [131] pour que la fonction quadratique indéfinie soit explicitement quasiconcave. Erenguc et Benson [61] ont proposé un algorithme pour trouver le minimum global d'une fonction quadratique indéfinie sur des entiers contenus dans un ensemble convexe compact en transformant le problème en un problème équivalent avec une fonction objectif séparable. Li et al.[85] ont proposé un algorithme basé sur la relaxation lagrangienne alors que Sun et al. [130] ont proposé une relaxation convexe et une décomposition lagrangienne pour résoudre les problèmes de la programmation en nombres entiers quadratiques indéfinis. Sachant que les problèmes quadratiques indéfinis sont des problèmes dont la fonction objectif quadratique n'est ni convexe ni concave (en retirant les conditions exigées par Swarup [131]), donc, la méthode utilisée par Swarup [131], (la technique similaire à la méthode du simplexe), nous nous conduit pas forcément à la solution optimale globale pour le problème (IQP) . De ce fait, nous avons évité d'utiliser la technique similaire à la méthode du simplexe pour résoudre ce problème. Cependant, la méthode retenue dans notre thèse, est basée sur le principe de Branch & Cut pour résoudre un problème quadratique indéfini en nombres entiers, où la solution optimale du problème (IQP) est une solution efficace de problème linéaire bicritère (BLP) associé au problème (IQP) (la méthode sera détaillée dans le chapitre 5 de la thèse).

1.6 Conclusion

Dans ce chapitre, nous nous sommes intéressés aux problèmes de l'optimisation non linéaire à savoir l'optimisation fractionnaire linéaire, et l'optimisation quadratique indéfinie, nous avons donné quelques travaux consacrés aux programmes fractionnaire linéaire et quadratique. Ce petit tour d'horizon nous a incité à étudier et détailler d'avantage ces types de problèmes que nous utiliserons à la suite de cette thèse dans le cadre d'une étude et de résolution d'un problème de l'optimisation non linéaire à deux niveaux en nombres entiers. Deux problèmes ont été développés, le premier est un problème fractionnaire quadratique / quadratique à deux niveaux, dont on a eu recours aux problèmes fractionnaires linéaires pour sa résolution. Le second est un problème quadratique indéfini à deux niveaux en nombres entiers.

Chapitre 2

Généralité sur l'optimisation multiobjectif

Dans ce chapitre, nous présentons l'optimisation mathématique multiobjectif. Nous rappelons les notions fondamentales en décision multiobjectif, les terminologies usuelles, les principaux résultats ainsi que les approches de base de la programmation mathématique multiobjectif.

2.1 Introduction

La principale difficulté que l'on rencontre en optimisation monoobjectif vient du fait que modéliser un problème sous la forme d'une équation unique peut être une tâche difficile. Avoir comme but de se ramener à une seule fonction objectif peut aussi biaiser la modélisation. L'optimisation multiobjectif autorise ces degrés de liberté qui manquaient en optimisation monoobjectif.

La plupart des problèmes d'optimisation réels sont alors décrits à l'aide de plusieurs objectifs ou critères qui sont souvent contradictoires ou conflictuels et qui doivent être optimisés simultanément.

Il n'existe généralement pas de solution qui optimise tous les critères en même temps, le concept de solution optimale devient alors plus difficile à définir, il faut donc trouver un compromis. Dans ce cas, la solution optimale cherchée n'est plus un point unique, mais un ensemble de compromis. Résoudre un problème comprenant plusieurs critères, consiste donc à calculer le meilleur ensemble de solutions compromis : *le front Pareto*.

Les premiers travaux dans ce domaine sont dus à Koopmans [83] en 1951 qui donna une condition nécessaire et suffisante d'efficacité d'une solution suivis par les travaux

de Kuhn et Tucker [84] la même année qui formulèrent un problème de maximisation vectorielle. Depuis, ce domaine a connu un développement fulgurant, traitant aussi bien du domaine de la programmation multiobjectif linéaire [128] et non-linéaire ([96, 97, 60, 39]) que des problèmes multi-objectifs booléen ou en nombres entiers [134].

2.2 Notions fondamentales

2.2.1 Position du problème

On définit un problème multiobjectif comme un problème de décision qui consiste à optimiser (maximiser ou minimiser) simultanément r fonctions réelles notées f_k , $k = 1, \dots, r$, appelées critères, souvent contradictoires, sur un ensemble de solutions S .

Ce problème peut être formulé mathématiquement comme :

$$(MOP) \begin{cases} \text{"opt"}[f_1(x), f_2(x), \dots, f_r(x)] \\ x \in S \end{cases}$$

Le symbole " " signifie qu'il n'est généralement pas possible de trouver dans S une solution qui optimise simultanément les r critères.

Dans la suite de ce chapitre, on considère que toutes les fonctions objectifs sont à "Maximiser".

- Chaque solution est caractérisée par un vecteur $x = (x_1, x_2, \dots, x_n)$ représentant le vecteur de décision avec x_i , $i = 1, \dots, n$, des variables de décision et n le nombre de ces variables,
- $f(x) = (f_1(x), f_2(x), \dots, f_r(x))$ est le vecteur de r critères f_k , $k = 1, \dots, r$ et r le nombre de critères,
- L'ensemble S est un sous-ensemble de R^n décrit implicitement par des contraintes d'égalités et/ou d'inégalités,
- L'ensemble $\mathcal{R}^n$ qui contient S est dit *espace de décision*,
- L'ensemble $\mathcal{R}^r$ qui contient F est dit *espace des critères*,
- L'ensemble $F = f(S)$ qui est la projection de l'espace S sur l'espace des critères.

2.2.2 La relation de dominance

Lors de résolution d'un problème multiobjectif, toute solution admissible ($x \in S$), ne constitue pas évidemment un bon compromis à considérer ; pour qu'elle le soit, on impose généralement une propriété, basée sur une relation d'ordre partiel et appelée dominance.

Définition 2.2.1 (Dominance) Soient deux vecteurs critères $f(x), f(y) \in F$. On dit que $f(x)$ domine $f(y)$ si et seulement si $f(x) \geq f(y)$ et $f(x) \neq f(y)$ i.e.

$$f_i(x) \geq f_i(y) \quad \forall i = \{1, \dots, r\} \quad \text{et} \quad \exists i \in \{1, \dots, r\} \quad \text{tel que} \quad f_i(x) > f_i(y).$$

Si $f(x)$ domine $f(y)$, alors $f(x)$ est au moins aussi bon que $f(y)$ sur tous les critères et meilleur que lui sur au moins un critère.

2.2.3 Optimalité de Pareto

Définition 2.2.2 (Efficacité) Une solution $\bar{x} \in S$ est dite solution efficace (ou Pareto optimale) si et seulement s'il n'existe pas de solution $x \in S$ telle que $f_i(x)$ domine $f_i(\bar{x})$ $\forall i \in \{1, \dots, r\}$.

Un point est efficace si son image par f est un vecteur critère non dominé. Une définition équivalente de l'efficacité est :

Définition 2.2.3 Une solution $\bar{x} \in S$ est dite solution efficace si et seulement s'il n'existe pas de solution $x \in S$ telle que

$$f_i(x) \geq f_i(\bar{x}), \quad \forall i \in \{1, \dots, r\} \quad \text{et} \quad \exists j \in \{1, \dots, r\} \quad \text{avec} \quad f_j(x) > f_j(\bar{x}).$$

A partir d'un point efficace, il est impossible d'augmenter la valeur d'un des critères sans diminuer la valeur d'au moins un autre critère.

Définition 2.2.4 Une solution $\bar{x} \in S$ est dite solution faiblement efficace si et seulement s'il n'existe pas de solution $x \in S$ telle que :

$$f_i(x) > f_i(\bar{x}), \quad \forall i \in \{1, \dots, r\}.$$

Il est clair qu'une solution efficace est faiblement efficace, mais l'inverse est faux.

Définition 2.2.5 (Ensemble Pareto optimal) L'ensemble Pareto optimal de S (ou l'ensemble efficients), est défini par l'ensemble $\mathcal{X}$:

$$\mathcal{X} = \{x \in S \mid \nexists x' \in S, f(x') \text{ domine } f(x)\}.$$

Définition 2.2.6 (Frontière Pareto de F) Soit F l'image dans l'espace des critères de l'ensemble réalisable S . La frontière Pareto SND de F est définie par :

$$SND = \{y \in F \mid \nexists y' \in F, y' \geq y\}.$$

Elle est aussi définie comme l'image de l'ensemble Pareto optimal dans l'espace F .

Un exemple de surface de Pareto en dimension 2 est montré à la figure 2.1.

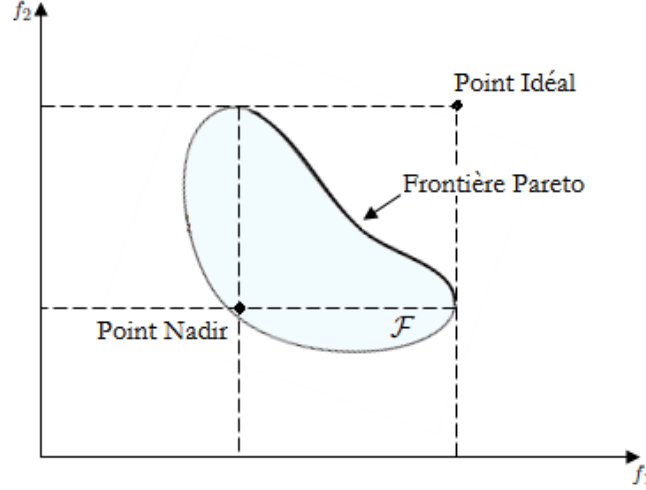


FIGURE 2.1 – La Frontière Pareto

Dans cet exemple, le problème considéré est un problème de maximisation avec deux critères.

Deux points particuliers apparaissent clairement : le *point idéal* et le *point Nadir*, ces deux points sont calculés à partir de la frontière Pareto. Le point idéal (resp. le point Nadir) domine (resp. est dominé par) tous les autres points de la surface de Pareto. Bien que ces points ne soient pas forcément compris dans la zone réalisable, ils servent comme *point de référence* permettant de discuter de l'interêt des solutions trouvées. Les coordonnées de ces points sont définies comme suit :

Définition 2.2.7 (Point Idéal) Les coordonnées du point idéal (Z^I) correspondent aux meilleures valeurs de chaque objectif des points du front Pareto. Les coordonnées de ce point correspondent aussi aux valeurs obtenues en optimisant chaque fonction objectif séparément :

$$Z_i^I = \max\{y_i \mid y \in \text{SND}(F)\}.$$

Définition 2.2.8 (Point Nadir) Les coordonnées du point Nadir (Z^{nad}) correspondent aux pires valeurs de chaque objectif des points du front Pareto.

$$Z_i^{nad} = \min\{y_i \mid y \in \text{SND}(F)\}.$$

La figure 2.1 nous indique aussi que la frontière Pareto peut avoir des propriétés particulières quant à sa forme. La principale caractéristique utilisée pour comparer les

formes de ces courbes est la convexité.

La convexité est le premier indicateur de la difficulté du problème. En effet, certaines méthodes sont dans l'incapacité de résoudre des problèmes non convexes de manière optimale. Mais il existe d'autres indicateurs tout aussi importants, notamment la continuité, la nature des variables de décision (entières ou réelles), etc.

2.2.4 Détection graphique de l'efficacité

Une version "graphique" de l'optimalité au sens de Pareto utilise le théorème du contact.

Définition 2.2.9 Soit $V \subset \mathcal{R}^r$, $V \neq \emptyset$. V est un cône si $\alpha.v \in V$ pour tout scalaire $\alpha \geq 0$ et tout $v \in V$.

Définition 2.2.10 (Cône non négatif) Un cône non négatif est défini dans $\mathcal{R}^r$ par $C^+ = \{x | f(x) \in \mathcal{R}^r \text{ et } f(x) \geq 0\}$.

Définition 2.2.11 (Théorème du contact) Un vecteur x est optimal au sens de Pareto pour un problème d'optimisation multiobjectif donné si $(C^+ + x) \cap F = \{x\}$.

L'utilisation de ce théorème est illustré par la figure 2.2

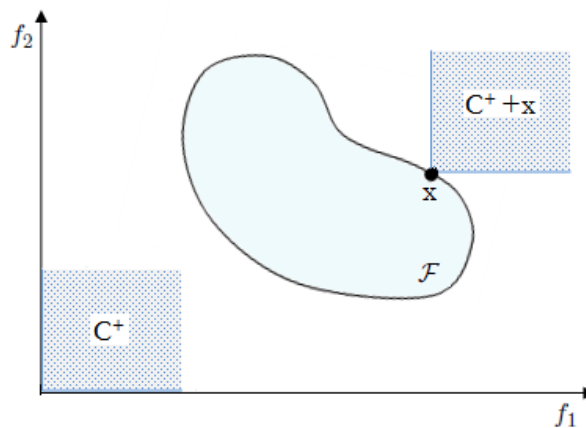


FIGURE 2.2 – Le théorème du contact

2.2.5 Caractérisation des solutions efficaces

Nous présentons quelques caractérisations qui permettent de tester l'efficacité d'une solution réalisable d'un problème d'optimisation multiobjectif.

Théorème 2.2.1 (Wendell[138]) Soit $x^* \in S$ une solution optimale du problème auxiliaire $(P(\hat{x}))$ suivant :

$$P(\hat{x}) \begin{cases} \max \sum_{i=1}^r f_i(x) \\ \text{s.c. } f_i(x) \geq f_i(\hat{x}), \forall i \in \{1, \dots, r\}, \quad x \in S. \end{cases} \quad (2.1)$$

où $\hat{x}$ un vecteur quelconque de S . Alors x^* est Pareto optimal pour le problème multiobjectif (MOP) avec $f_i(x^*) \geq f_i(\hat{x})$.

Théorème 2.2.2 (Benson [24]) Soit x^* une solution réalisable arbitraire donnée et soit le problème unicritère suivant :

$$\begin{cases} \max \Theta \sum_{i=1}^r \varepsilon_i \\ \text{s.c. } x \in S \text{ avec } f_i(x) - \varepsilon_i = f_i(x^*), \text{ avec } \varepsilon_i \geq 0 \forall i \in \{1, \dots, r\}. \end{cases} \quad (2.2)$$

Le vecteur x^* est Pareto optimal pour (MOP) si et seulement si la valeur optimale de la fonction objectif Θ est nulle dans le problème 2.2.

Si pour un point $\hat{x}$, la valeur de Θ est différente de zéro alors $\hat{x}$ est Pareto optimal.

Définition 2.2.12 Soit $S \in \mathcal{R}^n \rightarrow \mathcal{R}$ et $\bar{x} \in S$. Alors :

$L_{\geq} f(\bar{x}) = \{x \in S : f(x) \geq f(\bar{x})\}$ est appelé ensemble niveau (level set) de $\bar{x}$ pour f .

$L_{=} f(\bar{x}) = \{x \in S : f(x) = f(\bar{x})\}$ est appelé courbe niveau (level curve) de $\bar{x}$ pour f .

Théorème 2.2.3 (Ehrgott [59]) Soit $\bar{x} \in S$ est Pareto optimale pour le problème multiobjectif MOP si et seulement si :

$$\bigcap_{i=1}^r L_{\geq} f_i(\bar{x}) = \bigcap_{i=1}^r L_{=} f_i(\bar{x}) \quad (2.3)$$

2.3 Choix de la méthode d'aide à la décision

Dans la résolution d'un problème multiobjectif menant à la détermination d'un ensemble de solutions Pareto, il est nécessaire de faire intervenir l'humain à travers un décideur, pour le choix final de la solution à garder. Ainsi, avant de se lancer dans la résolution d'un problème multiobjectif, il faut se poser la question du type de méthode d'optimisation à utiliser. Elles peuvent être classées suivant trois catégories qui diffèrent selon les préférences du décideur pour la construction de sa fonction d'utilité.

Nous pouvons trouver les familles suivantes [54] :

- *Les méthodes d'optimisation à priori* : dans ce cas, le compromis que l'on désire faire entre les critères a été défini avant l'exécution de la méthode. Ainsi une seule exécution permettra d'obtenir la solution recherchée. Cette approche est donc rapide, mais il faut cependant prendre en compte le temps de modélisation du compromis et la possibilité pour le décideur de ne pas être satisfait de la solution trouvée et de relancer la recherche avec un autre compromis.
- *Les méthodes d'optimisation progressives* : ici, le décideur intervient dans le processus de recherche de solutions en répondant à différentes questions afin d'orienter la recherche. Cette approche permet donc de bien prendre en compte les préférences du décideur, mais nécessite sa présence tout au long du processus de recherche.
- *Les méthodes d'optimisation à posteriori* : Dans cette troisième famille de méthodes, on cherche à fournir au décideur un ensemble de bonnes solutions bien réparties. Il peut ensuite, au regard de l'ensemble des solutions, sélectionner celle qui lui semble la plus appropriée. Ainsi, il n'est plus nécessaire de modéliser les préférences du décideur (ce qui peut s'avérer être très difficile), mais il faut en contre-partie fournir un ensemble de solutions bien réparties, ce qui peut également être difficile et requérir un temps de calcul important (mais ne nécessite pas la présence du décideur).

2.4 Programmation linéaire multiobjectif en nombre entiers MOILP

La programmation linéaire est l'une des plus importantes techniques d'optimisation utilisées en recherche opérationnelle. Ses développements théoriques ont été suggérés et accélérés par un grand nombre d'applications pratiques, dans le domaine de l'économie, de la gestion et autres.

En pratique, très souvent, la présence de variables discrètes (ou entières) est inévitable dans la modélisation en optimisation et ces variables modifient considérablement la structure mathématique du problème. Il en résulte que des méthodes spécifiques à cette situation sont nécessaires.

Ce chapitre a pour objectif principal de présenter le contexte de l'optimisation linéaire multiobjectif discrète ainsi que l'optimisation non-linéaire multiobjectif en nombres entiers et de présenter quelques méthodes exactes permettant de caractériser totalement ou partiellement l'ensemble de solutions efficaces du problème.

2.4.1 Formulation du problème

Un problème de la programmation linéaire multiobjectif en nombres entiers est défini comme suit :

$$(P) \begin{cases} \max Z_1(x) = c^1 x \\ \max Z_2(x) = c^2 x \\ \vdots \\ \max Z_r(x) = c^r x \\ sc \\ x \in S = \{x \in \mathcal{R}^n | Ax \leq b, x \geq 0\} \\ x \text{ vecteur entier} \end{cases}$$

où $r \geq 2$, $c^i \in \mathcal{R}^n$ pour tout $i = \{1, \dots, r\}$; A est une $m \times n$ -matrice et b un m -vecteur à coefficients entiers.

La solution du problème de programmation multiobjectif linéaire à variables entières (P) consiste à trouver toutes les solutions efficaces au sens de la définition suivante :

Définition 2.4.1 *Un point $x^0 \in D$ est efficace ou Pareto optimale pour (P) si et seulement si il n'existe pas un autre point $x^1 \in S$ tel que $Z_i(x^1) \geq Z_i(x^0)$ pour tout $i \in \{1, \dots, r\}$ et $Z_i(x^1) > Z_i(x^0)$ pour au moins un $i \in \{1, \dots, r\}$.*

2.4.2 Solutions supportées et non supportées

Considérons le problème (P_R) relaxation continue du problème (P) :

$$(P_R) \begin{cases} \max Z_1(x) = c^1 x \\ \max Z_2(x) = c^2 x \\ \vdots \\ \max Z_r(x) = c^r x \\ sc \\ x \in S \end{cases}$$

Le théorème suivant s'applique à un programme avec objectifs linéaires et domaine réalisable convexe.

Théorème 2.4.1 (Geoffrion[63]) *Soit le problème unicritère linéaire suivant :*

$$(P_\lambda) \begin{cases} \max \sum_{i=1}^r \lambda_i Z_i(x) \\ sc \\ x \in S \text{ et } \lambda_i \in \Lambda, \forall i \in \{1, \dots, r\} \end{cases} \quad (2.4)$$

La solution x^* est efficace pour le problème (P_R) si et seulement si x^* est une solution optimale du problème paramétrique (P_λ) où $\Lambda = \{\lambda_i : \sum_{i=1}^r \lambda_i \geq 0, \forall i \in \{1, \dots, r\}\}$

D'après ce théorème, l'ensemble efficient du problème (P_R) sans les contraintes d'intégrité est bien caractérisé par les solutions du problème paramétrique (P_λ) . Ces solutions se trouvent sur la frontière de S , elles sont appelées *solutions supportées*.

Différemment du cas continu, la difficulté principale rencontrée lorsqu'on traite les problèmes multiobjectif à variables discrètes est l'existence de solutions efficaces pour (P) qui ne sont pas optimales pour (P_λ) et ce en raison de la non-convexité du domaine réalisable, ces solutions efficaces sont dites solutions *non supportées* (le front Pareto de (P) est l'union de l'ensemble des solutions supportées et de l'ensemble des solutions non supportées de (P)).

La nature des problèmes de programmation linéaire en variables continues et les problèmes de programmation linéaire en variables discrètes est différente. Contrairement à la programmation linéaire continue où on s'intéresse seulement aux solutions sommets du polyèdre, les solutions optimales du problème discret peuvent se trouver à l'intérieur du polyèdre et par conséquent la recherche d'une solution optimale d'un problème de la programmation en nombres entiers est souvent NP-difficile et peut être même NP-complet [118].

Il faut noter cependant que, si un problème d'optimisation combinatoire est facile à résoudre, il n'est pas de même pour sa version multiobjectif.

2.4.3 Quelques méthodes de résolution d'un problème *MOILP*

Plusieurs chercheurs, citons en particulier, Steuer et Choo, Klein et Hannan, Crema et Sylva, Gupta et Malhotra, Chergui et Moulai, Abbas et Moulai, Abbas et Chaa-bane ([128], [82], [46], [64], [43], [6], [2]), motivés par de nombreuses stratégies, se sont intéressés à caractériser totalement ou partiellement l'ensemble des solutions efficaces du problème de programmation linéaire multiobjectif en nombres entiers. Dans cette partie, quelques méthodes de résolution des problèmes *MOILP* sont exposées.

Méthode de D.Klein & E.Hannan :

La technique proposée par D.Klein & E.Hannan [82] peut être utilisée aussi bien pour identifier l'ensemble de toutes les solutions efficaces que pour en caractériser une partie seulement. Elle consiste à résoudre progressivement une séquence de programmes linéaires unicritère en nombres entiers avec des contraintes ajoutées à chaque étape. Les contraintes supplémentaires éliminent les solutions efficaces déjà trouvées, et font en sorte que les nouvelles solutions générées soient efficaces.

Méthode de M.Abbas et M.Moulai

Cette méthode a été proposée par M. Abbas & M. Moulai [6] pour la détermination de toutes les solutions efficaces du problème de programmation linéaire multiobjectif en nombres entiers. Elle peut être vue comme une alternative à celle de Gupta & Malhotra [64] (première procédure), où les auteurs ont proposé un autre test d'arrêt permettant à l'algorithme de fournir toutes les solutions efficaces.

Méthode de M.Abbas et D.Chaabane

M. Abbas et D. Chaabane ont proposé dans [2] la méthode de détermination des solutions efficaces dans l'espace des variables discrètes, cette méthode est une forme améliorée de la méthode de Gupta et Malhotra où le test d'arrêt est corrigé pour déterminer toutes les solutions efficaces du problème (P) sans n'en manquer aucune. Cette méthode utilise les mêmes paramètres que ceux définis dans la méthode précédente.

Méthode A.Sylva et J.Crema

La méthode développée par Crema et Sylva [46] est une variante de celle de Klein et Hannan étudiée précédemment. Son principe repose sur la résolution d'une succession de programmes linéaires en nombres entiers optimisant à chaque étape une combinaison positive des critères. Un ensemble de contraintes est rajouté à chaque fois assurant la détection d'une nouvelle solution efficace. A la fin, la méthode fournit l'ensemble de toutes les solutions non dominées du problème de programmation linéaire discrète à objectifs multiples.

Les auteurs considèrent que les données du problème (P) sont entières, aussi bien la

matrice des contraintes A , le vecteur second membre b que les coefficients c_j^i de tous les critères.

2.5 Programmation non linéaire multiobjectif en nombres entiers MOINLP

Etant donné, le grand nombre d'applications dans le domaine de l'économie, plusieurs auteurs, se sont intéressés à la programmation non linéaire multiobjectif en nombres entiers (MOINLP). Dans cette partie, nous nous sommes intéressés à la résolution d'un problème fractionnaire linéaire multiobjectif (Chergui M. E-A. et Moulai M. [43]). La méthode sera utilisée et réadaptée pour la résolution d'un problème linéaire bi-objectifs en nombres entiers dans le chapitre 5.

2.5.1 Programmation fractionnaire linéaire multiobjectif en nombres entiers MOIFLP

Un problème linéaire multiobjectif en nombres entiers est défini comme suit :

$$(P) \begin{cases} \max Z_1(x) = \frac{c^1 x + \alpha_1}{d^1 x + \beta_1} \\ \max Z_2(x) = \frac{c^2 x + \alpha_2}{d^2 x + \beta_2} \\ \vdots \\ \max Z_r(x) = \frac{c^r x + \alpha_r}{d^r x + \beta_r} \\ sc \\ x \in S = \{x \in \mathcal{R}^n | Ax \leq b, x \geq 0\} \\ x \text{ vecteur entier} \end{cases}$$

où $r \geq 2$, c^i, d^i sont des vecteurs de $\mathcal{R}^n$, α_i, β_r sont des réels, pour tout $i = \{1, \dots, r\}$; A est une $m \times n$ -matrice et b un m -vecteur à coefficients entiers.

La méthode proposée par M.Chergui et M.Moulai [43] est basée sur la procédure "Branch and Bound" et des coupes, appelées, les coupes efficaces pour générer toutes les solutions efficaces entières pour le problème. L'approche est basée sur la résolution d'un problème de programmation fractionnaire linéaire, à chaque étape l

$$(P_1) : \max \{Z_i = \frac{c^i, x + \alpha_i}{d^i, x + \beta_i} \in S_l\}$$

(L'indice i est pris arbitrairement dans $1, \dots, r$), avec $S_0 = S$ et sans la contrainte d'intégrité des variables. Si la solution optimale du problème P_l est entière, elle est

comparée à l'ensemble des solutions potentiellement efficaces déjà trouvées et l'ensemble des solutions efficaces est actualisé. L'approche utilise une coupe appelé la coupe efficace capable de supprimer les solutions entières qui ne sont pas efficaces pour le problème P_l . Dans le cas où cette solution optimale n'est pas entière, deux nouveaux programmes fractionnaires linéaires sont créés en utilisant le processus de branchement bien connu dans la méthode de Branch and Bound. Chacun d'eux sera résolu comme le problème P_l . Soit x_l^* la première solution entière obtenue après avoir résolu le problème P_l en utilisant, éventuellement, la procédure Branch and Bound, on définit I_l comme l'ensemble des indices des variables de base. et N_l comme l'ensemble des indices des variables hors base de x_l^* . Soit γ_i^j la $j^{\text{ème}}$ composante du vecteur de gradient réduit $\bar{\gamma}_i$ défini comme suit, pour chaque $i \in \{1, 2, \dots, r\}$; $\bar{\gamma}_i = \bar{\beta}_i \bar{c}_i - \bar{\alpha}_i \bar{d}_i$, où $\bar{\beta}_i, \bar{c}_i, \bar{\alpha}_i, \bar{d}_i$ sont des valeurs relatives à la solution optimale x_l^* . Le calcul de γ_i est suffisant pour déterminer la direction de croissance de chaque critère.

On définit la contrainte :

$$\sum_{j \in H_l} x_j \geq 1. \quad (2.5)$$

où,

$$H_l = \{j \in N_l \mid \exists i \in \{1, \dots, r\} \bar{\gamma}_j^i > 0\} \cup \{j \in N_l \mid \bar{\gamma}_j^i = 0, \forall i \in \{1, \dots, r\}\} \quad (2.6)$$

L'inégalité (2.5) est appelée coupe efficace [43]. Cette méthode sera détaillée dans le chapitre 5.

2.6 Conclusion

Nous avons présenté dans ce chapitre les principaux concepts de l'optimisation multiobjectif tels que la modélisation d'un problème multiobjectif, la notion de la dominance de Pareto et la structure de la surface de compromis. La méthode (Chergui M. E-A. et Moulaï M. [43]) sera utilisée et réadaptée pour la résolution d'un problème linéaire biobjectif en nombre entiers, afin, de résoudre le problème quadratique indéfini en nombres entiers (voir chapitre 5).

Chapitre 3

Programmation mathématique à deux niveaux

3.1 Introduction

Comme dans l'optimisation à un seul niveau, les problèmes de programmation à deux niveaux PBN peuvent être divisés en deux catégories : ceux dont les solutions sont continues, également connues sous le nom des problèmes de programmation à deux niveaux continu. et ceux dont les solutions entières sont connues sous le nom des problèmes de programmation à deux niveaux discrète ou en nombres entiers. Les PBNs combinant des variables de décision continues et discrètes sont considérées comme des PBNs mixtes. Un problème de programmation à deux niveaux PBN est une modélisation d'un problème mathématique hiérarchique où l'ensemble de toutes les variables est partitionné entre un vecteur x représentant la variable de décision du premier niveau et un vecteur y pour la variable de décision du second niveau. Sa formulation se présente comme suit :

$$\text{PBN} \left\{ \begin{array}{l} \max_{(x,y)} F(x,y) \\ s.t \\ \left\{ \begin{array}{l} g^1(x,y) \leq b^1 \\ \text{où } y \text{ résout} \\ \max_y f(x,y) \\ g^2(x,y) \leq b^2 \end{array} \right. \end{array} \right. \quad (3.1)$$

où,

$$x \in \mathbb{R}^{n_1} \text{ et } y \in \mathbb{R}^{n_2} \quad (3.2)$$

sont des variables contrôlées par le décideur du premier niveau et le décideur du second niveau, respectivement ;

$$F, f : \mathbb{R}^n \mapsto \mathbb{R}, \quad n = n_1 + n_2 \quad (3.3)$$

sont les fonctions objectifs du premier niveau de décision et de second niveau de décision respectivement, et les fonctions vectorielles

$$g^1 : \mathbb{R}^n \mapsto \mathbb{R}^{m_1}, \quad n = n_1 + n_2, \quad g^2 : \mathbb{R}^n \mapsto \mathbb{R}^{m_2}, \quad n = n_1 + n_2, \quad (3.4)$$

sont respectivement les contraintes du premier niveau et du second niveau. avec b^i sont des vecteurs de $\mathbb{R}^m, i = 1, 2$.

La présence des contraintes dites couplantes de premier niveau $g^1(x, y) \leq b^1$ attribue à cette formulation un caractère plus général.

La formulation générale d'un programme multi-niveaux, qui est en fait une généralisation du problème à deux niveaux est la suivante :

$$\text{PMN} \left\{ \begin{array}{l} \max_{(x_1, x_2, \dots, x_p)} f^1(x_1, x_2, \dots, x_p) \\ g^1(x_1, x_2, \dots, x_p) \leq b^1 \\ \text{où } (x_2, \dots, x_p) \text{ résout} \\ \left\{ \begin{array}{l} \max_{(x_1, x_2, \dots, x_p)} f^2(x_1, x_2, \dots, x_p) \\ g^2(x_1, x_2, \dots, x_p) \leq b^2 \\ \cdot \\ \cdot \\ \cdot \end{array} \right. \\ \text{où } x_p \text{ résout} \\ \left\{ \begin{array}{l} \max_{(x_1, \dots, x_p)} f^p(x_1, x_2, \dots, x_p) \\ g^p(x_1, x_2, \dots, x_p) \leq b^p \end{array} \right. \end{array} \right. \quad (3.5)$$

3.2 Quelques applications de l'optimisation à deux niveaux

Les structures hiérarchiques de problèmes apparaissent dans de nombreuses applications de la vie réelle où les décisions du second niveau dépendent des actions du premier niveau. En effet, diverses applications de problèmes d'optimisation à deux niveaux apparaissent dans la pratique.

- **Problème de paramétrage du péage :** Dans le domaine des transports, l'optimisation à deux niveaux apparaît couramment dans le problème de l'établissement des péages. Prenons l'exemple d'un réseau d'autoroutes exploité par le gouvernement. Le gouvernement veut maximiser ses revenus en choisissant la tarification optimale pour les autoroutes. Cependant, le gouvernement ne peut maximiser ses revenus qu'en tenant compte de la problématique des usagers de l'autoroute. Pour toute structure fiscale donnée, les usagers de l'autoroute résolvent leur propre problème d'optimisation, où ils minimisent leurs coûts de déplacement en décidant entre utiliser les autoroutes ou un itinéraire alternatif. Dans ces circonstances, le problème du gouvernement doit être formulé comme un problème d'optimisation à deux niveaux. Le premier niveau comprend les objectifs et les contraintes du gouvernement, et le second niveau comprend les utilisateurs de l'autoroute (objectifs et contraintes pour une structure fiscale donnée). Il est à noter que le gouvernement ne pourra identifier les revenus générés par une structure fiscale particulière qu'en résolvant le problème de niveau inférieur qui détermine dans quelle mesure les autoroutes sont utilisées.
- **Le problème de la menace des perturbateurs :** Le problème de la menace des perturbateurs peut être modélisé comme un problème de programmation à deux niveaux. L'analyse de vulnérabilité implique deux agents qui tentent de prendre des décisions optimales en fonction de leurs objectifs respectifs : les perturbateurs (c'est-à-dire les agents destructeurs, le premier niveau, le leader) attaquent le système dans le but de maximiser les dommages, tandis que l'opérateur du réseau (c'est-à-dire un agent du second niveau, suiveur) réagit pour minimiser ces dommages. Le problème d'optimisation du premier niveau est donc associé à l'agent perturbateur qui sélectionnera les éléments du système à attaquer afin de maximiser les dommages (par exemple, les lignes d'un réseau électrique). Dans le modèle du second niveau, l'opérateur du système réagit à ces attaques. Il détermine le fonctionnement optimal du système électrique (par exemple, délestage, redispatching de la production, commutation de lignes) qui minimisera les dommages causés par les perturbateurs.
- **Chaîne d'approvisionnement :** de nombreux problèmes de gestion ont été formulés en tant que des problèmes de programmation à deux niveaux.

Exemple 3.2.1 *Gestion de la chaîne d'approvisionnement : De nombreux problèmes de gestion de la chaîne d'approvisionnement peuvent être formulés comme des problèmes de programmation à deux niveaux. Le modèle actuel de la chaîne d'approvisionnement est avec deux centres de distribution et une usine d'assemblage.*

La notation des variables pour le modèle de chaîne d'approvisionnement est la suivante :

i : Le nombre de centres de distribution,

j : Le nombre d'usines d'assemblage,

D_{ij} : La demande du j^{eme} usine d'assemblage au i^{eme} centre de distribution,

P_{ij} : Le prix du produit que le i^{eme} centre de distribution fournit au j^{eme} usine d'assemblage,

X_i : La quantité totale de produits que le i^{eme} centre de distribution contient,

$C_i(X_i)$: Le coût unitaire pour le i^{eme} centre de distribution pour acheter le produit X_i ,

W_i : La contrainte de capacité pour le i^{eme} centre de distribution,

A_j : La quantité totale de produits que le j^{eme} usine d'assemblage a besoins,

$S_{ij}(D_{ij})$: Le coût unitaire du produit que le j^{eme} usine d'assemblage commande au i^{eme} centre de distribution,

$T_{ij}(D_{ij})$: Le coût unitaire de transport du produit livré depuis le i^{eme} centre de distribution au j^{eme} usine d'assemblage,

h_j : Le coût de possession unitaire pour le j^{eme} usine d'assemblage.

Cette étude suppose que les centres de distribution appartiennent au premier niveau, tandis que les usines d'assemblage sont au second niveau. L'objectif du premier niveau est de maximiser les bénéfices totaux des centres de distribution, tandis que l'objectif du second niveau est de minimiser les coûts totaux des usines d'assemblage.

Les paramètres associés sont définis comme suit :

$i = 2, j = 1, P_{11} = 100, P_{21} = 150, C_1(X_1) = 40, C_2(X_2) = 50, W_1 = 30, W_2 = 20, S_{11}(D_{11}) = 10, S_{21}(D_{21}) = 15, T_{11}(D_{11}) = 20, T_{21}(D_{21}) = 25, h_1 = 5$. Ainsi, le modèle correspondant peut être représenté par :

$$PBL \left\{ \begin{array}{l} \max_{(x,y)} F(x,y) = 100D_{11} + 150D_{21} - 40X_1 - 50X_2 \\ s.c \\ \left\{ \begin{array}{l} \min_y f(x,y) = 135D_{11} + 195D_{21} \\ D_{11} \leq X_1 \\ D_{21} \leq X_2 \\ X_1 \leq 30 \\ X_2 \leq 20 \\ X_1 + X_2 \geq 50 \\ X_i \geq 0, \quad D_{ij} \geq 0, \quad i = 1, 2, \quad j = 1 \end{array} \right. \end{array} \right. \quad (3.6)$$

3.3 Programmation linéaire à deux niveaux

La formulation d'un problème de programme à deux niveaux linéaire PBL est la suivante :

$$PBL \left\{ \begin{array}{l} \max_{(x,y)} F(x,y) = c^1x + c^2y \\ s.t \\ \left\{ \begin{array}{l} A^1x + A^2y \leq b^1 \\ \text{où } y \text{ résout} \\ \max_y f(x,y) = d^1x + d^2y \\ B^1x + B^2y \leq b^2 \\ x \geq 0, y \geq 0 \end{array} \right. \end{array} \right. \quad (3.7)$$

où,

$$x \in \mathbb{R}^{n_1} \quad \text{et} \quad y \in \mathbb{R}^{n_2} \quad (3.8)$$

A^i sont des matrices de dimension $m_1 \times n_i$, B^i sont des matrices de dimension $m_2 \times n_i$, b^i sont des vecteurs de dimension $\mathbb{R}^{m_i}$, $i = \{1, 2\}$.

c^1, d^1 sont des vecteurs colonnes de dimension $\mathbb{R}^{n_1}$; c^2, d^2 sont des vecteurs colonnes de dimension $\mathbb{R}^{n_2}$. Les contraintes $A^1x + A^2y \leq b^1$ (resp. $B^1x + B^2y \leq b^2$), sont des contraintes du premier (resp. second) niveau. La fonction linéaire $F(x, y)$ (resp. $f(x, y)$) représente la fonction économique du premier niveau (resp. du second niveau), x et y sont des variables contrôlées par le décideur du premier niveau et le décideur du second niveau, respectivement.

1. L'ensemble $S = \{(x, y) : x \geq 0, y \geq 0, A^1x + A^2y \leq b^1, B^1x + B^2y \leq b^2\}$ définit le domaine réalisable du problème PBL.
2. $S(x) = \{y \in \mathbb{R}^{n_2} : B^2y \leq b^2 - B^1x\}$ est l'ensemble des solutions réalisables du second niveau pour chaque valeur de la variable du premier niveau x .

Soit $S(X) = \{x \in \mathbb{R}^{n_1} : \exists y \in \mathbb{R}^{n_2}, A^1x + A^2y \leq b^1, B^1x + B^2y \leq b^2, x \geq 0, y \geq 0\}$ la projection de S sur $\mathbb{R}^{n_1}$. Pour chaque $x \in S(X)$, le décideur du second niveau résout le problème $(P(X))$:

$$(P(X)) \begin{cases} \max & f(x, y) \\ \text{s.t} & \\ & y \in S(x). \end{cases} \quad (3.9)$$

Soit $M(x)$ l'ensemble des solutions optimales de (3.9) :

$$M(x) = \{y^* \in \mathbb{R}^{n_2} : y^* \in \arg \max_y f(x, y) : y \in S(x)\} \quad (3.10)$$

La région réalisable du décideur du premier niveau, appelée région inductible IR, est implicitement définie par le problème d'optimisation du second niveau :

$$\text{IR} = \{(x, y^*) : x \in S_1, y^* \in M(x)\}$$

Exemple 3.3.1 *Dempe S.[50] Considérons le problème de programme à deux niveaux linéaire suivant :*

$$PBL \left\{ \begin{array}{l} \min_{(x,y)} F(x, y) = x + 3y \\ \text{s.t} \\ \left\{ \begin{array}{l} x \leq 6 \\ x \geq 1 \\ \text{où } y \text{ résout} \\ \min_y f(x, y) = -y \\ x + y \leq 8 \\ x + 2y \leq 13 \\ x + 4y \geq 8 \\ y \geq 0 \end{array} \right. \end{array} \right. \quad (3.11)$$

$$y(x) = \begin{cases} (13-x)/2 & \text{pour } 1 \leq x \leq 3 \\ 8-x & \text{pour } 3 \leq x \leq 6 \end{cases} \quad (3.12)$$

$$IR = \{(13 - x)/2 : 1 \leq x \leq 3\} \quad \cup \quad \{8 - x : 3 \leq x \leq 6\}$$

FIGURE 3.1 – Le problème de programme à deux niveaux linéaire PBL

3.4 Programmation non linéaire à deux niveaux

Considérons la formulation (3.1) d'un problème de programmation à deux niveaux PBN. Une valeur du vecteur x étant donnée, le vecteur y est choisi comme étant solution optimale $y = y(x)$ d'un problème d'optimisation paramétré en x :

$$y(x) \in M(x) = \{y^* \in \mathbb{R}^{n_2} : y^* \in \arg \max_y f(x, y) : y \in S(x)\} \quad (3.13)$$

Ce problème comme dans le cas linéaire représente le problème de second niveau. L'application multivoque :

$$M : \mathbb{R}^{n_1} \mapsto 2^{\mathbb{R}^{n_2}} \quad (3.14)$$

où $2^{\mathbb{R}^{n_2}}$, désigne l'ensemble de toutes les parties de $\mathbb{R}^{n_2}$. $M(x)$ représente l'ensemble des solutions optimales du problème (3.13). $y(x)$ est appelée réaction rationnelle du suiveur (décideur de second niveau) lorsque le leader (décideur du premier niveau) choisit x . Connaissant cette réaction, le problème PBN peut se formuler comme suit :

$$\text{PBN} \quad \left\{ \begin{array}{l} \max_x F(x, y) \\ s.t \\ \left\{ \begin{array}{l} g^1(x, y) \leq b^1 \\ y(x) \in M(x) \end{array} \right. \end{array} \right. \quad (3.15)$$

La valeur de la fonction objectif du second niveau $f(x)$ reste ambiguë du point de vue du leader (qui n'exerce de contrôle que sur x) si l'ensemble des solutions optimales $M(x)$ n'est pas réduit à un singleton. Car, pour ce cas le leader pourrait ne pas atteindre son optimum Bard [16].

3.5 Multiplicité des solutions optimales du second niveau

La notion de solution dans la programmation à deux niveaux n'est pas facile à définir. Elle dépend du nombre de solutions optimales dans le problème du second niveau pour certaines valeurs de paramètres. L'exemple suivant en est une illustration :

$$\text{PBNL} \left\{ \begin{array}{l} \min_{(x,y)} F(x,y) = x^2 + y^2 \\ \text{s.c} \\ \left\{ \begin{array}{l} x \leq 1 \\ \text{où } y \text{ résout} \\ \min_y f(x,y) = -xy \\ y \leq 1 \\ x \geq 0, y \geq 0 \end{array} \right. \end{array} \right. \quad (3.16)$$

Alors, en évaluant le problème de second niveau et en insérant la solution optimale de ce problème dans la fonction objectif du premier niveau, on obtient :

$$M(x) = \begin{cases} \{0\}, & x > 0 \\ \{1\}, & x < 0 \\ [0 \ 1], & x = 0 \end{cases} \quad (3.17)$$

$$F(x, y(x)) = \begin{cases} = x^2, & x > 0 \\ = 1 + x^2, & x < 0 \\ \in [0 \ 1], & x = 0 \end{cases} \quad (3.18)$$

La valeur de la fonction $F(x, y(x))$ au point $y = 0$ est vague. La valeur de la fonction infiniment petit $F(x, y(x))$ est égale à zéro mais cette valeur n'est atteinte que si $F(0, y(0)) = 0$. Cette situation est appelée position optimiste. Si ceci n'est pas le cas, alors le problème à deux niveaux n'a pas de solutions. Pour pallier cette ambiguïté, au moins deux approches ont été proposées. Dans le cas de la programmation à deux niveaux optimiste, on suppose que l'ensemble $M(x)$, n'est pas singleton, le leader est autorisé à choisir l'élément dans $M(x)$ qui lui convient le mieux. Dans cette situation, un point (x^*, y^*) est appelé une solution optimale optimiste du problème (*PBN*) si les règles suivantes sont vraies :

$$\left\{ \begin{array}{l} x^* \in S \\ g^1(x^*, y^*) \leq b_1 \\ y^* \in M(x^*) \\ F(x^*, y^*) \geq F(x^*, y), \forall y \in M(x^*) \end{array} \right. \quad (3.19)$$

Lorsque la coopération entre le leader et le suiveur n'est pas autorisée, ou que le leader a une aversion au risque et souhaite limiter les dommages résultant d'une sélection

indésirable de la part du suiveur. Dans cette situation, un point (x^*, y^*) est appelé une solution optimale pessimiste du problème (PBN) si les règles suivantes sont vraies :

$$\begin{cases} x^* \in S \\ g^1(x^*, y^*) \leq b_1 \\ y^* \in M(x^*) \\ F(x^*, y^*) \leq F(x^*, y), \forall y \in M(x^*) \end{cases} \quad (3.20)$$

3.5.1 L'approche optimiste

Cette approche est utilisée par le leader en supposant que le suiveur est disposé à coopérer, c'est-à-dire que le suiveur choisit une solution $y(x) \in M(x)$, la meilleure du point de vue du leader. On désigne par :

$$m^o(x) = \{\max_y F(x, y) : y \in M(x)\} \quad (3.21)$$

la valeur optimiste de la fonction objectif du premier niveau. Alors, l'approche optimiste du problème de programmation à deux niveaux est réduite à :

$$\max_{(x,y)} \{m^o(x) : g^1(x, y) \leq b^1\} \quad (3.22)$$

$(x^*, y(x^*))$ tel que x^* est solution de (3.22), et $y(x^*)$ solution de (3.21) pour $x = x^*$, est également une solution optimale pour le problème :

$$\max_{(x,y)} \{F(x, y) : g^1(x, y) \leq b^1, y \in M(x)\} \quad (3.23)$$

et vice versa, à condition que le dernier ait une solution optimale. La plupart des contributions à la programmation à deux niveaux sont consacrées à ce problème.

Le théorème suivant détermine les conditions d'existence d'une solution optimale pour le problème optimiste à deux niveaux (avec l'hypothèse que la contrainte du premier niveau ne dépend pas des variables du second niveau).

Théorème 3.5.1 *Dempe S.[50] Si l'ensemble $\{(x, y) : g^1(x) \leq b_1, g^2(x, y) \leq b_2\}$ est non vide et compact, et l'hypothèse de qualification des contraintes est satisfaite pour tout x tel que $g^1(x) \leq b_1$,), alors la fomulation optimiste pour le problème (PBN) a une solution optimale.*

3.5.2 L'approche pessimiste

Lorsque la coopération entre le leader et le suiveur n'est pas autorisée, l'approche optimiste semble impossible, ou alors lorsque l'aptitude du suiveur à respecter des ententes n'est pas assurée. Alors dans cette situation déplaisante, une voie à suivre pour

le leader consiste à se prémunir contre les choix indésirables du suiveur. Ce qui conduit au problème :

$$\max_{(x,y)} \{m^p(x) : g^1(x, y) \leq b^1\} \quad (3.24)$$

avec

$$m^p(x) = \{\min_y F(x, y) : y \in M(x)\} \quad (3.25)$$

représente la pire valeur de la fonction objectif du premier niveau qui peut être atteinte sur l'ensemble des solutions du problème du second niveau.

Le théorème suivant annonce les conditions d'existence d'une solution optimale pour le problème optimiste à deux niveaux (avec l'hypothèse que la contrainte du premier niveau ne dépend pas des variables du second niveau).

Théorème 3.5.2 *Dempe S.[50] Supposons que l'application multivoque $M(\cdot)$ est semi-continue inférieurement en tout point x tel que $g^1(x) \leq b_1$ et que l'hypothèse de qualification des contraintes est satisfaite en chacun de ces points. Alors, la formulation pessimiste pour le problème (PBN) a une solution optimale.*

Il faudrait rappeler que l'application multivoque $M(\cdot)$ est dite semi-continue inférieurement au point $x = x^0$ si, pour tout ensemble ouvert A avec $M(x^0) \cap A \neq \emptyset$, il existe un voisinage ouvert B de x^0 avec $M(x) \cap A \neq \emptyset$ pour tout $x \in B$.

3.6 Autres formulations du problème à deux niveaux

Sous l'hypothèse d'univocité de l'application $M(x)$ et celle de la compacité du domaine réalisable, le problème à deux niveaux se reformule comme un problème à un seul niveau en x :

$$\left\{ \begin{array}{l} \max_x F(x, y(x)) \\ s.t \\ \left\{ \begin{array}{l} x \in S_x \\ g^1(x, y(x)) \leq b^1 \end{array} \right. \end{array} \right. \quad (3.26)$$

où $S_x = \{x \in \mathbb{R}^{n_1} : \exists y \in \mathbb{R}^{n_2}, g^2(x, y(x)) \leq b^2\}$ représente la trace de l'ensemble S des variables du premier niveau x .

$S = \{(x, y) : x \geq 0, y \geq 0, g^1(x, y) \leq b^1, g^2(x, y) \leq b^2\}$.

Lorsque l'application $M(x)$ n'est pas univoque, on obtient la formulation suivante :

$$\left\{ \begin{array}{l} \max_x F(x, y) \\ s.c \\ \left\{ \begin{array}{l} (x, y) \in S_x \\ y \in M(x) \end{array} \right. \end{array} \right. \quad (3.27)$$

La résolution de ces problèmes est compliquée en raison de la présence des fonctions et des ensembles définis implicitement.

Étant donné les hypothèses de régularité sur le problème de second niveau ($M(x)$ est réduit en singleton) d'un problème de programmation à deux niveaux PBN, on peut remplacer ce dernier par ses conditions d'optimalité de Karush-Kuhn-Tucker (KKT). Il en résulte un programme mathématique non linéaire à un seul niveau dont la formulation est :

$$\text{PBN(KKT)} \left\{ \begin{array}{l} \max_{(x,y)} F(x, y) \\ s.c \\ \left\{ \begin{array}{l} g^1(x, y) \leq b^1 \\ g^2(x, y) \leq b^2 \\ \nabla_y f(x, y) + \lambda^T \nabla_y (g^2(x, y) - b^2) = 0 \\ \lambda^T (g^2(x, y) - b^2) = 0 \\ \lambda \geq 0, \end{array} \right. \end{array} \right. \quad (3.28)$$

où $\lambda \in \mathbb{R}^{m_2}$. Donc, la résolution du problème PBN, revient à résoudre directement le programme mathématique équivalent (3.28).

3.7 Conditions d'optimalité-Approche optimiste

La formulation des conditions d'optimalité pour les problèmes de programmation à deux niveaux se fait habituellement avec une formulation appropriée du problème comme programme d'optimisation à un niveau.

3.7.1 Cas linéaire

Depuis 1980, des efforts significatifs ont été consacrés à la compréhension des concepts associés à la programmation à deux niveaux. Pour ces problèmes, bien que le domaine induit soit non convexe en général, il possède certaines propriétés des ensembles convexes [16]. Les plus importantes étant les suivantes :

Propriété 3.7.1 (corollaire 3.1[16]). *Tout point extrême du domaine induit IR est aussi un point extrême du domaine réalisable S du problème linéaire à deux niveaux.*

Propriété 3.7.2 (corollaire1 [15]). *Lorsqu'un problème (PBL) admet une solution optimale, cette solutuion est atteinte en un point extrême du domaine induit IR .*

3.7.2 Utilisation des conditions de KKT

Si les conditions de KKT sont utilisées afin de remplacer le problème de second niveau du problème de programmation biniveaux (PBN) par un système d'équations d'égalités et d'inégalités, on obtient le problème :

$$PBN(KKT) \left\{ \begin{array}{l} \max_{(x,y)} F(x,y) \\ s.t \\ \left\{ \begin{array}{l} g^1(x,y) \leq b^1 \\ g^2(x,y) \leq b^2 \\ \nabla_y f(x,y) + \lambda^T \nabla_y (g^2(x,y) - b^2) = 0 \\ \lambda^T (g^2(x,y) - b^2) = 0 \\ \lambda \geq 0, \end{array} \right. \end{array} \right. \quad (3.29)$$

il est possible d'obtenir des conditions nécessaires d'optimalités pour un problème de programmation biniveaux (PBN) avec cette approche, lorsque le problème de second niveau est un problème paramétrique convexe et seulement dans l'approche optimiste. Sous des hypothèses de qualification des contraintes appropriées, un problème de programmation biniveaux (PBN) est équivalent à sa formulation de KKT, à savoir ($PBN(KKT)$). La propriété suivante énonce ce résultat.

Propriété 3.7.3 ([16]). *Si (x^*, y^*) est une solution optimale du problème (PBN) et si y^* est une solution optimale du problème de second niveau pour x fixé, alors il existe un vecteur de multiplicateurs λ^* tel que (x^*, y^*, λ^*) est une solution optimale du problème ($PBN(KKT)$).*

Remarque 3.7.1 *L'utilisation des conditions de KKT, pour la résolution des problèmes à deux niveaux (PBN) est très répondu en littérature. Néanmoins, la difficulté réside dans la résolution des problèmes non linéaires ($PBN(KKT)$) équivalents, à cause de la contrainte de complémentarité qui est non linéaire.*

3.8 Quelques méthodes de résolution pour les problèmes de programmation à deux niveaux

3.8.1 Méthode de la fonction de pénalité :

Les fonctions de pénalité ont été largement utilisées en optimisation non linéaire. et se trouvent dans presque tous les manuels d'optimisation non linéaire Bazaraa et al. [17]. Les méthodes de pénalité font partie des premières tentatives de résolution des problèmes de programmation à deux niveaux Aiyoshi E., et Shimizu K. [7]. L'idée des algorithmes basés sur cette méthode, proposée d'abord par Bard et Falk [14], est de placer la contrainte de complémentarité dans la fonction objectif du problème ($PBN(KKT)$) pondérée par une constante positive K suffisamment grande. L'idée est reprise par Anandalingam et White [9], en utilisant une fonction de pénalité qui est l'écart entre les valeurs des fonctions objectifs primale et duale du suiveur. Dans sa première version l'algorithme ne permettait de donner qu'une solution optimale locale, par la suite il a été amélioré et permet de donner une solution globale. Différents algorithmes utilisant une méthode de pénalité extérieure sont proposés par Shimizu et Lu [120] qui exigent simplement la convexité des fonctions objectifs et des fonctions contraintes pour garantir la convergence vers une solution optimale globale.

3.8.2 La méthode descente

Considérons le problème de la programmation à deux niveaux (PBN) :

$$PBN \quad \begin{cases} \max_x F(x, y) \\ s.t \\ \begin{cases} g^1(x, y) \leq b^1 \\ y(x) \in M(x) \end{cases} \end{cases} \quad (3.30)$$

$M(x)$ représente l'ensemble des solutions optimales du problème de second niveau (3.13), et est supposé un singleton pour toutes les valeurs du paramètre x .

$y(x)$ est appelée réaction rationnelle du suiveur (décideur de second niveau) lorsque le leader (décideur du premier niveau) choisit x , supposé unique.

$$y(x) \in M(x) = \{y^* \in \mathbb{R}^{n_2} : y^* \in \arg \max_y f(x, y) : y \in S(x)\} \quad (3.31)$$

Étant donné un point réalisable x , on tente de trouver une direction réalisable $d \in \mathbb{R}^{n_1}$ le long de laquelle l'objectif du premier niveau croît. Un nouveau point $x + \alpha d (\alpha > 0)$

est calculé de manière à assurer une augmentation raisonnable de F tout en maintenant la faisabilité du problème à deux niveaux. Cependant, un problème majeur est la détermination du gradient de l'objectif du premier niveau à un point réalisable.

3.8.3 La méthode de Branch and Bound :

Les méthodes de Branch and Bound sont largement appliquées aux programmes convexes à deux niveaux. Plusieurs approches exploitent la complémentarité entre les multiplicateurs et les variables d'écarts qui découlent des conditions KKT du problème du second niveau. En fait, lorsque le problème du second niveau est convexe et régulier, il peut être remplacé par ses conditions de Karush -kuhn-Tucker (KKT), ce qui donne le problème non convexe à un seul niveau ($PBN(KKT)$). Une des techniques développée par Bard J. F. et Falk J. E. [14], consiste à entourer l'ensemble des solutions réalisables du problème non convexe ($PBN(KKT)$) par un polyèdre et qu'on subdivise ensuite en sous-ensembles disjoints. On minimise la fonction objectif de ($PBN(KKT)$) sur chaque sous-ensemble, puis on sélectionne la plus petite valeur parmi toutes les valeurs obtenues. Après un test de cette valeur, on la prend comme valeur optimale globale si le test est positif, sinon on subdivise le sous-ensemble correspondant à cette valeur en nouveaux polyèdres et on recommence l'opération. Fortuni-Amat J. et McCarl B. [62] proposent une méthode similaire à la précédente, leur idée de base est de supprimer la contrainte de complémentarité de ($PBN(KKT)$) et de résoudre le sous-problème linéaire résultant. A chaque itération, on vérifie si la contrainte de complémentarité est satisfaite. Si oui, le point correspondant se trouve dans le domaine induit, d'où il présente une solution potentielle, sinon on utilise un procédé de branch and bound pour examiner toutes les combinaisons des contraintes de complémentarité relâchées (Shimuzu K., Ishizuka Y., Bard J. F. [121]).

3.9 La programmation à deux niveaux en nombres entiers

Un problème de programmation à deux niveaux en nombres entiers PBNE, est un problème de programmation à deux niveaux avec les contraintes d'intégralité portant sur uniquement le premier niveau ou uniquement sur le second niveau ou bien sur les deux niveaux. Le problème de programmation à deux niveaux en nombres entiers PBNE peut avoir une des trois formulations :

$$\text{PBNE} \left\{ \begin{array}{l} \max_{(x,y)} F(x,y) \\ s.t \\ \left\{ \begin{array}{l} g^1(x,y) \leq b^1 \\ \text{où } y \text{ résout} \\ \max_y f(x,y) \\ g^2(x,y) \leq b^2 \\ x \in \mathbb{R}^{n_1} \text{ et entier} \\ y \in \mathbb{R}^{n_2} \end{array} \right. \end{array} \right. \quad (3.32)$$

Ou,

$$\text{PBNE} \left\{ \begin{array}{l} \max_{(x,y)} F(x,y) \\ s.t \\ \left\{ \begin{array}{l} g^1(x,y) \leq b^1 \\ \text{où } y \text{ résout} \\ \max_y f(x,y) \\ g^2(x,y) \leq b^2 \\ x \in \mathbb{R}^{n_1} \\ y \in \mathbb{R}^{n_2} \text{ et entier} \end{array} \right. \end{array} \right. \quad (3.33)$$

Ou,

$$\text{PBNE} \left\{ \begin{array}{l} \max_{(x,y)} F(x,y) \\ s.t \\ \left\{ \begin{array}{l} g^1(x,y) \leq b^1 \\ \text{où } y \text{ résout} \\ \max_y f(x,y) \\ g^2(x,y) \leq b^2 \\ x \in \mathbb{R}^{n_1} \text{ et entier} \\ y \in \mathbb{R}^{n_2} \text{ et entier} \end{array} \right. \end{array} \right. \quad (3.34)$$

Exemple 3.9.1 Considérons le problème de programme à deux niveaux linéaire en nombres entiers *PBLE* suivant :

$$\text{PBLE} \left\{ \begin{array}{l} \min_{(x,y)} F(x,y) = -x - 10y \\ \text{s.t} \\ \left\{ \begin{array}{l} x \geq 0, \quad x \text{ entier} \\ \text{où } y \text{ résout} \\ \min_y f(x,y) = y \\ -25x + 20y \leq 30 \\ x + 2y \leq 10 \\ 2x - y \leq 15 \\ 2x + 10y \geq 15 \\ y \geq 0, \quad y \text{ entier} \end{array} \right. \end{array} \right. \quad (3.35)$$

L'ensemble réalisable de ce problème ainsi que sa relaxation continue sont présentés à la figure en dessous. La ligne épaisse est l'ensemble réalisable du problème de relaxation continue. L'unique solution optimale globale du problème de programmation à deux niveaux en nombres entiers est trouvée au point $(x^*, y^*) = (2; 2)^T$. La valeur de la fonction objective du premier niveau est $F_{opt} = -22$ et celle du second niveau est $f_{opt} = 2$

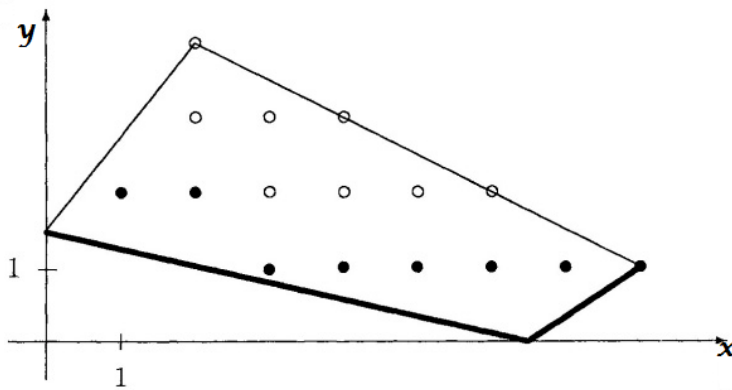


FIGURE 3.2 – Le problème de programme à deux niveaux linéaire en nombres entiers PBLE

Les problèmes de programmation à deux niveaux en nombres entiers n'ont pas encore reçu l'attention qu'ils devraient avoir du point de vue des applications potentielles. Il existe, peu d'auteurs qui se sont intéressés aux problèmes à deux niveaux en nombres entiers Dempe 2005 ; Dempe 2018 ([52, 51]) où le cas discret s'applique uniquement au second niveau, pour cette classe de problèmes, les méthodes de résolution utilisées sont basées sur les méthodes des coupes où cette méthode est appliquée pour le second niveau en nombres entiers dans le but de transformer le problème discret à deux niveaux en un problème linéaire à deux niveaux. En général, cette méthode est utilisée dans le cas où uniquement le second niveau est en nombres entiers. Néanmoins, la méthode de coupe utilisée seule ne donne pas de meilleurs résultats, cette méthode doit être combinée avec les algorithmes de Branch and Bound pour des meilleurs résultats [58]. Dans le chapitre 5, nous résolvons le problème non linéaire à deux niveaux en nombres entiers où la combinaison des deux méthodes (méthode de coupe et la méthode de Branch and Bound) est utilisée et bien détaillée.

3.10 La complexité de la programmation à deux niveaux

On dit qu'une classe de problème est NP-difficile (resp. fortement NP-difficile) dans le cas où il ne peut exister un algorithme de résolution pour un problème de cette classe en temps borné par une fonction polynomiale de la taille du problème (resp. fonction polynomiale de la taille et de la valeur maximale des données du problème), que si tous problèmes de cette classe peuvent être résolus en temps polynomial.

Tout comme Jeroslow R.G, Ben-Ayed O. et Blair C. ([78], [23]) ont établi que le problème de programmation à deux niveaux dans le cas linéaire est un problème NP-difficile, pour le démontrer Ben-Ayed O. et Blair C. [23] utilisent le fait que le problème du sac à dos en variables binaires (dont la résolution est NP-difficile) peut être reformulé comme un problème de programmation linéaire à deux niveaux.

Hansen P., Jaumard B. et Savard G. [73] ont montré que ce résultat pouvait être renforcé en considérant le problème max-min linéaire qui est un cas particulier du problème (*PBN*). Vicente L., Savard G. et Jùdice J. [136] ont montré que établir l'optimalité stricte ou locale d'une solution est également NP-difficile. Audet C. et al. [12] ont ensuite montré que les problèmes bien connus en optimisation combinatoire, comme le modèle général de programmation quadratique concave à contraintes linéaire ou alors des instances de programmation à variables mixtes peuvent être réduits de façon polynomiale en problème de programmation linéaire à deux niveaux. Frangioni (1995) a montré que tout problème

d'optimisation mixte peut être formulé comme un problème de programmation à deux niveaux (*PBN*). On a donc les résultats suivants :

Propriété 3.10.1 (*Corrolaire 3.2, Hansen P., Jaumard B. et Savard G.[73]*). *Le problème de programmation à deux niveaux (*PBN*) est fortement NP-difficile. Pour résoudre un problème de programmation à deux niveaux (*PBN*) non linéaire, Savard G. et Gauvin J. [111] ont proposé des conditions nécessaires d'optimalité et ont montré qu'une direction de descente de plus forte pente est obtenu en résolvant un programme quadratique à deux niveaux.*

3.11 Conclusion

Dans ce chapitre les principaux concepts et définitions de l'optimisation à deux niveaux tels que la modélisation d'un problème à deux niveaux linéaire et non-linéaire (continu et discret), la notion de la solution optimale ainsi que le domaine réalisable du premier niveau et du second niveau. Des exemples ont été donnés pour mieux comprendre le cas linéaire et le cas non-linéaire de cette classe de problèmes. Dans cette classe de problèmes, on parle d'une coopération ou non entre le leader (le premier niveau) et le suiveur (le second niveau) d'où la notion du cas optimiste et le cas pessimiste.

Chapitre 4

Programmation fractionnaire quadratique/quadratique à deux niveaux

Ce chapitre consiste à élaborer un algorithme pour résoudre un problème fractionnaire quadratique/quadratique à deux niveaux Maachou N. et Moulaï M. [88]. La méthode proposée est basée sur les conditions de Karush-Kuhn-Tucker, pour le quel le problème du second niveau est remplacé par ses conditions d'optimalité de (KKT). Il en résulte un programme mathématique non linéaire à un seul niveau (*PNL*), où la résolution de ce problème non linéaire s'appuie essentiellement sur des résultats trouvés dans notre travail Maachou N. et Moulaï M. [89]. Afin, de détailler nos travaux, nous avons subdivisé ce chapitre en deux parties : La première concerne la résolution d'un problème fractionnaire quadratique discret, alors que dans la seconde partie, nous exposons la méthode de résolution d'un problème fractionnaire quadratique/quadratique à deux niveaux.

4.1 Partie1 : Une méthode exacte pour un problème fractionnaire quadratique discret

4.1.1 Introduction

Les problèmes de programmation fractionnaires ont suscité un grand intérêt depuis qu'ils se posent dans de nombreux domaines comme planification agricole, l'analyse financière d'une entreprise, la théorie de la localisation, le problème de la budgétisation

du capital, le problème de la sélection du portefeuille, le problème de la coupe des stocks, le problème des processus stochastiques. Des algorithmes ont été présentés par de nombreux auteurs [3, 129].

4.1.2 Notations et définitions

Soit S l'ensemble des vecteurs $x \in \mathbb{R}^n$ satisfaisant les contraintes $x \geq 0$, x entier et $Ax \leq b$. où A est une matrice entière $m \times n$ et b un vecteur de $\mathbb{R}^m$. Soit C, D des vecteurs de $\mathbb{R}^n$, E, F sont des matrices symétriques réelles et α, β deux éléments de $\mathbb{R}$.

Le problème de programmation fractionnaire quadratique en nombres entiers (P) , à étudier, peut être énoncé mathématiquement comme suit :

$$(P) \left\{ \begin{array}{l} \text{Min } f(x) = \frac{C^T x + x^T E x + \alpha}{D^T x + x^T F x + \beta} \\ \text{s.c} \\ x \in S \end{array} \right. \quad (4.1)$$

$D^T x + x^T F x + \beta > 0$ pour tout $x \in \tilde{S} = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$ où l'ensemble $\tilde{S}$ est non vide et borné.

Pour trouver une solution entière optimale du problème (P) , les solutions entières réalisables sont classées par ordre croissant des valeurs de la fonction objectif. Pour obtenir diverses solutions entières réalisables, un problème de programmation fractionnaire linéaire entière (P_1) est construit.

$$(P_1) \left\{ \begin{array}{l} \text{Min } g(x) = \frac{U^T x + \alpha}{V^T x + \beta} \\ \text{s.t} \\ x \in S \end{array} \right. \quad (4.2)$$

où

$U_j = j^{\text{ème}}$ composante de $U \in \mathbb{R}^n$
 $= \min_{x \in \tilde{S}} (C_j + x^T E_j), j = 1, \dots, n$, C_j étant la $j^{\text{ème}}$ colonne de C et E_j étant la $j^{\text{ème}}$ colonne de E .

$V_j = j^{\text{ème}}$ composante de $V \in \mathbb{R}^n$
 $= \max_{x \in \tilde{S}} (D_j + x^T F_j), j = 1, \dots, n$, D_j étant la $j^{\text{ème}}$ colonne de D et F_j étant la $j^{\text{ème}}$ colonne de F .

4.1.3 Quelques résultats

Proposition 4.1.1

$$g(x) \leq f(x), \quad \forall x \in \tilde{S}. \quad (4.3)$$

Preuve 4.1.1 Par définition de U_j et V_j , nous avons :

$$\forall x \in \tilde{S}, \quad U_j \leq C_j + x^T E_j \text{ et } V_j \geq D_j + x^T F_j, \quad j = 1, \dots, n.$$

$$\text{As } x \geq 0, \quad \forall x \in \tilde{S} \quad U^T x \leq (C^T + x^T E)x \text{ et } V^T x \geq (D^T + x^T F)x$$

Il en résulte clairement que $\forall x \in \tilde{S}, \quad g(x) \leq f(x)$.

Rappelons que le problème fractionnaire linéaire en nombre entier (P_1) peut être résolu par une méthode de Branch and Bound basée sur le calcul de pénalités [4]. Considérons le problème (P_2) sans les contraintes d'intégralité

$$(P_2) \left\{ \begin{array}{l} \text{Min } g(x) = \frac{U^T x + \alpha}{V^T x + \beta} \\ \text{s.t} \\ x \in \tilde{S} \end{array} \right. \quad (4.4)$$

en introduisant des variables d'écarts et en le résolvant par la méthode de Cambini et Martein [35], nous trouvons la solution continue optimale. Soit le tableau de simplexe optimal donné par (P'_2) :

$$(P'_2) \left\{ \begin{array}{l} \text{Min } g(x) = \frac{\bar{\alpha} + \sum_{j \in I_N}^{p_j} x_j}{\bar{\beta} + \sum_{j \in I_N}^{q_j} x_j} \\ \text{s.t} \\ x_i + \sum_{j \in I_N} \bar{a}_{ij} x_j = e_i \quad i \in I_B \\ x_j \geq 0, \quad j \in I_N \end{array} \right. \quad (4.5)$$

où I_N est l'ensemble des indices des variables hors base, $x_i, i \in I_B$ est la variable de base, $\bar{\alpha}$ et $\bar{\beta}$ sont les coûts réduits dans le tableau de simplexe et $\bar{\alpha}/\bar{\beta}$ est la valeur de la fonction objectif. La solution optimale de base du problème (P_2) est donnée par

$$x_i = e_i, \quad i \in I_B \text{ sinon } x_j = 0 \quad j \in I_N.$$

Si e_i est entier pour tout $i \in I_B$, alors la solution optimale entière du problème (P_2) est trouvée. Si la solution trouvée est non entière, soit e_{kt} la valeur non entière de x_{kt} pour un certain $kt \in I_B$. Nous notons par plus grand entier inférieur à e_{kt} par $\lfloor e_{kt} \rfloor$ et le plus petit nombre entier supérieur à e_{kt} par $\lceil e_{kt} \rceil$. Puisque, x_{kt} doit être entier, $x_{kt} \leq \lfloor e_{kt} \rfloor$ ou $x_{kt} \geq \lceil e_{kt} \rceil$.

Considérons le premier $x_{kt} \leq \lfloor e_{kt} \rfloor$ ce qui donne lieu à la contrainte $x_{kt} + s = \lfloor e_{kt} \rfloor$ but $x_{kt} + \sum_{j \in I_N} \bar{a}_{ktj} x_j = e_{kt}$ à partir du tableau de simplexe. Alors

$$- \sum_{j \in I_N} \bar{a}_{ktj} x_j + s = \lfloor e_{kt} \rfloor - e_{kt}$$

Ainsi, nous avons :

$$- \sum_{j \in I_N} \bar{a}_{ktj} x_j \leq \lfloor e_{kt} \rfloor - e_{kt}. \quad (4.6)$$

Il est évident que $\lfloor e_{kt} \rfloor - e_{kt}$ est négative et la solution optimale du problème (P_1) donnée ci-dessous ne satisfait pas la contrainte (4.6). En augmentant cette contrainte au problème (P_1) , on obtient une des branches.

De même, correspondant à $x_{kt} \geq \lceil e_{kt} \rceil$, on obtient la contrainte

$$-x_{kt} + s = -\lceil e_{kt} \rceil$$

Alors

$$\sum_{j \in I_N} \bar{a}_{ktj} x_j \leq e_{kt} - \lceil e_{kt} \rceil < 0. \quad (4.7)$$

En introduisant ces contraintes dans le problème (P_1) , nous obtenons l'autre branche. Pour choisir une branche qui doit être ajoutée au tableau de simplexe optimal (P'_2) , on calcule les pénalités π_r and π'_r des contraintes $x_{kt} \leq \lfloor e_{kt} \rfloor$ et $x_{kt} \geq \lceil e_{kt} \rceil$, respectivement donnée par :

$$\pi_r = \frac{e\Delta_r}{\bar{\beta}(\bar{\beta} + \frac{e\bar{q}_r}{\bar{a}_{kt,r}})}$$

et

$$\pi'_r = \frac{(1-e)\Delta'_r}{\bar{\beta}(\bar{\beta} + \frac{(1-e)\bar{q}_r}{\bar{a}_{kt,r'}})}$$

où

$$\Delta_r = \min \left\{ \frac{\bar{\gamma}_j}{-\bar{a}_{kt,j}} \mid \bar{a}_{kt,j} > 0 \right\}, \quad \Delta'_r = \min \left\{ \frac{\bar{\gamma}_j}{\bar{a}_{kt,j}} \mid \bar{a}_{kt,j} < 0 \right\}$$

et $e = e_{kt} - \lfloor e_{kt} \rfloor$.

γ_j est la $j^{\text{ème}}$ composante du gradient réduit de g at x , $\bar{\gamma}_j = \bar{\beta} \times \bar{p}_j - \bar{\alpha} \times \bar{q}_j$, $j \in I_N$.

La branche correspondante à la plus petite pénalité est ajoutée au problème (P'_2) .

Proposition 4.1.2 *Si pour un $k \geq 1$, $g_k \geq \min \{f(x) \mid x \in T^k\} = f(\hat{x})$, alors $\hat{x}$ la solution optimale du problème (P) .*

Preuve 4.1.2 *Nous avons*

$$\forall x \in T^k \quad f(\hat{x}) \leq f(x) \quad (4.8)$$

D'autre part, d'après la proposition 4.1.1 et notre hypothèse, il en résulte que :

$$\forall x \in S/T^k \quad f(x) \geq g(x) \geq g_{k+1} > g_k \geq f(\hat{x}). \quad (4.9)$$

Les conditions (4.8) et (4.9) implique $\hat{x}$ est la solution optimale du problème (P) .

La proposition suivante montre que lorsque l'hypothèse de la proposition 4.1.2 n'est pas satisfaite, on a une information sur la valeur minimale du problème initial (P) .

Proposition 4.1.3 *Si $g_k < \min \{f(x) \mid x \in T^k\}$, alors $g_k < f_1 \leq \min \{f(x) \mid x \in T^{k+1}\}$.*

Preuve 4.1.3 *La deuxième inégalité est évidente. En premier, on note comme dans la proposition 4.1.2 que*

$$\forall x \in S/T^k \quad f(x) \geq g(x) \geq g_{k+1} > g_k \quad (4.10)$$

Cependant

$$\min \{f(x) : x \in S/T^k\} > g_k \quad (4.11)$$

Par hypothèse,

$$\min \{f(x) : x \in T^k\} > g_k \quad (4.12)$$

Alors les conditions (4.11) et (4.12) donnent $f_1 > g_k$.

4.1.4 Notations

Δ_i = L'ensemble des $i^{\text{ème}}$ meilleures solutions réalisables de (P_1) ;

Il clair que :

Δ_1 = l'ensemble des solutions optimales du problème (P_1) , g_1 la valeur optimale correspondante à Δ_1 et

Δ_2 = l'ensemble des $2^{\text{ème}}$ solutions du problème (P_1) , g_2 la valeur optimale correspondante à Δ_2 . Par conséquent, $g_1 < g_2$. Evidement pour $i = k$ nous avons $g_k < g_{k+1}$ et l'ensemble réalisable S étant fini. Introduisons les notations $T^k = \Delta_1 \cup \Delta_2 \cup \dots \cup \Delta_k$.

Algorithme

Etape 0 Déterminer $C_j = \underset{x \in \tilde{S}}{\text{Min}} (C_j + x^T E_j)$ et $V_j = \underset{x \in \tilde{S}}{\text{Max}} (D_j + x^T F_j)$, $j = 1 \dots n$.

Construire le problème relaxé de programmation fractionnaire linéaire en nombres entiers (P_1) et aller à l'étape 1.

Etape 1 Résoudre le probleme (P_2) .

- si une telle solution n'existe pas, stop. Soit $\sup_{x \in S} g = +\infty$
- Sinon, $k = 1, l = 1$ et aller à l'étape 2.

Etape 2 Soit x^1 la solution optimale continue du problème (P_2) .

- Si x^1 est entière, déterminer Δ_1 et calculer $g(x^1) = g_1$, aller à l'étape 5.
- Sinon pour $k = 1, l = 1$, soit x_{kt} pour un $kt \in I_B$, la composante non entière de x^1 avec la valeur correspondante e_{kt} . $\pi_l = 0$ aller à l'étape 3.

Etape 3 Calculer π_{2k-1} et π_{2k} . Soit $\pi_{2k-1} = \pi_{2k-1} + \pi_l, \pi_{2k} = \pi_{2k} + \pi_l$ et $\pi_l = +\infty$. Calculer $\pi_l = \min_{1 \leq j \leq 2k} \{\pi_j\}$. Ajouter la contrainte au tableau optimal du simplexe, le résoudre et aller à l'étape 4.

Etape 4 si x^l est entière, déterminer Δ_l et calculer $g(x^l) = g_l$, aller à l'étape 7. Sinon, les problèmes augmentés n'admettent pas de solutions, stop. Soit x_{kt}^l la composante non entière de x^l avec la valeur correspondante e_{kt}^l . $k = k + 1$, aller à l'étape 3.

Etape 5 Test

- Si $g_1 = \min \{f(x) \mid x \in T^1\} = f(\hat{x}_1)$. Alors $\hat{x}_1$ est la solution optimale entière du problème (P) .
- Si $g_1 < \min \{f(x) \mid x \in T^1\}$, $l = 2$, aller à l'étape 6.

Etape 6 Trouver la meilleure suivante solution du problème (P_2) , aller à l'étape 4.

Etape 7 Test

- Si $g_l \geq \min \{f(x) \mid x \in T^l\} = f(\hat{x}_1)$, alors $\hat{x}_1$ est la solution optimale entière du problème (P) .

- Si $g_l < \min \{f(x) \mid x \in T^l\}$, $l = l + 1$, aller à l'étape 6.

Exemple 4.1.1 *Considérons le problème de programmation fractionnaire quadratique en nombres entiers (P) :*

$$(P) \left\{ \begin{array}{l} \text{Min} \quad f(x) = \frac{-5x_1 - 8x_2 - x_1^2 - 2x_1x_2 - 3}{2x_1 + x_1^2 + 2} \\ \text{s.c} \\ 2x_1 + 3x_2 \leq 6 \\ 3x_1 + 2x_2 \leq 5 \\ x_1 \geq 0, x_2 \geq 0, \text{ entiers} \end{array} \right.$$

Etape 0 : *Le problème fractionnaire linéaire relaxé (P_1) est construit comme suit :*

$$(P_1) \left\{ \begin{array}{l} \text{Min} \quad f(x) = \frac{-7x_1 - 9x_2 - 3}{3x_1 + 4x_2 + 2} \\ \text{s.c} \\ 2x_1 + 3x_2 \leq 6 \\ 3x_1 + 2x_2 \leq 5 \\ x_1 \geq 0, x_2 \geq 0, \text{ entiers} \end{array} \right.$$

Pour résoudre le problème (P_1), un problème fractionnaire linéaire relaxé (P_2) est déterminé comme suit :

$$(P_2) \left\{ \begin{array}{l} \text{Min} \quad f(x) = \frac{-7x_1 - 9x_2 - 3}{3x_1 + 4x_2 + 2} \\ \text{s.c} \\ 2x_1 + 3x_2 \leq 6 \\ 3x_1 + 2x_2 \leq 5 \\ x_1 \geq 0, x_2 \geq 0 \end{array} \right.$$

Etape 1 : *La solution optimale est $x_1 = 3/5$, $x_2 = 8/5$ et $x_j = 0$.*

Etape 2 : *Puisque x_1 et x_2 ne sont pas entières, aller à l'étape 3.*

Etape 3 : *Calculer les pénalités π_1 and π_2 des contraintes $x_1 \leq [3/5]$ et $x_1 \geq [3/5]$, respectivement, nous obtenons $\pi_1 = 3/170$ et $\pi_2 = 1/153$. Ensuite, nous sélectionnons la branche dont la pénalité est la plus petite et augmentons la contrainte respective au tableau optimal de simplexe (P'_2).*

Etape 4 : *La solution obtenue est $x_1 = 1$, $x_2 = 1$, $x_3 = 1$ et $x_j = 0$. $\Delta_1 = \{(1, 1)\}$ et $g_1 = 19/9$.*

Etape 5 : $\min \{f(x) \mid x \in T^1\} = 19/9$. Alors $g_1 = \min \{f(x) \mid x \in T^1\} = f(\hat{x}_1)$. Alors $\hat{x}_1$ est la solution optimale du problème (P).

4.1.5 Conclusion

Dans cette partie, nous avons déterminé un algorithme pour résoudre une classe des problèmes non linéaire, le problème principal a été reformulé en un problème fractionnaire linéaire qui a été résolu par une méthode exacte basée sur la méthode Branch and Bound améliorée, car la branche suivie est basée sur la pénalité minimale pour un calcul plus rapide.

4.2 Partie2 : Résolution d'un porogramme fractionnaire quadratique/quadratique à deux niveaux

4.2.1 Introduction

la programmation fractionnaire et la programmation quadratique lorsqu'il n'existe qu'un seul niveau de décision ont fait l'objet d'une attention particulière dans la littérature Calvete et al., Gupta R., Puri M. C.[33, 68]. Il convient de mentionner que les fonctions objectifs qui sont des fractions apparaissent fréquemment, par exemple, lorsqu'une mesure d'efficacité d'un système doit être optimisée ou lorsqu'on aborde un problème de programmation stochastique. Les problèmes quadratiques apparaissent directement dans des applications telles que la régression des moindres carrés avec des limites ou des contraintes linéaires, l'optimisation de portefeuille ou l'ajustement robuste de données. Les problèmes fractionnaires à deux niveaux ont été étudiés par Calvete et Galé [33]. Les problèmes quadratiques à deux niveaux ont été traités par Vicente L., Savard G. and Jùdice J. [136]. Calvete et Galé [33] ont étudié les conditions d'optimalité pour le problème de programmation linéaire fractionnaire/ quadratique à deux niveaux en se basant sur les conditions de Karush - Kuhn - Tucker et la théorie de la dualité. Pour une plus grande classe de problèmes de programmation fractionnaire/quadratique à deux niveaux, nous nous intéressons dans cette partie aux problèmes de programmation fractionnaire quadratique/ quadratique à deux niveaux (BQFQP), où la fonction objectif de premier niveau est quadratique fractionnaire Gotoh J.Y. et Konno H. [71] et la fonction objectif de second niveau est quadratique. Les variables associées aux deux problèmes (premier et second niveau) sont liées par des contraintes linéaires. Pour résoudre le problème, nous construisons le problème linéaire fractionnaire/quadratique à deux niveaux et utilisons les conditions d'optimalité de Kuhn-Tucker avec les conditions de dualité, en appliquant certaines propositions, nous avons pu déterminer une solution optimale pour le problème principal (BQFQP).

4.2.2 Notations et définitions

Soit S l'ensemble des solutions réalisables $(x, y) : x \in \mathcal{R}^{n_1}, y \in \mathcal{R}^{n_2}$ satisfaisant les contraintes : $\begin{cases} Ax + By \leq r \\ x \geq 0, y \geq 0 \end{cases}$ où A une matrice de dimension $m \times n_1$, B une matrice de dimension $m \times n_2$ et r un vecteur de $\mathcal{R}^m$. Soit C, H, P des vecteurs de dimension $\mathcal{R}^{n_1}$, D, M, R des vecteurs dans $\mathcal{R}^{n_2}$; E, F, L, G, Q sont des matrices symétriques dans $\mathcal{R}$ et α, β deux éléments dans $\mathcal{R}$. Le problème de programmation fractionnaire quadratique/ quadratique à deux niveaux (BQFQP) [102], que l'on souhaite étudier peut être énoncé mathématiquement comme suit :

$$(BQFQP) \begin{cases} \max_x F(x, y) = \frac{C^T x + x^T E x + D^T y + y^T F y + \alpha}{H^T x + x^T L x + M y + y^T G y + \beta} \\ \text{où } y \text{ résout} \\ \begin{cases} \max f(x, y) = P x + R y + (x, y)^T Q(x, y) \\ \text{s.c. } (x, y) \in S \\ (x, y) \geq 0 \end{cases} \end{cases} \quad (4.13)$$

Définition 4.2.1 .

L'ensemble réalisable (x, y) du problème (BQFQP) est l'ensemble S définie par :

$$S = \{(x, y) : Ax + By \leq r, x, y \geq 0\} .$$

Définition 4.2.2 .

La projection de S sur l'espace de décision du premier niveau est :

$$S(X) = \{x \geq 0 : \exists y \in \mathcal{R}^{n_2} : (x, y) \in S\}.$$

Afin de s'assurer que le problème 4.13 est bien posé, nous donnons les hypothèses suivantes :

- S est non vide et borné.
- $C^T x + x^T E x + D^T y + y^T F y + \alpha \geq 0$, pour tout $(x, y) \in S$
- $H^T x + x^T L x + M y + y^T G y + \beta > 0$, pour tout $(x, y) \in S$

Dans le problème 4.13, soit

$$Q = \begin{pmatrix} Q_3 & Q_2^T \\ Q_2 & Q_1 \end{pmatrix} \text{ où } Q_1, Q_2 \text{ et } Q_3 \text{ sont des matrices de dimensions conformes.}$$

Alors $f(x, y)$ est transformé en

$$f(x, y) = P^T x + x^T Q_3 x + y^T Q_1 y + (R + 2Q_2 x)^T y$$

Pour x fixé, le problème du second niveau est équivalent à :

$$(P_x) \left\{ \begin{array}{l} \max_y f(x, y) = y^T Q_1 y + (R + 2Q_2 x)^T y \\ \text{s.t} \\ By \leq r - Ax \\ (x, y) \geq 0 \end{array} \right. \quad (4.14)$$

Pour tout $x \in S(X)$ ($S(X)$ la projection de S sur $\mathcal{R}^{n_1}$), nous supposons que Q_1 est définie positive de sorte qu'il y aura une solution optimale unique au problème de second niveau. Autrement dit, $M(x)$ est un singleton pour tous les $S(X)$, où $M(x)$ est l'ensemble des solutions optimales de (P_x) . Pour chaque $x \in S(X)$, $S(x)$ (l'ensemble réalisable du second niveau) est également un polyèdre compact non vide. Enfin, la région inductible, ou la région réalisable du décideur de premier niveau, appelée région inductible IR , est implicitement définie par le problème d'optimisation de second niveau :

$$IR = \{(x, y^*) : x \in S(X), y^* \in M(x)\}$$

Cependant $(BQFQP)$ est équivalent à :

$$(BQFQP') \left\{ \begin{array}{l} \max_x F(x, y) = \frac{C^T x + x^T E x + D^T y + y^T F y + \alpha}{H^T x + x^T L x + M y + y^T G y + \beta} \\ \text{où } y \text{ résout} \\ \left\{ \begin{array}{l} \max_y f(x, y) = y^T Q_1 y + (R + 2Q_2 x)^T y \\ \text{s.c} \quad By \leq r - Ax \\ (x, y) \geq 0 \end{array} \right. \end{array} \right. \quad (4.15)$$

Définition 4.2.3 .

Un point (x^*, y^*) est dit optimal pour $(BQFQP')$ si $(x^*, y^*) \in IR$ et $F(x^*, y^*) \geq F(x, y)$ pour tout $(x, y) \in IR$

$(BQFQP')$ peut s'écrire :

$$(BQFQP') \left\{ \begin{array}{l} \max_x F(x, y) = \frac{C^T x + x^T E x + D^T y + y^T F y + \alpha}{H^T x + x^T L x + M y + y^T G y + \beta} \\ \text{s.c} \\ (x, y) \in IR \end{array} \right. \quad (4.16)$$

Pour trouver une solution optimale au problème $(BQFQP')$, un problème de programmation fractionnaire linéaire /quadratique à deux niveaux $(BLFQP)$ est construit :

$$(BLFQP) \left\{ \begin{array}{l} \max \quad g(x, y) = \frac{a^T x + b^T y + \alpha}{c^T x + d^T y + \beta} \\ \text{s.c} \\ (x, y) \in IR \end{array} \right. \quad (4.17)$$

où

$$a = \max_{x \in IR} (C + x^T E) \text{ et } b = \max_{x \in IR} (D + y^T F).$$

$$c = \min_{x \in IR} (H + x^T L) \text{ et } d = \min_{x \in IR} (M + y^T G).$$

4.2.3 Quelques résultats

Proposition 4.2.1

$$g(x, y) \geq F(x, y), \text{ pour tout } (x, y) \in IR. \quad (4.18)$$

Preuve 4.2.1 Par définition de a, b, c et d , nous avons pour tout $(x, y) \in IR$:

$$\begin{array}{lcl} a \geq C + x^T E & & c \leq H + x^T L \\ b \geq D + y^T F & \text{et} & d \leq M + y^T G \end{array} .$$

Comme $(x, y) \geq 0$,

$$a^T x \geq (C^T + x^T E)x \quad \text{et} \quad b^T y \geq (D^T + y^T F)y, \quad \text{pour tout } (x, y) \in IR$$

Alors

$$a^T x + b^T y \geq (C^T + x^T E)x + (D^T + y^T F)y, \quad \forall (x, y) \in IR.$$

De la même manière : $c^T x + d^T y \leq (H_j + x^T L_j)x + (M_j + y^T G_j)y$, pour tout $(x, y) \in IR$.

Il s'ensuit clairement que $g(x, y) \geq F(x, y)$, pour tout $(x, y) \in IR$.

Rappelons que le problème de programmation fractionnaire linéaire/quadratique à deux niveaux $(BLFQP)$ peut être résolu en appliquant la condition d'optimalité basée sur la dualité et les conditions de Karush-Kuhn-Tucker (H. I.Calvete et C. Galé.[33]).

Considérons le problème (*BLFQP*) :

$$(BLFQP) \left\{ \begin{array}{l} \max_{(x,y)} g(x,y) = \frac{a^T x + b^T y + \alpha}{c^T x + d^T y + \beta} \\ \text{s.c} \\ (x,y) \in IR \end{array} \right. \quad (4.19)$$

En appliquant les conditions nécessaires et suffisantes de Karush-Kuhn-Tucker, il existe $\omega \in \mathcal{R}^m$, tel que (x, y, ω) satisfait :

$$Ax + By \leq r \quad (4.20)$$

$$\omega^T(Ax + By - r) = 0 \quad (4.21)$$

$$2Q_2x + 2Q_1y + B^T\omega = -R^T \quad (4.22)$$

$$\omega \geq 0 \quad (4.23)$$

De même, si (x, y, ω) satisfait 4.20 - 4.23, alors $(x, y) \in IR$.

Ainsi, le problème (*BLFQP*) donné devient un problème de programmation fractionnaire linéaire (*LFP*) donné par :

$$(LFP) \left\{ \begin{array}{l} \max_{(x,y,\omega)} g(x,y) = \frac{a^T x + b^T y + \alpha}{c^T x + d^T y + \beta} \\ \text{s.c} \\ Ax + By \leq r \\ 2Q_2x + 2Q_1y + B^T\omega = -R^T \\ x, y, \omega \geq 0 \end{array} \right. \quad (4.24)$$

avec la condition

$$\omega^T(Ax + By - r) = 0 \quad (4.25)$$

Théorème 4.2.1 (x, y) est une solution optimale au problème (*BLFQP*) si et seulement s'il existe $\omega^* \in \mathcal{R}^m$ tel que (x^*, y^*, ω^*) est une solution optimale au problème de programmation non linéaire à un seul niveau (*NLP*) :

$$(NLP) \left\{ \begin{array}{l} \max_{(x,y,\omega)} g(x,y) = \frac{a^T x + b^T y + \alpha}{c^T x + d^T y + \beta} \\ \text{s.c} \\ 4.20 - 4.23 \end{array} \right. \quad (4.26)$$

En utilisant la théorie de la dualité, nous concluons l'existence de $\omega^* \in \mathcal{R}^m$ tel que (x^*, y^*, ω^*) est une solution optimale du problème (NLP) (H. I.Calvete et C. Galé.[33]). Par conséquent, d'après le théorème de H. I.Calvete et C. Galé. ([33]), nous obtenons que (x^*, y^*) est une solution optimale du problème $(BLFQP)$ ([33]).

Le programme non linéaire (NLP) peut être réécrite comme suit :

$$(LFP) \begin{cases} \max_{(x,y,\omega)} g(x,y) = \frac{a^T x + b^T y + \alpha}{c^T x + d^T y + \beta} \\ \text{s.c} \quad Ax + By \leq r \\ \quad 2Q_2 x + 2Q_1 y + B^T \omega = -R^T \\ \quad x, y, \omega \geq 0 \end{cases} \quad (4.27)$$

avec les conditions

$$\omega^T (Ax + By - r) = 0 \quad (4.28)$$

Ainsi, pour déterminer la solution optimale de $(BLFQP)$, nous résolvons le problème fractionnaire linéaire (LFP) (en utilisant notre travail [89](Partie 1)) avec la condition 4.28. Comme la fonction objectif est fractionnaire linéaire, elle est à la fois pseudoconcave et pseudoconvexe et donc, sa solution optimale sera une solution réalisable de base. Nous nous basons sur la recherche d'une solution réalisable de base qui satisfait la condition suivante :

$$\omega^T (Ax + By - r) = 0 \quad (4.29)$$

Rappelons que, pour obtenir les solutions réalisables de base, le programme fractionnaire linéaire (LFP) est transformé en un programme linéaire (LP) par la méthode de Charnes et Cooper [38]. La méthode de Murty [100] est appliquée à un programme linéaire (LP) pour déterminer toutes les solutions de base réalisables. Cette méthode est basée sur la génération d'un vecteur de coefficient objectif à chaque itération, de telle sorte que l'optimisation de cette fonction objectif soumise aux contraintes spécifiées par des techniques de programmation linéaire mène à une nouvelle solution réalisable de base, jusqu'à ce que toutes les solutions réalisables de base soient obtenues.

Notations

Δ_i = L'ensemble de la $i^{\text{ème}}$ meilleure solution réalisable de $(BLFQP)$,
 g_i la valeur correspondante à Δ_i

Il est clair que :

Δ_1 = L'ensemble des solutions optimales de $(BLFQP)$,
 g_1 la valeur optimale correspondante à Δ_1 et

$\Delta_2 =$ L'ensemble de la 2^{ème} meilleure solution réalisable de $(BLFQP)$, g_2 la valeur correspondante à Δ_2 .

Il en résulte que $g_1 > g_2$.

Évidemment, pour $i = k$, on a $g_k > g_{k+1}$. Introduisons les notations

$$T^k = \Delta_1 \cup \Delta_2 \cup \dots \cup \Delta_k,$$

$$\max \{F(x, y) \mid (x, y) \in IR\} = F(x^*, y^*) = F_1$$

Proposition 4.2.2 *Si pour un certain $k \geq 1$, $g_k \leq \max \{F(x, y) \mid (x, y) \in T^k\} = F(\hat{x}, \hat{y})$, alors $(\hat{x}, \hat{y})$ est la solution optimale de $(BQFQP')$.*

Preuve 4.2.2 *Nous avons*

$$\forall (x, y) \in T^k \quad F(\hat{x}, \hat{y}) \geq F(x, y) \quad (4.30)$$

D'autre part, d'après la proposition 4.2.1 et notre hypothèse, il suit que :

$$\forall (x, y) \in IR/T^k \quad F(x, y) \leq g(x, y) \leq g_{k+1} < g_k \leq F(\hat{x}, \hat{y}). \quad (4.31)$$

Les conditions (4.30) et (4.31) impliquent que $(\hat{x}, \hat{y})$ est la solution optimale du problème $(BQFQP')$.

La proposition suivante montre que lorsque l'hypothèse de la proposition 4.2.2 n'est pas satisfaite, on a une indication sur la valeur maximale du problème initial $(BQFQP')$.

Proposition 4.2.3 *Si $g_k > \max \{F(x, y) \mid (x, y) \in T^k\}$, alors $g_k > F_1 \geq \max \{F(x, y) \mid (x, y) \in T^{k+1}\}$.*

Preuve 4.2.3 *La deuxième inégalité est évidente. Pour la première partie, on note comme dans la proposition 4.2.2 que*

$$\text{pour tout } (x, y) \in IR/T^k, \quad F(x, y) \leq g(x, y) \leq g_{k+1} < g_k \quad (4.32)$$

Cependant

$$\max \{F(x, y) : (x, y) \in IR/T^k\} < g_k \quad (4.33)$$

Par hypothèse,

$$\max \{F(x, y) : (x, y) \in T^k\} < g_k \quad (4.34)$$

Alors les conditions (4.33) et (4.34) implique que $F_1 < g_k$.

Algorithme

Etape 0 Résolution du problème linéaire-quadratique à deux niveaux (en utilisant les conditions nécessaires et suffisantes de Karush-Kuhn-Tucker [1]) :

$$\begin{cases} a = \max_{(x,y) \in IR} (C + x^T E), b = \max_{(x,y) \in IR} (D + y^T F), \\ c = \min_{(x,y) \in IR} (H + x^T L), d = \min_{(x,y) \in IR} (M + y^T G). \end{cases} \quad (4.35)$$

et construire le problème fractionnaire linéaire /quadratique à deux niveaux ($BLFQP$), aller à l'étape 1.

Etape 1 Résoudre le problème de programmation linéaire fractionnaire (LFP) .

$$(LFP) \begin{cases} \max_{(x,y,\omega)} g(x,y) = \frac{a^T x + b^T y + \alpha}{c^T x + d^T y + \beta} \\ \text{s.c} \quad Ax + By \leq r \\ 2Q_2x + 2Q_1y + B^T \omega = -R^T \\ x, y, \omega \geq 0 \end{cases} \quad (4.36)$$

Soit (x^*, y^*, ω^*) la solution optimale. $l = 1$ et aller à l'étape 2.

Etape 2 . Si la solution obtenue est telle que : $\omega^*(Ax^* + By^* - r) = 0$, alors $(x^*, y^*) = (x^l, y^l)$ est la solution optimale du problème ($BLFQP$), soit $\Delta_l = \{(x^l, y^l)\}$, calculer $g(x^l, y^l) = g_l$. Aller à l'étape 4. Sinon, aller à l'étape 3.

Etape 3 Trouver la meilleure solution de base réalisable suivante au problème (LFP) en utilisant la méthode de Murty [100] et passer à l'étape 2.

Etape 4 Test

- Si $g_l \leq \max \{F(x, y) \mid (x, y) \in T^l\} = F(\hat{x}, \hat{y})$. Alors $(\hat{x}, \hat{y})$ est la solution optimale du problème ($BQFQP$).
- Si $g_l > \max \{F(x, y) \mid (x, y) \in T^l\}$, $l = l + 1$ et aller à l'étape 3.

Comme nous l'avons vu précédemment, l'algorithme converge en un nombre fini d'étapes puisque le nombre de solutions réalisables de base dans S est fini et qu'une fois qu'un point est testé pour l'optimalité de ($BQFQP$), il est supprimé de l'ensemble et ne réapparaît donc pas dans le processus.

4.2.4 conclusion

Dans cette partie, nous avons résolu le problème fractionnaire quadratique linéaire /quadratique à deux niveaux. Notre procédure est basée sur le problème de programmation linéaire fractionnaire/quadratique à deux niveaux et sur les conditions de Karush-Kuhn-Tucker. Ces conditions ont été utilisées dans la littérature pour résoudre un grand

nombre de problèmes de programmation non linéaire. Le problème de programmation linéaire fractionnaire/quadratique à deux niveaux lié au problème principal est formulé et ses solutions réalisables de base sont scannées de manière systématique jusqu'à ce qu'une solution optimale du problème soit obtenue.

Chapitre 5

La méthode Branch & Cut pour la résolution d'un problème quadratique indéfini à deux niveaux discret

5.1 Introduction

La programmation à deux niveaux en nombres entiers a été abordée par Thirwani et Arora ; Narang et Arora ([135, 104]). Les auteurs ont utilisés la principe des coupes pour arriver à une solution entière. L'algorithme développé par Thirwani et Arora [135] pour résoudre les problèmes quadratiques à deux niveaux PQB pour les variables entières est basé sur la technique de linéarisation et en utilisant la coupe de Gomory et la méthode dual de simplexe. Narang et Arora [104] ont présenté un algorithme pour résoudre les problèmes quadratiques indéfinis à deux niveaux en nombres entiers avec des variables bornées. Ils ont résolu le problème en résolvant le problème relaxé et ont développé une technique de solution par une coupe pour trouver la solution entière. Dans ce chapitre, nous considérons une classe spéciale de programmation à deux niveaux non linéaire dans laquelle la fonction objectif du premier niveau et la fonction objectif du second niveau sont un produit de deux fonctions linéaires, soit IIQBP. En supposant que pour la fonction objectif du premier niveau et la fonction objectif du second niveau, les deux facteurs linéaires sont positifs, des cas de maximisation sont considérés pour les deux niveaux. Les variables associées aux deux niveaux du problème sont liées par des contraintes linéaires et entières. Cette modélisation a été largement appliquée, notamment en économie (Hen-

derson et Quandt [69]). L'algorithme développé ici pour résoudre le **IIQBP** est basé sur une technique de Branch & Cut. En effet, pour déterminer la solution optimale entière du problème principal, on résout, en premier, le problème du premier niveau. La solution trouvée est ensuite testée pour l'optimalité du problème principal, en résolvant le problème du second niveau. Si cette solution n'est pas optimale pour le problème principal **IIQBP**, on l'élimine du domaine, en ajoutant la coupe de Dantzig au problème du premier niveau et on continue l'exploration.

5.2 Définitions et notations

Soit S l'ensemble des solutions réalisables $(x, y) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$ satisfaisant les contraintes

$$\begin{cases} A^1x + A^2y \leq b \\ x \geq 0, y \geq 0 \end{cases} \quad (5.1)$$

où, A^1 est $m \times n_1$ matrice, A^2 est $m \times n_2$ matrice et b un vecteur de dimension $\mathbb{R}^m$. Soit c^1, d^1, p^1, q^1 sont des vecteurs lignes dans $\mathbb{R}^{n_1}$; c^2, d^2, p^2, q^2 sont des vecteurs lignes dans $\mathbb{R}^{n_2}$, les constantes $\alpha_1, \alpha_2, \beta_1, \beta_2$ sont réelles et $\mathbb{Z}^n$ est l'ensemble des nombres entiers. Nous supposons que S est non vide et borné. Le problème quadratique indéfini à deux niveaux en nombres entiers **IIQBP**, que nous traitons peut être défini comme suit :

$$\text{IIQBP} \begin{cases} \max_{(x,y)} F(x, y) = (c^1x + c^2y + \alpha_1)(d^1x + d^2y + \beta_1) \\ \text{où } y \text{ résout} \\ \begin{cases} \max_y f(x, y) = (p^1x + p^2y + \alpha_2)(q^1x + q^2y + \beta_2) \\ s.c \\ (x, y) \in \mathcal{D} = S \cap \mathbb{Z}^n \end{cases} \end{cases} \quad (5.2)$$

où, $(c^1x + c^2y + \alpha_1)$, $(d^1x + d^2y + \beta_1)$, $(p^1x + p^2y + \alpha_2)$ et $(q^1x + q^2y + \beta_2)$ sont supposés être positifs pour tous les $x, y \in S$. Soit le problème quadratique indéfini en nombres entier du premier niveau, séparément du second niveau, appelé **ULIIQP**, contrôle la variable de décision x et le problème quadratique indéfini en nombres entier du second niveau, séparément du premier niveau, appelé **LLIIQP** contrôle la variable de décision y .

Mathématiquement, le problème du premier niveau **ULIIQP** peut être écrit comme suit :

$$\text{ULIIQP} \begin{cases} \max_{(x,y)} F(x, y) = (c^1x + c^2y + \alpha_1)(d^1x + d^2y + \beta_1) \\ s.c \quad (x, y) \in \mathcal{D} \end{cases} \quad (5.3)$$

et le problème du second niveau LLIIQP peut être écrit comme suit :

$$\text{LLIIQP} \begin{cases} \max_y f(x, y) = (p^1 x + p^2 y + \alpha_2)(q^1 x + q^2 y + \beta_2) \\ s.c \quad (x, y) \in \mathcal{D} \end{cases} \quad (5.4)$$

Pour $z = (x, y)$, $c = (c^1, c^2)^T$, $d = (d^1, d^2)^T$, $p = (p^1, p^2)^T$, $q = (q^1, q^2)^T$, le problème du premier niveau ULIIQP peut être réécrit :

$$\text{ULIIQP}, \begin{cases} \max_z F(z) = (c^T z + \alpha_1)(d^T z + \beta_1) \\ s.c \quad z \in \mathcal{D} \end{cases} \quad (5.5)$$

et le problème du second niveau LLIIQP est réécrit comme :

$$\text{LLIIQP}, \begin{cases} \max_y f(x, y) = (p^T \begin{pmatrix} x \\ y \end{pmatrix} + \alpha_2)(q^T \begin{pmatrix} x \\ y \end{pmatrix} + \beta_2) \\ s.c \quad (x, y) \in \mathcal{D} \end{cases} \quad (5.6)$$

Dans notre méthode, le problème quadratique indéfini à deux niveaux en nombres entiers IIQBP est résolu dans sa formulation originale. Le décideur du premier niveau choisit d'abord sa stratégie, puis en utilisant la solution optimale x^* du problème du premier niveau, le décideur du second niveau doit choisir sa meilleure stratégie y^* en optimisant son propre objectif Dempe S. [50]. Dans le problème principal, les problèmes du premier niveau et du second niveau sont tous deux des problèmes quadratiques indéfinis en nombres entiers. Par conséquent, la technique pour résoudre le problème de programmation quadratique indéfini en nombres entiers est présentée dans la section suivante.

5.3 Technique pour résoudre un problème quadratique indéfini en nombres entiers

Considérons le problème quadratique indéfini en nombres entiers suivant :

$$\text{IQP} \begin{cases} \max g(z) = (u^1 z + \alpha)(u^2 z + \beta) \\ s.c \quad z \in \mathcal{D} \end{cases} \quad (5.7)$$

où, $z = (x, y) \in \mathbb{R}^{n_1+n_2}$, $\mathcal{D} = S \cap \mathbb{Z}^n$, $S = \{z \mid Az \leq b, \quad z \geq 0\}$ est non vide et borné, $b \in \mathbb{R}^m$, u^1, u^2 sont des vecteurs ligne dans $\mathbb{R}^{n_1+n_2}$, α, β sont réelles et A est une matrice de dimension $m \times (n_1 + n_2)$. Nous supposons que les fonctions linéaires $(u^1 z + \alpha)$ et $(u^2 z + \beta)$ sont positives pour tout $z \in S$.

Considérons le problème linéaire bicritère en nombres entiers suivant associé à IQP :

$$\text{BLP} \quad \begin{cases} \max & g_1(z) = u^1 z + \alpha \\ \max & g_2(z) = u^2 z + \beta \\ \text{s.t} & z \in \mathcal{D} \end{cases} \quad (5.8)$$

La solution optimale de IQP est une solution efficace du problème BLP.

Théorème 5.3.1 *Une solution optimale $\hat{z}$ du problème IQP est une solution efficace du problème linéaire bicritère BLP qui maximise la valeur de g .*

Preuve 5.3.1 *Supposons que $\hat{z}$ soit une solution optimale du problème IQP mais qu'elle ne soit pas une solution efficace du problème linéaire bicritère BLP.*

Par définition, cela implique qu'il doit exister une solution réalisable notée z dans $\mathcal{D}$ qui domine $\hat{z}$. En d'autres termes, pour $g_i(z) = g_i$ et $g_i(\hat{z}) = \hat{g}_i$, il faut que l'une ou l'autre des deux conditions suivantes soit remplie

$$\hat{g}_1 \leq g_1 \quad \text{et} \quad \hat{g}_2 < g_2 \quad (5.9)$$

ou

$$\hat{g}_1 < g_1 \quad \text{et} \quad \hat{g}_2 \leq g_2. \quad (5.10)$$

Aussi, par hypothèse du problème IQP nous savons que $\hat{g} > 0$.

Par conséquent, nous devons avoir :

$$0 < \hat{g}_1 * \hat{g}_2 < g_1 * g_2 \quad (5.11)$$

Par conséquent, $\hat{z}$ ne peut pas être optimale.

Ainsi, le problème quadratique indéfini IQP est équivalent au problème suivant (Nash O.F. [105]) :

$$\text{IQP}, \quad \begin{cases} \max & g(z) = g_1(z) * g_2(z) \\ \text{s.t} & z \in \mathcal{D}_E \end{cases} \quad (5.12)$$

où, $g_1(z) = (u^1 z + \alpha)$, $g_2(z) = (u^2 z + \beta)$ et $\mathcal{D}_E$ est l'ensemble des solutions efficaces du problème linéaire bicritère BLP.

Par conséquent, la méthode de résolution du problème quadratique indéfini IQP consisterait à générer l'ensemble $\mathcal{D}_E$, l'ensemble des solutions efficaces du problème linéaire bicritère BLP puis à sélectionner la meilleure valeur optimale de g .

De nombreux algorithmes ont été proposés pour générer l'ensemble efficace de BLP en utilisant le concept d'efficacité.

Définition 5.3.1 *La solution z^1 est dominée par une autre solution z^2 , si $g_i(z^2) \geq g_i(z^1)$, pour $i = 1, 2$ et $g_k(z^2) > g_k(z^1)$, pour au moins un $k = 1, 2$.*

Définition 5.3.2 *Une solution réalisable z est appelée une solution efficace pour le problème BLP si et seulement si, il n'existe pas d'autres solutions $\tilde{z} \in \mathcal{D}$, tel que $g_i(\tilde{z}) \geq g_i(z)$, pour $i = 1, 2$ et $g_k(\tilde{z}) > g_k(z)$, pour au moins un $k = 1, 2$. Sinon, z est dominée par $\tilde{z}$.*

Dans ce qui suit, la procédure permettant de générer l'ensemble efficace de BLP est présentée en détail.

5.3.1 Procédure pour générer les solutions efficaces entières d'un problème linéaire bicritère

L'ensemble des solutions efficaces pour le problème BLP est généré, en utilisant, une stratégie basée sur la méthode "Branch & cut", décrite comme suit :

- **Initialisation** : Soit $\mathcal{D}_E$, $\bar{\mathcal{D}}_E$ sont les ensembles de toutes les solutions efficaces et de toutes les solutions dominées, respectivement, du problème linéaire bicritère (5.8) initialisés à l'ensemble vide, $l = 0$ et $S_0 = S$.
- **Le processus du pranchement** : La recherche de l'ensemble efficace est une exploration implicite structurée d'un arbre. A chaque noeud l de l'arbre, on optimise une des deux fonctions $g_i, i \in 1, 2$ sur le sous-domaine S_l (le choix d'une fonction ne changera pas l'ensemble efficient final, mais peut changer l'ordre des éléments dans celui-ci. Dans la suite, nous choisissons la première fonction). Le sous-problème associé à un nœud l est défini comme suit :

$$\text{LP}_l \begin{cases} \max & g_1(z) = u^1 z + \alpha \\ \text{s.c} & z \in S_l \end{cases} \quad (5.13)$$

où, $z = (x, y) \in \mathbb{R}^{n_1+n_2}$. On résout le problème LP_l en utilisant la méthode de simplexe, trois cas peuvent se présenter :

LP_l n'a pas de solution : Le noeud n'a pas de descendant et il est sondé.

LP_l n'a pas de solution entière : Soit $\hat{z}_j^{(l)}$ la composante de la solution optimale non entière $\hat{z}^{(l)}$ telle que $\hat{z}_j^{(l)} = \nu_j$, où, ν_j est un nombre fractionnaire. Le noeud l de l'arbre est alors séparé en deux nœuds l_1, l_2 , ($l_1 > l, l_2 > l, l_1 \neq l_2$) dont l'ensemble réalisable S_l est partitionné en deux parties (S_{l_1} et S_{l_2}) en ajoutant à S_l nouvelles contraintes $\hat{z}_j^{(l)} \leq \lfloor \nu_j \rfloor$ et $\hat{z}_j^{(l)} \geq \lfloor \nu_j \rfloor + 1$, respectivement.

LP_l a une solution entière : Soit $\hat{z}^{(l)}$ la solution optimale du problème LP_l.

- **Mise à jour de $\mathcal{D}_E$ et de $\bar{\mathcal{D}}_E$.** Cette solution est ensuite comparée aux solutions de l'ensemble $\mathcal{D}_E$ et $\bar{\mathcal{D}}_E$, puis les deux ensembles sont mis à jour : Au nœud l , correspondant à la solution $\hat{z}^{(l)}$, s'il existe $z \in \mathcal{D}_E$ tel que $(g_1(z), g_2(z))$ domine $(g_1(\hat{z}^{(l)}), g_2(\hat{z}^{(l)}))$ alors $\mathcal{D}_E$ n'est pas mis à jour et $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E \cup \{\hat{z}^{(l)}\}$. Sinon, s'il n'y a pas de solution $z \in \mathcal{D}_E$ tel que $(g_1(z), g_2(z))$ domine $(g_1(\hat{z}^{(l)}), g_2(\hat{z}^{(l)}))$ alors la solution $\hat{z}^{(l)}$ est ajoutée à l'ensemble $\mathcal{D}_E$ et toutes les solutions sont supprimées, tel que $z \in \mathcal{D}_E$, $(g_1(\hat{z}^{(l)}), g_2(\hat{z}^{(l)}))$ domine $(g_1(z), g_2(z))$, $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E \cup \{z\}$.
- **La technique des coupes :** Au nœud l , soit I_l , N_l sont les ensembles d'indices des variables de base et hors base respectivement (pour $j \in N_l, z_j = 0$) de la solution optimale entière $\hat{z}^{(l)}$ au tableau optimal de simplexe. On définit la contrainte :

$$\sum_{j \in H_l} z_j \geq 1. \quad (5.14)$$

où,

$$H_l = \{j \in N_l \mid \bar{u}_j^2 > 0\} \cup \{j \in N_l \mid \bar{u}_j^i = 0, \forall i \in \{1, 2\}\} \quad (5.15)$$

et $\bar{u}_j^i$ la j^{th} composante du vecteur gradient réduit $\bar{u}^i$ pour chaque fonction objectif g_i , $i \in \{1, 2\}$ au tableau optimal de simplexe.

L'inégalité (5.14) est appelée coupe efficace Chergui M. E-A. et Moulaï M.[43]. Si $H_l = \emptyset$ alors le domaine restant ne contient pas de solution efficace et le nœud l est sondé. (Proposition 1). Au contraire, si $H_l \neq \emptyset$ le nœud l a un successeur $l + 1$. Le domaine correspondant du nœud $l + 1$, le successeur de l , est obtenu en appliquant (5.14) à $\hat{z}^{(l)}$:

$$S_{l+1} = \left\{ z \in S_l : \sum_{j \in H_l} z_j \geq 1 \right\} \quad (5.16)$$

Chaque nœud l de l'arbre est sondé lorsque le problème correspondant LP_l n'est pas réalisable ou $H_l = \emptyset$.

L'algorithme est décrit comme suit :

Algorithm 1 Générer l'ensemble efficace du problème linéaire bicritère BLP

Etape 1 : Initialisation $l = 0$, $S_0 = S$, $\mathcal{D}_E = \emptyset$, $\bar{\mathcal{D}}_E = \emptyset$.

Etape 2 : Étape générale.

Tant qu'il y a pas de noeud non sondé dans l'arbre, on choisit le noeud l non encore sondé et on résout le problème linéaire correspondant (5.13).

- Si le problème (5.13) est irréalisable, alors le noeud correspondant l est sondé.
- Sinon, soit $(\hat{z}^{(l)})$ est une solution optimale du problème (5.13).
 1. Si $\hat{z}^{(l)}$ est entière, aller à l'étape 2a.
 2. Sinon, aller à l'étape 2b.

Etape 2a :

- Si le vecteur $(g_1(\hat{z}^{(l)}), g_2(\hat{z}^{(l)}))$ est non dominé par le vecteur $(g_1(z), g_2(z)) \forall z \in \mathcal{D}_E$ alors $\mathcal{D}_E = \mathcal{D}_E \cup \{\hat{z}^{(l)}\}$ et $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E$.
- S'il existe une solution $z \in \mathcal{D}_E$ tel que $(g_1(\hat{z}^{(l)}), g_2(\hat{z}^{(l)}))$ domine $(g_1(z), g_2(z))$ alors $\mathcal{D}_E = \mathcal{D}_E \setminus \{z\} \cup \{\hat{z}^{(l)}\}$ et $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E \cup \{z\}$.

Déterminer les ensembles N_l , H_l ;

1. Si $H_l = \emptyset$ alors le noeud l est sondé, aller à l'étape 2;
2. Sinon, ajouter la coupe efficace $\sum_{j \in H_l} z_j \geq 1$. au problème (5.13) et aller à l'étape 2, un nouveau noeud est créé, $l = l + 1$.

Step 2b : Processus de branchement Puisque parmi les valeurs $\hat{z}^{(l)}$ il existe au moins une valeur non entière, choisissez l'une d'entre elles de telle sorte que $\hat{z}_j^{(l)} = \nu_j$, où ν_j est un nombre fractionnaire. Partitionner l'ensemble réalisable S_l en deux parties (S_{l_1} et S_{l_2}) ($l_1 > l + 1, l_2 > l + 1, l_1 \neq l_2$) en ajoutant à S_l de nouvelles contraintes $\hat{z}_j^{(l)} \leq \lfloor \nu_j \rfloor$ et $\hat{z}_j^{(l)} \geq \lfloor \nu_j \rfloor + 1$, respectivement, aller à l'étape 2.

5.3.2 Résultats théoriques

Les notes théoriques suivantes montrent que l'algorithme 1 donne l'ensemble des solutions efficaces entières pour le problème (5.8) en un nombre fini d'itérations.

Théorème 5.3.2 *Supposons que $H_l \neq \emptyset$ à la solution entière courante $\hat{z}^{(l)}$.*

Si $z \neq \hat{z}^{(l)}$ est une solution efficace entière du problème (5.8) dans le domaine S_l , alors $z \in S_{l+1}$ (noeud $l+1$ est le successeur du noeud l).

Preuve 5.3.2 . *Soit $z \neq \hat{z}^{(l)}$ une solution entière dans le domaine S_l telle que $z \notin S_{l+1}$, ceci implique :*

$$z \in \left\{ z \in S_l : \sum_{j \in H_l} z_j < 1 \right\} \quad (5.17)$$

Par conséquent, z vérifie les inégalités.

$$\sum_{j \in H_l} z_j < 1 \quad (5.18)$$

$$\sum_{j \in N_l \setminus H_l} z_j \geq 1 \quad (5.19)$$

Cela signifie que $z_j = 0$ pour tout $j \in S_l$, et $z_j \geq 1$ pour au moins un indice $j \in N_l \setminus H_l$. Du tableau de simplexe correspondant à la solution $\hat{z}^{(l)}$, on déduit l'égalité suivante :

$$u^2 z = u^2 \hat{z}^{(l)} + \sum_{j \in N_l} \bar{u}_j^2 z_j = u^2 \hat{z}^{(l)} + \sum_{j \in H_l} \bar{u}_j^2 z_j + \sum_{j \in N_l \setminus H_l} \bar{u}_j^2 z_j \quad (5.20)$$

alors

$$u^2 z = u^2 \hat{z}^{(l)} + \sum_{j \in N_l \setminus H_l} \bar{u}_j^2 z_j \quad (5.21)$$

Ainsi, $u^2 z \leq u^2 \hat{z}^{(l)}$, avec $u^2 z < u^2 \hat{z}^{(l)}$ et $u^1 z \leq u^1 \hat{z}^{(l)}$ puisque, $\hat{z}^{(l)}$ est la solution optimale du problème, alors $z \notin \mathcal{D}_E$.

Proposition 5.3.1 *Supposons que $H_l = \emptyset$ à la solution entière courante. $\hat{z}^{(l)}$ alors il n'y a pas de solution efficace dans le domaine restant $S_l \setminus \hat{z}^{(l)}$.*

Preuve 5.3.3 *Supposons $H_l = \emptyset$ alors pour tout $j \in N_l$, $\bar{u}_j^2 < 0$ ou $\bar{u}_j^2 = 0$ et $\bar{u}_j^1 < 0$, on a $\bar{u}_j^1 < 0$ pour tout $j \in N_l$ puisque, c'est une solution optimale pour LP_l alors $\hat{z}^{(l)}$ est la meilleure solution dans le domaine S_l .*

Théorème 5.3.3 *L'algorithme proposé génère toutes les solutions entières efficaces du problème linéaire bicritère (5.8) en un nombre fini d'itérations, si de telles solutions existent.*

Preuve 5.3.4 *Soit $\mathcal{D}$, l'ensemble des solutions entières réalisables du problème BLP, est un ensemble fini borné contenu dans S . La cardinalité de l'ensemble efficace $\mathcal{D}_E$ est un nombre fini puisqu'il contient un nombre fini de solutions entières. L'arborescence aurait alors un nombre fini de branches. Par conséquent, l'algorithme se termine en un nombre fini d'étapes.*

Exemple 5.3.1 *Soit le problème linéaire bicritère BLP suivant :*

$$BLP \left\{ \begin{array}{l} \max \quad g_1(x_1, x_2, x_3) = 1/2x_1 + 5x_2 - 1/2x_3 + 2 \\ \max \quad g_2(x_1, x_2, x_3) = -1/3x_2 + 3x_3 + 4 \\ s.c \\ x_1 - 2x_2 \leq 5 \\ x_1 + 3x_2 \leq 8 \\ 2x_2 + 5x_3 \leq 6 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, \text{entiers} \end{array} \right. \quad (5.22)$$

où $(y_1, y_2) = (x_2, x_3)$. Le problème BLP est résolu en utilisant l'algorithme 1 :

Le correspondant problème linéaire LP_0 est :

$$LP_0 \left\{ \begin{array}{l} \max \quad g_1(x_1, x_2, x_3) = 1/2x_1 + 5x_2 - 1/2x_3 + 2 \\ s.c \\ x_1 - 2x_2 \leq 5 \\ x_1 + 3x_2 \leq 8 \\ 2x_2 + 5x_3 \leq 6 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, \text{entiers} \end{array} \right. \quad (5.23)$$

Notons qu'à la place de g_1 , on peut de la même façon considérer le problème LP_0 avec l'objectif g_2 .

Etape 1 : $l = 0$, $\mathcal{D}_E^0 = \emptyset$. Le programme (LP_0) est résolu. La solution optimale solution est $x^{*(0)} = (0, 8/3, 0)$, donnée dans la Table 1.

TABLE 1 –

B	x_1	x_3	x_5	RHS
x_4	$5/3$	0	$2/3$	$31/3$
x_2	$1/3$	0	$1/3$	$8/3$
x_6	$-2/3$	5	$-2/3$	$2/3$
$\bar{u}_j^1$	$-7/6$	$-1/2$	$-5/3$	$g_1 = 46/3$
$\bar{u}_j^2$	$1/9$	3	$1/9$	$g_2 = 28/9$

qui n'est pas entière. Le processus du branchement génère deux noeuds $node_1$, $node_2$ correspondant aux contraintes suivantes :

$node_1 : x_2 \geq 3$.

$node_2 : x_2 \leq 2$.

$node_2$: la contrainte $x_2 \leq 2$ est ajoutée au problème LP_0 , on obtient le problème LP_2 , la solution optimale entière du problème est $x^{*(2)} = (2, 2, 0)$ donnée dans la table 2 :

TABLE 2 –

B	x_3	x_5	x_7	RHS
x_4	0	-1	5	7
x_2	0	0	1	2
x_6	5	0	-2	2
x_1	0	1	-3	2
$\bar{u}_j^1$	$-1/2$	$-1/2$	$-7/2$	$g_1 = 13$
$\bar{u}_j^2$	3	0	$1/3$	$g_2 = 10/3$

Le vecteur critère est $(13, 10/3)$, cependant, l'ensemble $\mathcal{D}_E = \{(2, 2, 0)\}$ est mis à jour. $H_2 = \{3, 7\}$. Appliquer la coupe efficace $x_3 + x_7 \geq 1$, et résoudre le nouveau problème LP_3 , la solution optimale est $x^{*(3)} = (23/7, 11/7, 4/7)$ donnée dans la table 3 :

TABLE 3 –

B	x_5	x_6	x_8	RHS
x_4	-1	$5/7$	$25/7$	$34/7$
x_2	0	$1/7$	$5/7$	$11/7$
x_7	0	$-1/7$	$-5/7$	$3/7$
x_1	1	$-3/7$	$-15/7$	$23/7$
x_3	0	$1/7$	$-2/7$	$4/7$
$\bar{u}_j^1$	$-1/2$	$-3/7$	$-37/14$	$g_1 = 129/14$
$\bar{u}_j^2$	0	$-8/21$	$23/21$	$g_2 = 25/21$

qui n'est pas entière. Le processus du branchement génère deux noeuds $node_4$, $node_5$ correspondant aux contraintes suivantes :

$node_4 : x_1 \geq 4$.

$node_5 : x_1 \leq 3$.

$node_5$: la contrainte $x_1 \leq 3$ est ajoutée au problème LP_3 , on obtient le problème LP_5 , le problème est résolu, la solution optimale est $x^{*(5)} = (3, 11/7, 4/7)$ donnée dans la table 4 :

TABLE 4 –

B	x_6	x_8	x_9	RHS
x_4	$2/7$	$10/7$	-1	$36/7$
x_2	$1/7$	$5/7$	0	$11/7$
x_7	$-1/7$	$-5/7$	0	$3/7$
x_1	0	0	1	3
x_3	$1/7$	$-2/7$	0	$4/7$
x_5	$-3/7$	$-15/7$	-1	$2/7$
$\bar{u}_j^1$	$-9/14$	$-26/7$	$-1/2$	$g_1 = 155/14$
$\bar{u}_j^2$	$-8/21$	$23/21$	0	$g_2 = 109/21$

qui n'est pas entière. Le processus du branchement génère deux noeuds $node_6$, $node_7$ correspondant aux contraintes suivantes :

$node_6 : x_2 \leq 1$.

$node_7 : x_2 \geq 2$.

$node_6$: la contrainte $x_2 \leq 1$ est ajoutée au problème LP_5 , on obtient le problème LP_6 , dont, la solution optimale est $x^{*(6)} = (3, 1, 0)$, donnée dans la table 5 :

TABLE 5 –

B	x_8	x_9	x_{10}	RHS
x_4	0	-1	2	4
x_2	0	0	1	1
x_7	0	0	-1	1
x_1	0	1	0	3
x_3	-1	0	1	0
x_5	0	-1	-3	2
x_6	5	0	-7	4
$\bar{u}_j^1$	$-1/2$	$-1/2$	$-9/2$	$g_1 = 17/2$
$\bar{u}_j^2$	3	0	$-8/3$	$g_2 = 11/3$

le vecteur critère est $(17/2, 11/3)$, non dominé, cependant, l'ensemble $\mathcal{D}_E = \{(2, 2, 0), (3, 1, 0)\}$ est mis à jour. $H_2 = \{8\}$. Appliquer la coupe efficace $x_8 \geq 1$, est résoudre le nouveau

problème LP_8 , la solution optimale est $x^{*(8)} = (3, 6/7, 6/7)$ donnée dans la table 6 :

TABLE 6 –

B	x_6	x_9	x_{11}	RHS
x_4	$2/7$	-1	$10/7$	$26/7$
x_2	$1/7$	0	$5/7$	$6/7$
x_7	$-1/7$	0	$-5/7$	$8/7$
x_1	0	1	0	3
x_3	$1/7$	0	$-2/7$	$6/7$
x_5	$-3/7$	-1	$-15/7$	$17/7$
x_{10}	$-1/7$	0	$-5/7$	$1/7$
x_8	0	0	-1	1
$\bar{u}_j^1$	$-9/14$	$-1/2$	$-26/7$	$g_1 = 75/14$
$\bar{u}_j^2$	$-8/21$	0	$23/21$	$g_2 = 16/7$

qui n'est pas entière. Le processus du branchement génère deux noeuds $node_9$, $node_{10}$ correspondant aux contraintes suivantes :

$node_9 : x_2 \geq 1$.

$node_{10} : x_2 \leq 0$.

$node_{10}$: la contrainte $x_2 \leq 0$ est ajoutée au problème LP_8 , on obtient le problème LP_{10} , dont, la solution optimale est $x^{*(10)} = (3, 0, 0)$ donnée la table 7 :

TABLE 7 –

B	x_9	x_{11}	x_{12}	RHS
x_4	-1	0	2	2
x_2	0	0	1	0
x_7	0	0	-1	2
x_1	1	0	0	3
x_3	0	-1	1	0
x_5	-1	0	-3	5
x_{10}	0	0	-1	1
x_8	0	-1	0	1
x_6	0	5	-7	6
$\bar{u}_j^1$	-1/2	-1/2	-9/2	$g_1 = 7/2$
$\bar{u}_j^2$	0	3	-8/3	$g_2 = 4$

le vecteur critère est $(7/2, 4)$, non dominé, cependant, l'ensemble $\mathcal{D}_E = \{(2, 2, 0), (3, 1, 0), (3, 0, 0)\}$ est mis à jour. $H_2 = \{11\}$. Appliquer la coupe efficace $x_{11} \geq 1$, et résoudre le nouveau problème LP_{11} , la solution optimale est $x^{*(11)} = (3, 0, 1)$ donnée dans la table 8 :

TABLE 8 –

B	x_9	x_{12}	x_{13}	RHS
x_4	-1	2	0	2
x_2	0	1	0	0
x_7	0	-1	0	2
x_1	1	0	0	3
x_3	0	1	-1	1
x_5	-1	-3	0	5
x_{10}	0	-1	0	1
x_8	0	0	-1	2
x_6	0	-7	5	1
x_{11}	0	0	-1	1
$\bar{u}_j^1$	-1/2	-9/2	-1/2	$g_1 = 3$
$\bar{u}_j^2$	0	-8/3	3	$g_2 = 7$

le vecteur critère est $(3, 7)$, non dominé, cependant, l'ensemble $\mathcal{D}_E = \{(2, 2, 0), (3, 1, 0), (3, 0, 0), (3, 0, 1)\}$ est mis à jour. $H_2 = \{13\}$. La coupe efficace correspondante est ajoutée au problème LP_{11} donne le problème LP_{12} qui est irréalisable, alors le noeud est sondé.

$node_9$: la contrainte $x_2 \geq 1$ ajoutée au problème LP_8 , le problème résultat est irréalisable, donc, le noeud est sondé.

$node_7$: la contrainte $x_2 \geq 2$ ajoutée au problème LP_5 le problème résultat est irréalisable, donc, le noeud est sondé.

$node_4$: la contrainte $x_1 \geq 4$ ajoutée au problème LP_3 , et le problème résultat LP_4 , est résolu, la solution optimale est $x^{*(4)} = (4, 4/3, 1/3)$ donnée dans la table 9 :

qui n'est pas entière. Le processus du branchement génère deux noeuds $node_{13}$, $node_{14}$ correspondant aux contraintes suivantes :

$node_{13}$: $x_2 \leq 1$.

$node_{14}$: $x_2 \geq 2$.

$node_{13}$: la contrainte $x_2 \leq 1$ est ajoutée au problème LP_4 , on obtient le problème LP_{13} , dont, la solution optimale est $x^{*(13)} = (5, 1, 0)$ donnée dans la table 10 :

le vecteur critère est $(19/2, 11/3)$, non dominé, alors que, le vecteur correspondant à la solution $(3, 1, 0)$ est dominé, cependant, l'ensemble $\mathcal{D}_E = \{(2, 2, 0), (3, 0, 0), (3, 0, 1), (5, 1, 0)\}$ est mis à jour. $H_2 = \{5, 8\}$. Appliquer la coupe $x_5 + x_8 \geq 1$, et résoudre le nouveau

TABLE 9 –

B	x_5	x_8	x_9	RHS
x_4	$2/3$	0	$5/3$	$11/3$
x_2	$1/3$	0	$1/3$	$4/3$
x_7	$-1/3$	0	$-1/3$	$2/3$
x_1	0	0	-1	4
x_3	$1/3$	-1	$1/3$	$1/3$
x_6	$-7/3$	5	$-7/3$	$5/3$
$\bar{u}_j^1$	$-3/2$	$-1/2$	-1	$g_1 = 21/2$
$\bar{u}_j^2$	$-8/9$	3	$-8/9$	$g_2 = 41/9$

TABLE 10 –

B	x_5	x_8	x_9	RHS
x_4	-1	0	5	2
x_2	0	0	1	1
x_7	0	0	-1	1
x_1	1	0	-3	5
x_3	0	-1	1	0
x_6	0	5	-7	4
x_9	1	0	-3	1
$\bar{u}_j^1$	$-1/2$	$-1/2$	-3	$g_1 = 19/2$
$\bar{u}_j^2$	0	3	$-8/3$	$g_2 = 11/3$

problème LP_{15} , la solution optimale est $x^{*(15)} = (24/5, 1, 4/5)$ donnée dans la table 11 :

TABLE 11 –

B	x_6	x_{10}	x_{11}	RHS
x_4	$-1/5$	$32/5$	-1	$11/5$
x_2	0	1	0	1
x_7	0	-1	0	1
x_1	$1/5$	$-22/5$	1	$24/5$
x_3	$1/5$	$-2/5$	0	$4/5$
x_5	$-1/5$	$7/5$	-1	$1/5$
x_9	$1/5$	$-22/5$	1	$4/5$
x_8	$1/5$	$-7/5$	0	$4/5$
$\bar{u}_j^1$	0	-3	$-1/2$	$g_1 = 7$
$\bar{u}_j^2$	$-3/5$	$23/15$	0	$g_2 = 31/15$

qui n'est pas entière. Deux noeuds sont générés $node_{16}$, $node_{17}$:

$node_{16} : x_1 \leq 4$.

$node_{17} : x_1 \geq 5$.

$node_{16}$: la contrainte $x_1 \leq 4$ est ajoutée au problème LP_{15} , on obtient le problème LP_{16} , où, la solution optimale est $x^{*(16)} = (4, 1, 0)$, donnée dans la table 12 :

le vecteur critère est $(9, 11/3)$, il est dominée, $\mathcal{D}_E = \{(2, 2, 0), (3, 0, 0), (3, 0, 1), (5, 1, 0)\}$.

$H_2 = \{11, 12\}$. Appliquer la coupe efficace $x_{11} + x_{12} \geq 1$, et résoudre le nouveau problème LP_{18} , la solution optimale est $x^{*(18)} = (4, 21/22, 9/11)$ table 13 :

TABLE 12 –

B	x_{10}	x_{11}	x_{12}	RHS
x_4	2	0	-1	3
x_2	1	0	0	1
x_7	-1	0	0	1
x_1	0	0	1	4
x_3	4	-1	1	0
x_5	-3	0	-1	1
x_9	0	0	1	0
x_8	3	-1	1	0
x_6	-22	5	-5	4
$\bar{u}_j^1$	-3	-1/2	0	$g_1 = 9$
$\bar{u}_j^2$	-35/3	3	-3	$g_2 = 11/3$

TABLE 13 –

B	x_6	x_9	x_{13}	RHS
x_4	1/11	21/11	5/11	32/11
x_2	1/22	5/11	5/22	21/22
x_7	-1/22	-5/11	-5/22	23/22
x_1	0	-1	0	4
x_{11}	0	-1	-1	1
x_5	-3/22	-4/11	-15/22	25/22
x_3	2/11	-2/11	-1/11	9/11
x_8	3/22	-7/11	-7/22	19/22
x_{10}	-1/22	-5/11	-5/22	1/22
x_{12}	0	1	0	0
$\bar{u}_j^1$	-3/22	-41/22	-13/11	$g_1 = 70/11$
$\bar{u}_j^2$	-35/66	23/33	23/66	$g_2 = 47/22$

qui n'est pas entière. Deux noeuds sont générés $node_{19}$, $node_{20}$:
 $node_{19} : x_2 \geq 1$.

$node_{20} : x_2 \leq 0$.

$node_{20} : la\ contrainte\ x_2 \leq 0\ est\ ajoutée\ au\ problème\ LP_{18},\ le\ problème\ obtenu\ est\ LP_{20},$
 $la\ solution\ optimale\ est\ x^{*(20)} = (4, 0, 0),\ table\ 14 :$

TABLE 14 –

B	x_3	x_{12}	x_{14}	RHS
x_4	0	-1	2	1
x_2	0	0	1	0
x_7	0	0	-1	2
x_1	0	1	0	4
x_{11}	-1	-1	-4	4
x_5	0	-1	-3/22	4
x_9	0	1	0	0
x_8	-1	0	-1	1
x_{10}	0	0	-1	1
x_{13}	-1	-2	-4	3
x_6	5	0	-2	6
$\bar{u}_j^1$	-1/2	-1/2	-5	$g_1 = 4$
$\bar{u}_j^2$	3	0	1/3	$g_2 = 4$

le vecteur critère est $(4, 4)$, non dominé, par contre le vecteur correspondant à la solution $(3, 0, 0)$ est dominé, donc $\mathcal{D}_E = \{(2, 2, 0), (3, 0, 1), (5, 1, 0), (4, 0, 0)\}$. $H_2 = \{3, 14\}$. La coupe efficace $x_3 + x_{14} \geq 1$ est appliqué, et le nouveau problème LP_{21} est résolu, la solution optimale est $x^{*(21)} = (4, 0, 1)$, table 15 :

le vecteur critère est $(7/2, 7)$, non dominé, mais le vecteur correspondant à la solution $(3, 0, 1)$ est dominé, cependant, $\mathcal{D}_E = \{(2, 2, 0), (5, 1, 0), (4, 0, 0), (4, 0, 1)\}$. $H_2 = \{15\}$, la coupe efficace est ajoutée au problème LP_{21} , le problème LP_{22} obtenu est irréalisable, alors, le noeud est sondé .

$node_{19} : la\ contrainte\ x_2 \geq 1\ ajoutée\ au\ problème\ LP_{18},\ le\ problème\ obtenu\ est\ irréalisable,$
 $le\ noeud\ est\ sondé.$

$node_{17} : la\ contrainte\ x_1 \geq 5\ ajoutée\ au\ problème\ LP_{15},\ la\ solution\ optimale\ du\ problème\ LP_{17}$
 $est\ x^{*(17)} = (5, 21/22, 9/11),\ table\ 16 :$

TABLE 15 –

B	x_{12}	x_{14}	x_{15}	RHS
x_4	-1	2	0	1
x_2	0	1	0	0
x_7	0	-1	0	2
x_1	1	0	0	4
x_{11}	-1	-3	-1	5
x_5	-1	-3	0	4
x_9	1	0	0	0
x_8	0	0	-1	2
x_{10}	0	-1	0	1
x_{13}	-2	-3	-1	4
x_6	0	-7	5	1
x_3	0	1	-1	1
$\bar{u}_j^1$	$-1/2$	$-9/2$	$-1/2$	$g_1 = 7/2$
$\bar{u}_j^2$	0	$-8/3$	3	$g_2 = 7$

TABLE 16 –

B	x_6	x_{11}	x_{12}	RHS
x_4	$1/11$	$5/11$	$16/11$	$21/11$
x_2	$1/22$	$5/22$	$5/22$	$21/22$
x_7	$-1/22$	$-5/22$	$-5/22$	$23/22$
x_1	0	0	-1	5
x_3	$2/11$	$-1/11$	$-1/11$	$9/11$
x_5	$-3/22$	$-15/22$	$7/22$	$3/22$
x_9	0	0	-1	1
x_8	$3/22$	$-7/22$	$-7/22$	$19/22$
x_{10}	$-1/22$	$-5/22$	$-5/22$	$1/22$
$\bar{u}_j^1$	$-3/22$	$-13/11$	$-15/22$	$g_1 = 195/22$
$\bar{u}_j^2$	$-35/66$	$23/66$	$23/66$	$g_2 = 135/22$

qui n'est pas entière. Deux noeuds sont générés $node_{23}$, $node_{24}$:

$node_{23} : x_2 \geq 1$.

$node_{24} : x_2 \leq 0$.

$node_{24}$: la contrainte $x_2 \leq 0$ est ajoutée au problème LP_{17} , le problème obtenu LP_{24} est résolu, la solution optimale est $x^{*(24)} = (5, 0, 0)$, table 17 :

TABLE 17 –

B	x_3	x_4	x_{13}	RHS
x_{11}	-1	-1	6	3
x_2	0	0	1	0
x_7	0	0	-1	2
x_1	0	1	2	5
x_{12}	0	1	2	0
x_5	0	-1	-5	3
x_9	0	1	2	1
x_8	-1	0	-1	1
x_{10}	0	0	-1	1
x_6	5	0	-2	6
$\bar{u}_j^1$	-1/2	-1/2	-6	$g_1 = 9/2$
$\bar{u}_j^2$	3	0	1/3	$g_2 = 4$

le vecteur critère est $(9/2, 4)$, non dominé, par contre le vecteur correspondant à la solution $(4, 0, 0)$ est dominé, d'où, $\mathcal{D}_E = \{(2, 2, 0), (5, 1, 0), (4, 0, 1), (5, 0, 0)\}$.

$H_2 = \{3, 4, 13\}$ la coupe efficace correspondante est ajoutée au problème LP_{17} , le problème obtenu LP_{25} est irréalisable, le noeud est sondé.

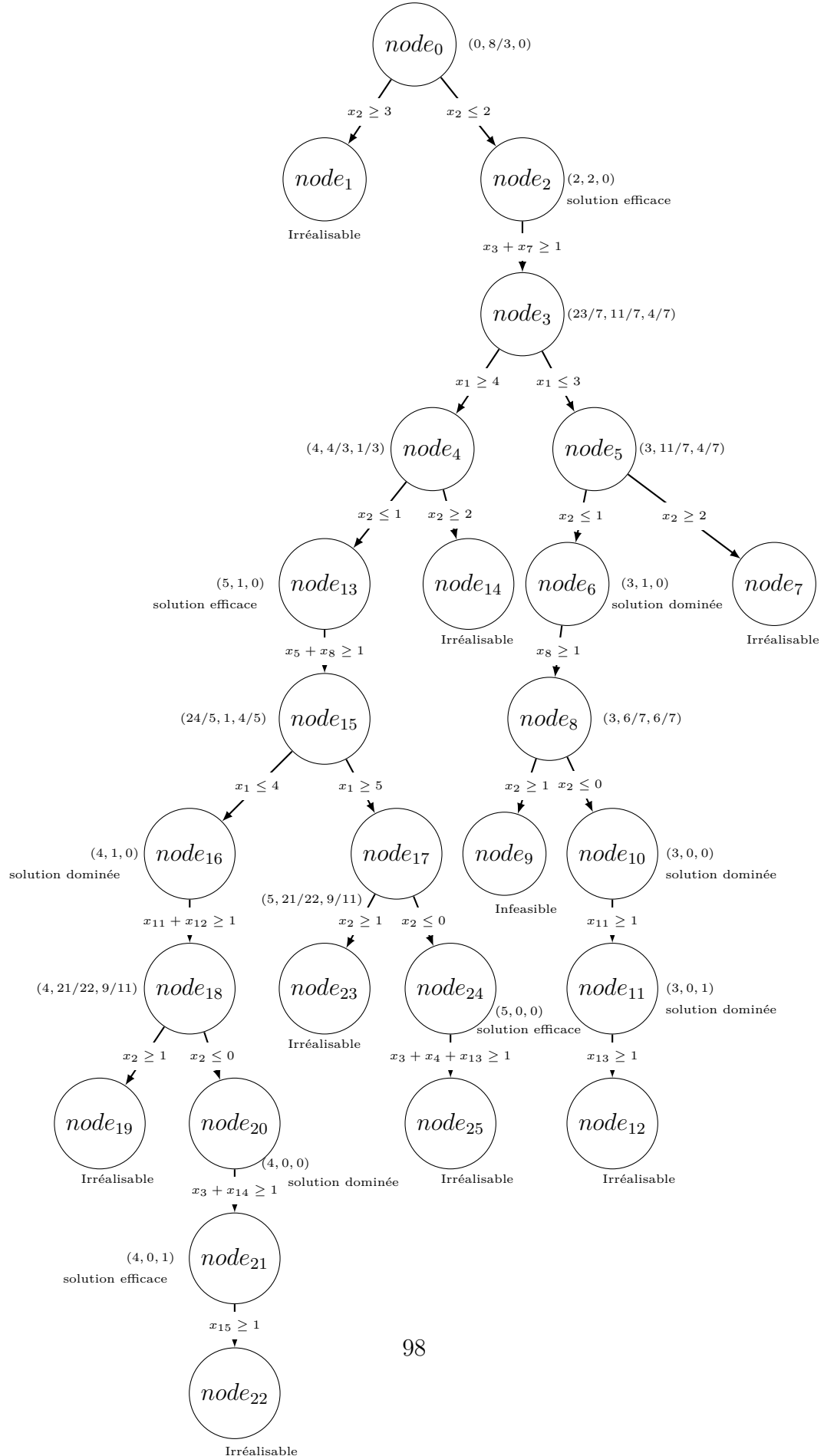
$node_{23}$: la contrainte $x_2 \geq 1$ ajoutée au problème LP_{17} le problème obtenu est irréalisable, le noeud est sondé.

$node_{14}$: la contrainte $x_2 \geq 2$ ajoutée au problème LP_4 le problème obtenu est irréalisable, le noeud est sondé.

$node_1$: la contrainte $x_2 \geq 3$ ajoutée au problème LP_0 le problème obtenu est irréalisable, le noeud est sondé.

Donc, l'ensemble des solutions efficaces du problème bicritère est

$\mathcal{D}_E = \{(2, 2, 0), (5, 1, 0), (4, 0, 1), (5, 0, 0)\}$.



5.3.3 Méthodologie et Algorithme pour la résolution de problème quadratique indéfini en nombres entiers IQP

En utilisant le théorème 1, nous pouvons proposer l'algorithme 2 pour résoudre le problème IQP (5.12).

Algorithm 2 Résolution du problème quadratique indéfini IQP

Etape 0 : Initialisation De IQP, on détermine le problème bicritère correspondant BLP.

Etape 1 : En utilisant l'algorithme ci-dessus (Algorithme 1), on génère l'ensemble de toutes les solutions efficaces entières du problème BLP.

Etape 2 : Soit $\mathcal{D}_E$, est l'ensemble de toutes les solutions efficaces entières du problème BLP, alors la solution optimale entière du problème IQP est $z^* = \arg \max_{z \in \mathcal{D}_E} g(z)$ et $g_{opt} = g(z^*)$.

Exemple 5.3.2 Pour illustrer le déroulement de l'Algorithme 2, considérons le problème quadratique indéfini suivant pour $z = (x_1, y_1, y_2)$:

$$IQP \left\{ \begin{array}{l} \max \quad g(x_1, y_1, y_2) = (1/2x_1 + 5y_1 - 1/2y_2 + 2)(-1/3y_1 + 3y_2 + 4) \\ s.c \\ x_1 - 2y_1 \leq 5 \\ x_1 + 3y_1 \leq 8 \\ 2y_1 + 5y_2 \leq 6 \\ x_1 \geq 0, y_1 \geq 0, y_2 \geq 0, \text{entiers} \end{array} \right. \quad (5.24)$$

Pour résoudre le problème quadratique indéfini IQP, soit le problème linéaire bicritère correspondant défini comme :

$$BLP \left\{ \begin{array}{l} \max \quad g_1(x_1, x_2, x_3) = 1/2x_1 + 5x_2 - 1/2x_3 + 2 \\ \max \quad g_2(x_1, x_2, x_3) = -1/3x_2 + 3x_3 + 4 \\ s.c \\ x_1 - 2x_2 \leq 5 \\ x_1 + 3x_2 \leq 8 \\ 2x_2 + 5x_3 \leq 6 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, \text{entiers} \end{array} \right. \quad (5.25)$$

où $(y_1, y_2) = (x_2, x_3)$. Le **BLP** est résolu en utilisant l'algorithme 1 et l'ensemble des solutions efficaces est déterminé $\mathcal{D}_E = \{(2, 2, 0), (5, 1, 0), (4, 0, 1), (5, 0, 0)\}$, les vecteurs de critère correspondants, respectivement, sont $\{(13, 10/3), (19/2, 11/3), (7/2, 7), (9/2, 4)\}$, alors la solution optimale entière de du problème **IQP** est $z^* = (2, 2, 0) \in \mathcal{D}_E$, où $g_{opt} = g(z^*) = \max g(z) = 13 * 10/13 = 130/3$.

5.4 Méthode "Branch & cut" pour la résolution des problèmes quadratiques indéfinis à deux niveaux en nombres entiers IIQBP

Dans cette section, une méthode exacte basée sur la technique de "Branch & cut" est présentée en détail. Ensuite, nous prouverons la convergence de la méthode.

5.4.1 Description de la méthode

Considérons un programme quadratique indéfini à deux niveaux en nombres entiers IIQBP, les étapes générales pour résoudre ce problème peuvent être décrites comme suit :

- **Etape 1 : Initialisation** On initialise $F_{opt} = -\infty$, $f_{opt} = -\infty$, $z_{opt} = \emptyset$, $k = 0$ et $S_0 = S$.
- **Etape 2 :** Nous résolvons le problème du premier niveau (5.26) sur le sous-domaine S_k correspondant au nœud k en utilisant l'algorithme 2.

$$\text{ULIIQP}_k \begin{cases} \max_z F(z) = F_1(z) * F_2(z) \\ s.c \quad z \in S_k \end{cases} \quad (5.26)$$

où $F_1(z) = c^T z + \alpha_1$ et $F_2(z) = d^T z + \beta_1$.

Si le problème (5.26) n'a pas de solution, le problème principal est irréalisable et l'algorithme s'arrête. Sinon, aller à l'étape 3.

- **Etape 3 :** Soit $z^{*(k)} = (x^{*(k)}, y^{*(k)})$ est la solution optimale entière du premier niveau et $F_{opt} = F(z^{*(k)})$. I_k , N_k sont les ensembles d'indices des variables de base et hors base respectivement, de la solution optimale $z^{*(k)}$ obtenue par l'algorithme 2. Cette solution est ensuite testée pour vérifier l'optimalité du problème principal en résolvant le problème de second niveau (5.27) [50] : On fixe $x = x^{*(k)}$ et utiliser à nouveau l'algorithme 2 ci-dessus pour résoudre le second niveau (5.27),

$$\text{LLIIQP} \begin{cases} \max_y f(x^{*(k)}, y) = (p^1 x^{*(k)} + p^2 y + \alpha_2)(q^1 x^{*(k)} + q^2 y + \beta_2) \\ s.c \quad (x^{*(k)}, y) \in S_k \end{cases} \quad (5.27)$$

Soit $\tilde{y}$ la solution optimale entière. Deux cas peuvent se présenter :

1. Si $y^{*(k)} = \tilde{y}$, alors $z^{*(k)} = (x^{*(k)}, y^{*(k)})$ est la solution optimale entière du problème principal (5.2), $F_{opt} = F(z^{*(k)})$, $f_{opt} = f(z^{*(k)})$, $z_{opt} = z^{*(k)}$ et l'algorithme se termine.
 2. Sinon, si $y^{*(k)} \neq \tilde{y}$ ou le second niveau (5.27) est irréalisable, aller à l'étape 4.
- **Etape 4 :** Afin d'améliorer les performances de l'algorithme, au lieu d'initialiser $\mathcal{D}_E = \emptyset$, $\bar{\mathcal{D}}_E = \emptyset$ dans l'algorithme 2, on initialise $\mathcal{D}_E = \mathcal{D}_E^{k-1}$, $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E^{k-1}$ (pour $k \geq 1$). Pour éviter de réexplorer les solutions déjà trouvées auparavant. Nous mettons à jour l'ensemble $\mathcal{D}_E^k$ et l'ensemble $\bar{\mathcal{D}}_E^k$.
- $\mathcal{D}_E^k = \mathcal{D}_E^{k-1} \setminus \{z^{*(k)}\}$ et $\bar{\mathcal{D}}_E^k = \bar{\mathcal{D}}_E^{k-1}$.
 - S'il existe une solution $\bar{z} \in \bar{\mathcal{D}}_E^k$ de telle sorte que le vecteur $(F_1(\bar{z}), F_2(\bar{z}))$ n'est pas dominé par le vecteur $(F_1(z), F_2(z)) \forall z \in \mathcal{D}_E^k$, alors $\mathcal{D}_E^k = \mathcal{D}_E^k \cup \{\bar{z}\}$, $\bar{\mathcal{D}}_E^k = \bar{\mathcal{D}}_E^k \setminus \{\bar{z}\}$, aller à l'étape 5.
- **Etape 5 :** Dans cette étape, la solution $z^{*(k)}$ n'est pas optimale pour le problème principal IIQBP. Donc, on élimine cette solution du domaine S_k et on continue l'exploration à partir du nœud k , où S_{k+1} est le domaine correspondant du nœud $k+1$, successeur de k , obtenu en appliquant l'inégalité (5.29) à $z^{*(k)}$:

$$S_{k+1} = \left\{ z \in S_k : \sum_{j \in N_k} z_j \geq 1 \right\} \quad (5.28)$$

L'inégalité (5.29) est définie comme suit :

$$\sum_{j \in N_k} z_j \geq 1. \quad (5.29)$$

où N_k est l'ensemble des indices des variables hors base de la solution optimale $z^{*(k)}$ obtenue par l'algorithme 2. Cette inégalité (5.29) est appelée une coupe de Dantzig et a la propriété d'éliminer la solution entière $z^{*(k)}$ du problème du premier niveau (5.26) et elle est ajoutée aux nœuds successeurs de k .

$\mathcal{D}_E = \mathcal{D}_E^k$, $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E^k$, $k = k+1$, aller à l'étape 2.

L'algorithme qui détermine une solution optimale entière de notre problème principal (5.2) se résume comme suit :

Algorithm 3 Algorithme "Branch & cut" pour la résolution du problème IIQBP.

Etape 1 : Initialisation $k = 0$, $F_{opt} = -\infty$ et $f_{opt} = -\infty$, $z_{opt} = \emptyset$.

Résoudre le problème du premier niveau $ULIQP_0$ en utilisant l'algorithme 2 ci-dessus, l'ensemble des solutions efficaces et les solutions dominées, respectivement, $\mathcal{D}_E$ et $\bar{\mathcal{D}}_E$ sont déterminées, $\mathcal{D}_E^0 = \mathcal{D}_E$, $\bar{\mathcal{D}}_E^0 = \bar{\mathcal{D}}_E$.

- Si $ULIQP_0$ est irréalisable, alors la solution optimale entière à deux niveaux du problème principal (5.2) n'existe pas, et l'algorithme s'arrête.
- Sinon, soit $z^{*(0)} = (x^{*(0)}, y^{*(0)})$, la solution optimale du premier niveau ($ULIQP_0$), $F_{opt} = F(z^{*(0)})$ aller à l'étape 3.

Etape 2 : Étape générale. Résoudre le problème du premier niveau (5.26).

Utiliser l'algorithme 2 ci-dessus. L'ensemble des solutions efficaces et les solutions dominées, respectivement, $\mathcal{D}_E$ et $\bar{\mathcal{D}}_E$ sont mises à jour, alors, $\mathcal{D}_E^k = \mathcal{D}_E$, $\bar{\mathcal{D}}_E^k = \bar{\mathcal{D}}_E$. Soit $z^{*(k)} = (x^{*(k)}, y^{*(k)})$, la solution optimale du premier niveau (5.26).

- Si $ULIQP_k$ est irréalisable ou $F(z^{*(k)}) > F_{opt}$ (pour $k \geq 1$), alors le nœud correspondant k est sondé.
- Sinon, $F_{opt} = F(z^{*(k)})$ aller à l'étape 3.

Etape 3 : Pour $x = x^{*(k)}$, résoudre le problème du second niveau (5.27) par la même technique citée dans l'algorithme 2.

- Soit $\tilde{y}$ est la solution optimale entière du problème de second niveau (5.27).
 1. Si $y^{*(k)} = \tilde{y}$ alors $z^{*(k)} = (x^{*(k)}, y^{*(k)})$, est la solution optimale entière à deux niveaux du problème principal (5.2), $F_{opt} = F(z^{*(k)})$, $f_{opt} = f(z^{*(k)})$, $z_{opt} = z^{*(k)}$ et l'algorithme s'arrête.
 2. Dans le cas contraire, passez à l'étape 4.

Etape 4 : Mise à jour de l'ensemble $\mathcal{D}_E^k$ et l'ensemble $\bar{\mathcal{D}}_E^k$. $\mathcal{D}_E^k = \mathcal{D}_E^k \setminus \{z^{*(k)}\}$ et $\bar{\mathcal{D}}_E^k = \bar{\mathcal{D}}_E^k$.

- S'il existe une solution $\bar{z} \in \bar{\mathcal{D}}_E^k$ tel que le vecteur $(F_1(\bar{z}), F_2(\bar{z}))$ est non dominé par le vecteur $(F_1(z), F_2(z))$ pour tout $z \in \mathcal{D}_E^k$, alors $\mathcal{D}_E^k = \mathcal{D}_E^k \cup \{\bar{z}\}$, $\bar{\mathcal{D}}_E^k = \bar{\mathcal{D}}_E^k \setminus \{\bar{z}\}$, aller à l'étape 5.

Etape 5 : Déterminer l'ensemble N_k correspondant à la solution entière $z^{*(k)}$ obtenue par l'algorithme 2. Une coupe de Dantzig (5.29) est alors ajoutée au domaine correspondant à la solution entière $z^{*(k)}$. $\mathcal{D}_E = \mathcal{D}_E^k$, $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E^k$, $k = k + 1$, aller à l'étape 2.

A chaque itération de l'algorithme 3, le premier niveau est résolu à l'aide de l'algorithme 2 (qui utilise l'Algorithme 1), puis la solution obtenue est testée pour l'optimalité du problème principal. Si cette solution n'est pas optimale, on l'élimine du domaine et on continue l'exploration. Si une solution optimale est obtenue pour le problème principal ou si le premier niveau n'a pas de solution, l'algorithme s'arrête.

Théorème 5.4.1 *L'algorithme proposé fournit une solution entière optimale du programme IIQBP en un nombre fini d'itérations, si une telle solution existe.*

Preuve 5.4.1 . Soit $\mathcal{D}$ l'ensemble des solutions entières réalisables du problème IIQBP, un ensemble fini borné contenu dans S . La cardinalité de l'ensemble des solutions entières réalisables $\mathcal{D}$ est un nombre fini, puisqu'il contient un nombre fini de solutions entières. Afin qu'à chaque étape k de l'algorithme 3, si une solution entière $z^*(k)$ n'est pas optimale du problème principal, la coupe de Dantzig (5.29) est construite pour éliminer $z^*(k)$ puis un nouveau nœud est déterminé. Ainsi, dans le cas où l'ensemble S_k ne contient aucune solution entière optimale, le sous-arbre ayant pour racine le nœud k est exploré et tous les nœuds sont sondés après un nombre fini d'étapes. Ainsi, l'arborescence de recherche aura un nombre fini de branches et l'algorithme se terminera en un nombre fini d'étapes.

La complexité de l'algorithme 3, basé sur la procédure de Branch & Bound avec la technique des coupes, est exponentielle. En raison du fait que, les problèmes d'optimisation en nombres entiers sont NP-hard Schrijver A. [118]. De plus, les problèmes de programmation à deux niveaux sont NP-hard Dempe S. [50].

5.5 Un exemple illustratif :

Pour illustrer le déroulement de l'algorithme 3, considérons le problème suivant :

5.5.1 Exemple1

$$IIQBP \left\{ \begin{array}{l} \max_{x_1} F(x_1, y_1, y_2) = (3x_1 - y_1 + 2)(-x_1 + y_1 - y_2 + 4) \\ \text{où } (y_1, y_2) \text{ résout} \\ \left\{ \begin{array}{l} \max f(x_1, y_1, y_2) = (x_1 + y_1 + 5y_2 - 1)(x_1 + y_1 - 3y_2 + 4) \\ s.c \\ 4x_1 + 3y_1 + 4y_2 \leq 20 \\ x_1 + 2y_1 - y_2 \leq 3 \\ 3x_1 + y_1 - y_2 \leq 4 \\ x_1 \geq 0, y_1 \geq 0, y_2 \geq 0, \text{ entiers} \end{array} \right. \end{array} \right. \quad (5.30)$$

Algorithm 3 (Algorithme "Branch & cut" pour la résolution du problème $IIQBP$.) :

Etape 1 : $k = 0$, $\mathcal{D}_E^0 = \emptyset$, $\bar{\mathcal{D}}_E^0 = \emptyset$. Résoudre le problème du premier niveau $UIIQP_0$.

Utiliser l'algorithme 2 ci-dessus :

Pour résoudre le problème du premier niveau $UIIQP_0$, soit le problème linéaire bicritère correspondant au problème du premier niveau défini comme suit :

$$BLP \left\{ \begin{array}{l} \max F_1(x_1, y_1, y_2) = (3x_1 - y_1 + 2) \\ \max F_2(x_1, y_1, y_2) = (-x_1 + y_1 - y_2 + 4) \\ s.c \\ 4x_1 + 3y_1 + 4y_2 \leq 20 \\ x_1 + 2y_1 - y_2 \leq 3 \\ 3x_1 + y_1 - y_2 \leq 4 \\ x_1 \geq 0, y_1 \geq 0, y_2 \geq 0, \text{ entiers} \end{array} \right. \quad (5.31)$$

Pour $(y_1, y_2) = (x_2, x_3)$, le problème *BLP* peut être réécrit comme suit :

$$BLP \left\{ \begin{array}{ll} \max & F_1(x_1, x_2, x_3) = (3x_1 - x_2 + 2) \\ \max & F_2(x_1, x_2, x_3) = (-x_1 + x_2 - x_3 + 4) \\ s.c & \\ 4x_1 + 3x_2 + 4x_3 & \leq 20 \\ x_1 + 2x_2 - x_3 & \leq 3 \\ 3x_1 + x_2 - x_3 & \leq 4 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, & \text{entiers} \end{array} \right. \quad (5.32)$$

$l = 0$, $\mathcal{D}_E = \emptyset$, $\bar{\mathcal{D}}_E = \emptyset$, $F_{opt} = -\infty$, $f_{opt} = -\infty$ et $z_{opt} = \emptyset$. En résolvant le programme correspondant LP_0 on obtient le tableau 1 :

La solution optimale est $z^{*(0)} = (x_1^{*(0)}, x_2^{*(0)}, x_3^{*(0)}) = (9/4, 0, 11/4)$.

TABLE 1 – Tableau du simplexe au noeud 0.

B	x_2	x_4	x_6	RHS
x_3	5/16	3/16	-1/4	11/4
x_5	15/8	1/8	-1/2	7/2
x_1	7/16	1/16	1/4	9/4
$\bar{c}_j$	-37/16	-3/16	-3/4	$F_1 = 35/4$
$\bar{d}_j$	7/4	1/4	0	$F_2 = -1$

qui n'est pas entière. Donc, deux noeuds sont générés $node_1$ et $node_2$ correspondant aux contraintes suivantes :

$node_1 : x_1 \leq 2$.

$node_2 : x_1 \geq 3$.

$node_1$: la contrainte $x_1 \leq 2$ est ajoutée aux contraintes du problème LP_0 , on obtient le problème LP_1 , il est résolu, la solution optimale entière correspondante $z^{*(1)} = (2, 0, 2)$, est donnée dans la Table 2, $\mathcal{D}_E = \{(2, 0, 2)\}$, $\bar{\mathcal{D}}_E = \emptyset$.

$H_1 = \{7\}$. Appliquer la coupe efficace $x_7 \geq 1$, et résoudre le nouveau problème LP_3 , la solution optimale $z^{*(3)} = (1, 0, 0)$ est donnée dans la Table 3 :

TABLE 2 – Tableau du simplexe au noeud 1.

B	x_2	x_6	x_7	RHS
x_3	-1	-1	3	2
x_5	1	-1	2	3
x_1	0	0	1	2
x_4	7	4	-16	4
$\bar{c}_j$	-1	0	-3	$F_1 = 8$
$\bar{d}_j$	0	-1	4	$F_2 = 0$

TABLE 3 – Tableau du simplexe au noeud 3.

B	x_2	x_3	x_8	RHS
x_6	1	-1	-3	1
x_5	2	-1	-1	2
x_1	0	0	1	1
x_4	3	4	-4	16
x_7	0	0	-1	1
$\bar{c}_j$	-1	0	-3	$F_1 = 5$
$\bar{d}_j$	1	-1	1	$F_2 = 3$

Le vecteur critère est $(5, 3)$, donc, $\mathcal{D}_E = \{(2, 0, 2), (1, 0, 0)\}$ est mis à jour, $\bar{\mathcal{D}}_E = \emptyset$. $H_2 = \{2, 8\}$, le domaine réalisable de LP_3 est réduit avec l'utilisation de la coupe efficace $x_2 + x_8 \geq 1$. La solution optimale entière du problème LP_4 est $z^{*(4)} = (1, 1, 0)$ comme indiqué dans la Table 4 :

le vecteur critère est $(4, 4)$, cependant $\mathcal{D}_E$ est mis à jour, $\mathcal{D}_E = \{(2, 0, 2), (1, 0, 0), (1, 1, 0)\}$, $\bar{\mathcal{D}}_E = \emptyset$. $H_3 = \{9\}$. En ajoutant la contrainte de la coupe efficace $x_9 \geq 1$, la Table 5 donne la solution optimale entière $z^{*(5)} = (1, 2, 2)$ du nouveau problème LP_5 avec le vecteur critère $(3, 3)$. Comme le vecteur correspondant à cette solution est dominé, alors, $\mathcal{D}_E = \{(2, 0, 2), (1, 0, 0), (1, 1, 0)\}$, $\bar{\mathcal{D}}_E = \{(1, 2, 2)\}$.

TABLE 4 – Tableau du simplexe au noeud 4.

B	x_3	x_8	x_9	RHS
x_6	-1	-4	1	0
x_5	-1	-3	2	0
x_1	0	1	0	1
x_4	4	-7	3	13
x_7	0	-1	0	1
x_2	0	1	-1	1
$\bar{c}_j$	0	-2	-1	$F_1 = 4$
$\bar{d}_j$	-1	0	1	$F_2 = 4$

TABLE 5 – Tableau du simplexe au noeud 5.

B	x_5	x_8	x_{10}	RHS
x_6	-1	-1	-1	1
x_3	-1	3	-2	2
x_1	0	1	0	1
x_4	4	-19	11	2
x_7	0	-1	0	1
x_2	0	1	-1	2
x_9	0	0	-1	1
$\bar{c}_j$	0	-2	-1	$F_1 = 3$
$\bar{d}_j$	-1	3	-1	$F_2 = 3$

$H_4 = \{8\}$. Après l'ajout de la coupe efficace $x_8 \geq 1$ au problème LP_4 on obtient le problème LP_6 . La solution optimale entière $z^{*(6)} = (0, 1, 0)$ est donnée dans la Table 6 : le vecteur critère correspondant est $(1, 5)$, donc, $\mathcal{D}_E = \{(2, 0, 2), (1, 0, 0), (1, 1, 0), (0, 1, 0)\}$ est mis à jour, $\bar{\mathcal{D}}_E = \{(1, 2, 2)\}$. $H_5 = \{10\}$. On ajoute la contrainte de coupe efficace $x_{10} \geq 1$, on obtient le problème LP_7 .

Table 7 indique la solution optimale $z^{*(7)} = (0, 2, 1)$: qui est dominée, le vecteur critère est $(1, 5)$, d'où, $\mathcal{D}_E = \{(2, 0, 2), (1, 0, 0), (1, 1, 0), (0, 1, 0)\}$, $\bar{\mathcal{D}}_E = \{(1, 2, 2), (0, 2, 1)\}$. $H_5 = \{11\}$.

TABLE 6 – Tableau du simplexe au noeud 6.

B	x_3	x_{10}	x_{11}	RHS
x_6	-1	1	-4	3
x_5	-1	2	-3	1
x_1	0	0	1	0
x_4	4	3	-7	17
x_7	0	0	-1	2
x_2	0	-1	1	1
x_9	0	-1	0	1
x_8	0	0	-1	1
$\bar{c}_j$	0	-1	-2	$F_1 = 1$
$\bar{d}_j$	-1	1	0	$F_2 = 5$

TABLE 7 – Tableau du simplexe au noeud 7.

B	x_5	x_{11}	x_{12}	RHS
x_6	-1	-1	-1	3
x_3	-1	3	-2	1
x_1	0	1	0	0
x_4	4	-19	11	10
x_7	0	-1	0	2
x_2	0	1	-1	2
x_9	0	0	-1	2
x_8	0	-1	0	1
x_{10}	0	0	-1	1
$\bar{c}_j$	0	-2	-1	$F_1 = 0$
$\bar{d}_j$	-1	3	-1	$F_2 = 5$

La coupe efficace $x_{11} \geq 1$ est ajoutée à LP_7 donne un problème augmenté LP_8 irréalisable, alors le nœud est sondé.

$node_2$: la contrainte $x_1 \geq 3$ est ajoutée à LP_0 , donne un problème augmenté irréalisable, alors le nœud est sondé.

La solution efficace $z \in \mathcal{D}_E$, qui maximise la valeur de la fonction objectif du problème $UIIQP_0$, $F(z)$, ($F(z) = F_1(z) * F_2(z)$) est $z^{*(5)} = (1, 1, 0)$ alors $(1, 1, 0)$ est la solution optimale du premier niveau $UIIQP_0$ et $F_{opt} = 16$. $\mathcal{D}_E^0 = \mathcal{D}_E$, $\bar{\mathcal{D}}_E^0 = \bar{\mathcal{D}}_E$, allez à l'étape 3.

Etape 3 : On pose $x_1 = x_1^{*(5)}$ et on résout le problème du second niveau en utilisant la même technique que celle citée dans l'algorithme 2 ci-dessus. La solution optimale est $(\hat{x}_2, \hat{x}_3) = (1, 1) \neq (1, 0) = (x_2^5, x_3^5)$, alors $z^{*(5)} = (1, 1, 0)$ n'est pas optimale pour le problème principal (5.2), passez à l'étape 4.

Etape 4 : $\mathcal{D}_E^0$ et $\bar{\mathcal{D}}_E^0$ sont mis à jour .
 $\mathcal{D}_E^1 = \mathcal{D}_E^0 \setminus \{z^{*(5)}\}$ et $\bar{\mathcal{D}}_E^1 = \bar{\mathcal{D}}_E^0$.

Etape 5 : $N_0 = \{3, 8, 9\}$, Nous réduisons le domaine réalisable de $UIIQP_0$ en utilisant la coupe $x_3 + x_8 + x_9 \geq 1$, on obtient le nouveau problème quadratique indéfini en nombres entiers $UIIQP_1$ et le résoudre.

Etape 2 : Utiliser l'algorithme 2 ci-dessus pour résoudre le problème $UIIQP_1$. Pour cela, ajouter la coupe $x_3 + x_8 + x_9 \geq 1$ au programme LP_5 et utiliser la méthode dual de simplexe pour résoudre le nouveau problème LP_8 . La solution optimale $z^{*(8)} = (1, 1, 1)$ est donnée par Table 8 :

TABLE 8 – Tableau du simplexe au noeud 8.

B	x_8	x_9	x_{10}	RHS
x_6	-3	2	-1	1
x_5	-2	3	-1	1
x_1	1	0	0	1
x_4	-11	-1	4	9
x_7	-1	0	0	1
x_2	1	-1	0	1
x_3	1	1	-1	1
$\bar{c}_j$	-2	-1	0	$F_1 = 4$
$\bar{d}_j$	1	2	-1	$F_2 = 3$

Le vecteur correspondant est dominé, alors $\mathcal{D}_E^1 = \mathcal{D}_E^1$,

$\bar{\mathcal{D}}_E^1 = \bar{\mathcal{D}}_E^1 \cup \{(1, 1, 1)\}$. $H_6 = \{8, 9\}$. La coupe efficace est $x_8 + x_9 \geq 1$, on obtient le nouveau problème LP_9 . La solution optimale $z^{*(9)} = (1, 2, 2)$ est indiquée dans la Table 9 :

TABLE 9 – Tableau du simplexe au noeud 9.

B	x_5	x_8	x_{11}	RHS
x_6	-1	0	-1	1
x_{10}	-1	5	-3	2
x_1	0	1	0	1
x_4	4	-30	11	2
x_7	0	-1	0	1
x_2	0	2	-1	2
x_3	-1	5	-2	2
x_9	0	1	-1	1
$\bar{c}_j$	0	-1	-1	$F_1 = 3$
$\bar{d}_j$	-1	4	-1	$F_2 = 3$

cette solution n'est pas efficace, alors $\mathcal{D}_E^1 = \mathcal{D}_E^1$, $\bar{\mathcal{D}}_E^1 = \bar{\mathcal{D}}_E^1 \cup \{(1, 2, 2)\}$. $H_7 = \{8\}$, la coupe efficace $x_8 \geq 1$, on obtient le nouveau problème LP_{10} . La solution optimale $z^{*(10)} = (0, 0, 0)$ est indiquée dans la Table 10 : avec le vecteur critère $(2, 4)$. Comme la solution est efficace, donc, $\mathcal{D}_E^1 = \mathcal{D}_E^1 \cup \{(0, 0, 0)\}$, $\bar{\mathcal{D}}_E^1 = \bar{\mathcal{D}}_E^1$. $H_8 = \{10\}$. La coupe efficace est $x_{11} \geq 1$, le nouveau problème est LP_{11} et la solution optimale est $z^{*(11)} = (0, 1, 0)$, voir la Table 11 :

le vecteur critère est $(1, 5)$. $\mathcal{D}_E^1 = \mathcal{D}_E^1 \cup \{(0, 1, 0)\}$, $\bar{\mathcal{D}}_E^1 = \bar{\mathcal{D}}_E^1$.

$H_9 = \{13\}$. Appliquer la coupe efficace $x_{13} \geq 1$, on obtient le problème LP_{12} . En utilisant la méthode dual de simplexe, une solution optimale $z^{*(12)} = (0, 2, 1)$ est déterminée dans la Table 12 :

TABLE 10 – Tableau du simplexe au noeud 10.

B	x_{10}	x_{11}	x_{12}	RHS
x_6	-1	2	-5	4
x_5	-1	3	-5	3
x_1	0	0	1	0
x_4	4	-1	-10	20
x_7	0	0	-1	2
x_2	0	-1	2	0
x_3	-1	1	0	0
x_9	0	-1	1	0
x_8	0	0	-1	1
$\bar{c}_j$	0	-1	-1	$F_1 = 2$
$\bar{d}_j$	-1	2	-1	$F_2 = 4$

TABLE 11 – Tableau du simplexe au noeud 11.

B	x_3	x_{12}	x_{13}	RHS
x_6	-1	-5	1	3
x_5	-1	-5	2	1
x_1	0	1	0	0
x_4	4	-10	3	17
x_7	0	-1	0	2
x_2	0	2	-1	1
x_{10}	-1	0	-1	1
x_9	0	1	-11	1
x_8	0	-1	0	1
x_{11}	0	0	-1	1
$\bar{c}_j$	0	-1	-1	$F_1 = 1$
$\bar{d}_j$	-1	-1	1	$F_2 = 5$

le vecteur correspondant à cette solution est dominé alors $\mathcal{D}_E^1 = \mathcal{D}_E^1$, $\bar{\mathcal{D}}_E^1 = \bar{\mathcal{D}}_E^1 \cup \{(0, 2, 1)\}$.

TABLE 12 – Tableau du simplexe au noeud 12.

B	x_5	x_{12}	x_{14}	RHS
x_6	-1	0	-1	3
x_3	-1	5	-2	1
x_1	0	1	0	0
x_4	4	-30	11	10
x_7	0	-1	0	2
x_2	0	2	-1	2
x_{10}	-1	5	-3	3
x_9	0	1	-1	2
x_8	0	-1	0	1
x_{11}	0	0	-1	2
x_{13}	0	0	-1	1
$\bar{c}_j$	0	-1	-1	$F_1 = 0$
$\bar{d}_j$	-1	4	-1	$F_2 = 5$

$H_{10} = \{12\}$, la coupe efficace $x_{12} \geq 1$ le problème correspondant est LP_{13} est irréalisable, donc, le noeud est sondé. La solution optimale du problème $UIIQP_1$ est $z^{*(4)} = (1, 0, 0)$, $F_{opt} = 15$.

Etape 3 : Pour $x_1 = x_1^{*(4)}$ le problème du second niveau est résolu par la même technique citée dans l'algorithme 2 ci-dessus.

La solution optimale est $(\tilde{x}_2, \tilde{x}_3) = (1, 1) \neq (0, 0) = (x_2^4, x_3^4)$, alors $z^{*(4)} = (1, 0, 0)$ qui n'est pas optimale pour le problème principal (5.2), aller à l'étape 4.

Etape 4 : La mise à jour de $\mathcal{D}_E^1$ et l'ensemble $\bar{\mathcal{D}}_E^1$.
 $\mathcal{D}_E^2 = \mathcal{D}_E^1 \setminus \{z^{*(4)}\}$ et $\bar{\mathcal{D}}_E^2 = \bar{\mathcal{D}}_E^1$. Il existe une solution $(1, 1, 1) \in \bar{\mathcal{D}}_E^2$ qui a pour le vecteur critère $(4, 3)$ avec le vecteur correspondant est non dominé, alors $\mathcal{D}_E^2 = \mathcal{D}_E^2 \cup \{(1, 1, 1)\}$ et $\bar{\mathcal{D}}_E^2 = \bar{\mathcal{D}}_E^2 \setminus \{(1, 1, 1)\}$, aller à l'étape 5.

Etape 5 : $N_1 = \{2, 3, 8\}$, le domaine réalisable du problème $UIIQP_1$ est réduit en rajoutant la coupe $x_2 + x_3 + x_8 \geq 1$, on obtient le problème $UIIQP_2$ et le résoudre.

Etape 2 : Le problème $UIIQP_2$ est résolu en utilisant l'algorithme 2 ci-dessus . Pour cela, on ajoute la coupe $x_2 + x_3 + x_8 \geq 1$ au programme LP_4 et on utilise la méthode dual de simplexe pour résoudre le nouveau problème LP_{14} . La solution optimale $z^{*(14)} =$

$(1, 0, 1)$ est donnée par le tableau 13 : le vecteur critère est $(5, 2)$. Cette solution existe

TABLE 13 – Tableau du simplexe au noeud 14.

B	x_2	x_8	x_9	RHS
x_6	2	-2	-1	2
x_5	3	0	-1	3
x_1	0	1	0	1
x_4	-1	-8	4	12
x_7	0	-1	0	1
x_3	1	1	-1	1
$\bar{c}_j$	-1	-3	0	$F_1 = 5$
$\bar{d}_j$	2	2	-1	$F_2 = 2$

déjà dans $\mathcal{D}_E^2$, alors $\mathcal{D}_E^2 = \mathcal{D}_E^2$, $\bar{\mathcal{D}}_E^2 = \bar{\mathcal{D}}_E^2$. $H_{11} = \{2, 8\}$. La coupe efficace $x_2 + x_9 \geq 1$, le nouveau programme est LP_{15} avec la solution optimale $z^{*(15)} = (1, 1, 0)$, $F_{opt} < F(z^{*(15)})$, alors le noeud est sondé. La solution optimale du problème $UIIQP_2$ est $z^{*(10)} = (1, 1, 1)$, $F_{opt} = 12$.

Etape 3 : Pour $x_1 = x_1^{*(10)}$, résoudre le problème du second niveau par la même technique citée dans l'algorithme 2 ci-dessus. La solution optimale est $(\tilde{x}_2, \tilde{x}_3) = (1, 1) = (1, 1) = (x_2^8, x_3^8)$, alors $z^{*(8)} = (1, 1, 1)$ est la solution optimale du problème quadratique indéfini à deux niveaux en nombres entiers. (5.2), $F_{opt} = 12$ et $f_{opt} = 18$.

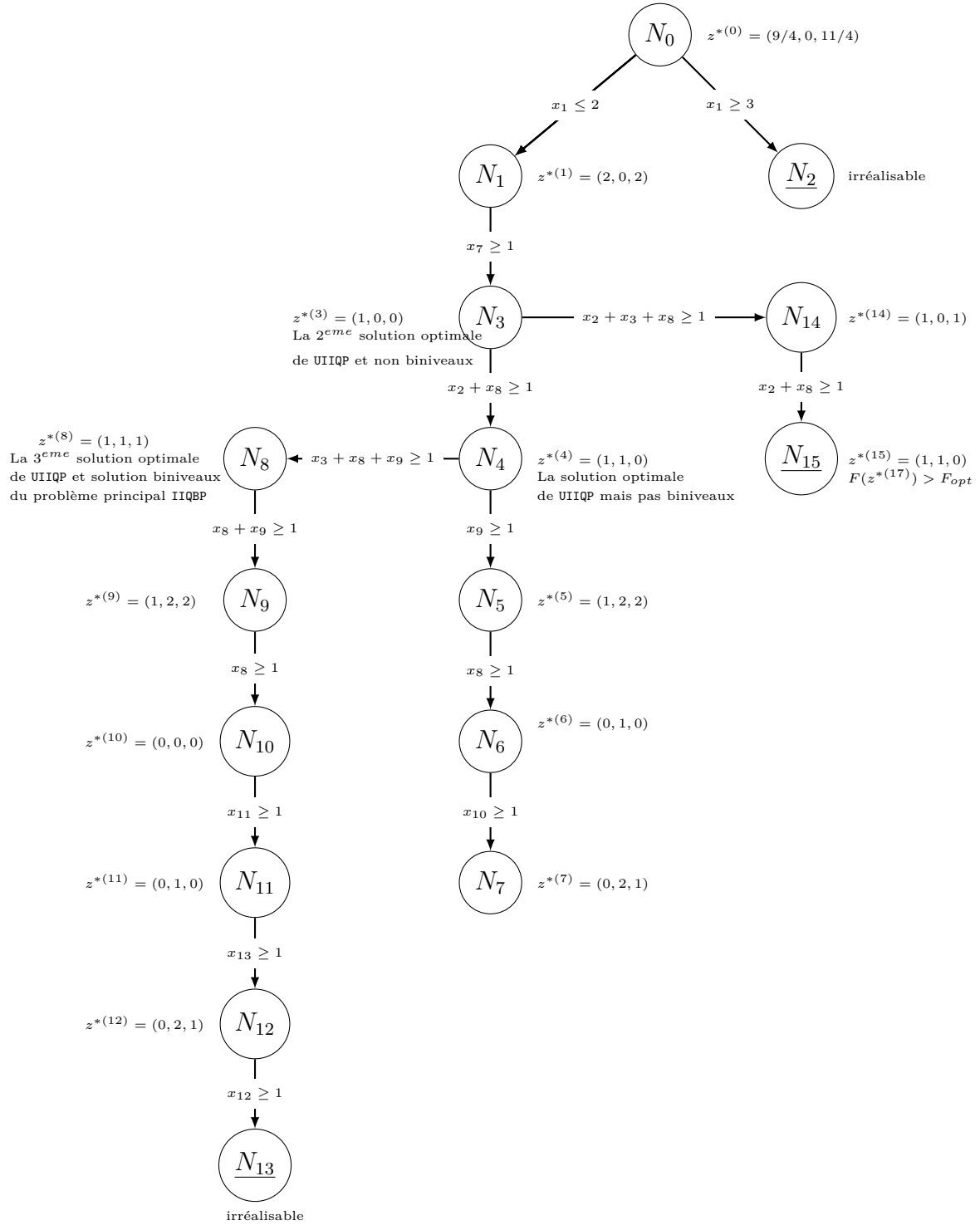


FIGURE 5.2 – Arbre représentant les états des nœuds dans l'algorithme Branch and Cut pour la résolution du problème IIQBP.

5.5.2 Exemple2

$$IIQBP \left\{ \begin{array}{l} \max_{x_1} F(x_1, y_1, y_2) = (1/2x_1 + 5y_1 - 1/2y_2 + 3)(-1/3y_1 + 3y_2 + 5) \\ \text{où } (y_1, y_2) \text{ résout} \\ \left\{ \begin{array}{l} \max f(x_1, y_1, y_2) = (-x_1 + 2y_1 - y_2 + 19/2)(-8y_1 + 15y_2) \\ s.c \\ x_1 - 2y_1 \leq 10 \\ x_1 + 3y_1 \leq 16 \\ 2y_1 + 5y_2 \leq 12 \\ x_1 \geq 0, y_1 \geq 0, y_2 \geq 0, \text{entiers} \end{array} \right. \end{array} \right. \quad (5.33)$$

Algorithm 3 (L'algorithme de "Branch & cut" pour résoudre le problème $IIQBP$) :

Etape 1 : $k = 0$, $\mathcal{D}_E^0 = \emptyset$, $\bar{\mathcal{D}}_E^0 = \emptyset$.

Résoudre le problème du premier niveau $UIIQP_0$.

Utiliser l'algorithme 2 :

Pour résoudre le problème du premier niveau $UIIQP_0$, soit le problème linéaire bicritère correspondant au problème du premier niveau est défini comme suit :

$$BLP \left\{ \begin{array}{l} \max F_1(x_1, y_1, y_2) = (1/2x_1 + 5y_1 - 1/2y_2 + 3) \\ \max F_1(x_1, y_1, y_2) = (-1/3y_1 + 3y_2 + 5) \\ s.c \\ x_1 - 2y_1 \leq 10 \\ x_1 + 3y_1 \leq 16 \\ 2y_1 + 5y_2 \leq 12 \\ x_1 \geq 0, y_1 \geq 0, y_2 \geq 0, \text{entiers} \end{array} \right. \quad (5.34)$$

Pour $(y_1, y_2) = (x_2, x_3)$, le problème BLP peut s'écrire comme suit :

$$BLP \left\{ \begin{array}{l} \max F_1(x_1, x_2, x_3) = (1/2x_1 + 5x_2 - 1/3x_3 + 3) \\ \max F_1(x_1, x_2, x_3) = (-1/3x_2 + 3x_3 + 5) \\ s.c \\ x_1 - 2x_2 \leq 10 \\ x_1 + 3x_2 \leq 16 \\ 2x_2 + 5x_3 \leq 12 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, \text{integers} \end{array} \right. \quad (5.35)$$

$l = 0$, $\mathcal{D}_E = \emptyset$, $\bar{\mathcal{D}}_E = \emptyset$. Le problème correspondant LP_0 est résolu. La première solution

optimale est : $z^{*(0)} = (x_1^{*(0)}, x_2^{*(0)}, x_3^{*(0)}) = (0, 16/3, 0)$, qui n'est pas entière, donnée dans la Table 1 :

TABLE 1 –

B	x_1	x_3	x_5	RHS
x_4	$5/3$	0	$2/3$	$62/3$
x_2	$1/3$	0	$1/3$	$16/3$
x_6	$-2/3$	5	$-2/3$	$4/3$
x_7	$7/6$	$1/2$	$5/3$	$89/3$
x_8	$-1/9$	-3	$-1/9$	$29/9$
$\bar{c}_j$	$-7/6$	$-1/2$	$-5/3$	$F_1 = 89/3$
$\bar{d}_j$	$1/9$	3	$1/9$	$F_2 = 29/9$

Comme cette solution optimale n'est pas entière, le noeud est partitionné en deux noeuds : $noeud_1$, $noeud_2$. correspondant aux contraintes suivantes :

$node_1 : x_2 \leq 5$.

$node_2 : x_2 \geq 6$.

$node_1$: la contrainte $x_2 \leq 5$ est ajoutée au problème LP_0 , nous obtenons LP_1 qui est une solution optimale entière $z^{*(1)} = (x_1^{*(1)}, x_2^{*(1)}, x_3^{*(1)}) = (1, 5, 0)$ donnée dans la Table 2, le vecteur critère est $(89/3, 29/9)$. $\mathcal{D}_E = \{(1, 5, 0)\}$, $\bar{\mathcal{D}}_E = \emptyset$. $H_1 = \{3, 9\}$.

TABLE 2 –

B	x_3	x_5	x_9	RHS
x_4	0	-1	5	19
x_2	0	0	1	5
x_6	5	0	-2	2
x_7	$1/2$	$1/2$	$7/2$	29
x_8	-3	0	$-1/3$	3
x_1	0	1	-3	1
$\bar{c}_j$	$-1/2$	$-1/2$	$-7/2$	$F_1 = 57/2$
$\bar{d}_j$	3	0	$1/3$	$F_2 = 10/3$

Le domaine réalisable LP_1 est réduit, après l'utilisation de la coupe efficace

$x_3 + x_9 \geq 1$, on obtient le problème LP_2 , la solution optimale est donnée dans la Table 3 :

TABLE 3 –

B	x_5	x_6	x_{10}	RHS
x_4	-1	$5/7$	$25/7$	$118/7$
x_2	0	$1/7$	$5/7$	$32/7$
x_9	0	$-1/7$	$-5/7$	$3/7$
x_7	$1/2$	$3/7$	$37/14$	$381/14$
x_8	0	$8/21$	$-23/21$	$34/7$
x_1	1	$-3/7$	$-15/7$	$16/7$
x_3	0	$1/7$	$-2/7$	$4/7$
$\bar{c}_j$	$-1/2$	$-3/7$	$-37/14$	$F_1 = 187/7$
$\bar{d}_j$	0	$-8/21$	$23/21$	$F_2 = 109/21$

$z^{*(2)} = (16/7, 32/7, 4/7)$, qui n'est pas entière. Le branchement appliqué.

$node_5 : x_1 \leq 2$.

$node : x_1 \geq 3$.

$node_5$: la contrainte $x_1 \leq 2$ est ajoutée aux contraintes du problème LP_2 , on obtient le problème LP_3 la solution optimale est $z^{*(3)} = (2, 32/7, 4/7)$ voir la Table 4 :

TABLE 4 –

B	x_6	x_{10}	x_{11}	RHS
x_4	$2/7$	$10/7$	-1	$120/7$
x_2	$1/7$	$5/7$	0	$32/7$
x_9	$-1/7$	$-5/7$	0	$3/7$
x_7	$9/14$	$26/7$	$1/2$	$379/14$
x_8	$8/21$	$-23/21$	0	$34/7$
x_1	0	0	1	2
x_3	$1/7$	$-2/7$	0	$4/7$
x_5	$-3/7$	$-15/7$	-1	$2/7$
$\bar{c}_j$	$-9/14$	$-26/7$	$-1/2$	$F_1 = 186/7$
$\bar{d}_j$	$-8/21$	$23/21$	0	$F_2 = 109/21$

qui n'est pas entière. Donc, on applique le branchement.

$node_7 : x_2 \leq 4$.

$node_8 : x_2 \geq 5$.

$node_7$: la contrainte $x_2 \leq 4$ est appliquée au problème LP_3 , le problème obtenu est LP_4 avec la solution optimale $z^{*(4)} = (2, 4, 0)$, donné dans la Table 5 :

TABLE 5 –

B	x_{10}	x_{11}	x_{12}	RHS
x_4	0	-1	2	16
x_2	0	0	1	4
x_9	0	0	-1	1
x_7	1/2	1/2	9/2	25
x_8	-3	0	8/3	3
x_1	0	1	0	2
x_3	-1	0	1	0
x_5	0	-1	-3	2
x_6	5	0	-7	4
$\bar{c}_j$	-1/2	-1/2	-9/2	$F_1 = 24$
$\bar{d}_j$	3	0	-8/3	$F_2 = 11/3$

le vecteur critère est $(24, 11/3)$. Donc, $\mathcal{D}_E = \{(1, 5, 0), (2, 4, 0)\}$, $\bar{\mathcal{D}}_E = \emptyset$.

De la même manière, en appliquant la coupe efficace (5.14), nous déterminons l'ensemble des solutions efficaces, $\mathcal{D}_E = \{(1, 5, 0), (4, 4, 0), (7, 3, 0), (7, 3, 1), (10, 2, 1), (10, 0, 2), (12, 1, 1), (12, 1, 2)\}$. Ces solutions sont données respectivement dans les Tables (6) à (12) :

TABLE 6 –

B	x_5	x_{10}	x_{12}	RHS
x_4	-1	0	5	14
x_2	0	0	1	4
x_9	0	0	-1	1
x_7	$1/2$	$1/2$	3	26
x_8	0	-3	$8/3$	3
x_1	1	0	-3	4
x_3	0	-1	1	0
x_6	0	5	-7	4
x_{11}	1	0	-3	1
$\bar{c}_j$	$-1/2$	$-1/2$	-3	$F_1 = 25$
$\bar{d}_j$	0	3	$-8/3$	$F_2 = 11/3$

TABLE 7 –

B	x_3	x_{13}	x_{15}	RHS
x_4	1	-1	6	9
x_2	0	0	1	3
x_9	0	0	-1	2
x_7	0	$1/2$	3	23
x_{14}	-1	1	-4	3
x_1	-1	1	-4	7
x_8	-3	0	$-1/3$	3
x_5	1	-1	1	0
x_{11}	-1	1	-4	4
x_{10}	-1	0	-1	1
x_{12}	0	0	-1	1
x_6	5	0	-2	6
$\bar{c}_j$	0	$-1/2$	-3	$F_1 = 43/2$
$\bar{d}_j$	3	0	$1/3$	$F_2 = 4$

TABLE 8 –

B	x_5	x_{15}	x_{16}	RHS
x_4	-1	5	0	9
x_2	0	1	0	3
x_9	0	-1	0	2
x_7	$1/2$	3	$1/2$	23
x_{14}	1	-3	0	3
x_1	1	-3	0	7
x_8	0	$8/3$	-3	6
x_{13}	-1	0	-1	1
x_{11}	1	-3	0	4
x_{10}	0	0	-1	2
x_{12}	0	-1	0	1
x_6	0	-7	5	1
x_3	0	1	-1	1
$\bar{c}_j$	$-1/2$	-3	$-1/2$	$F_1 = 21$
$\bar{d}_j$	0	$-8/3$	3	$F_2 = 7$

TABLE 9 –

B	x_5	x_{17}	x_{19}	RHS
x_4	-1	0	5	4
x_2	0	0	1	2
x_9	0	0	-1	3
x_7	$1/2$	$1/2$	3	20
x_{14}	1	0	-3	6
x_1	1	0	-3	10
x_8	0	-3	$8/3$	6
x_{13}	-1	-1	0	2
x_{11}	1	0	-3	7
x_{10}	0	-1	0	3
x_{12}	0	0	-1	2
x_{15}	0	0	-1	1
x_3	0	-1	1	1
x_{16}	0	-1	0	1
x_6	0	5	-7	3
x_{18}	1	0	-3	1
$\bar{c}_j$	$-1/2$	$-1/2$	-3	$F_1 = 35/2$
$\bar{d}_j$	0	3	$-8/3$	$F_2 = 22/3$

TABLE 10 –

B	x_{22}	x_{25}	x_{26}	RHS
x_4	-1	0	2	0
x_2	0	0	1	0
x_9	0	0	-1	5
x_7	$1/2$	$1/2$	$9/2$	11
x_{14}	1	0	0	6
x_1	1	0	0	10
x_8	0	-3	$8/3$	9
x_{13}	-1	-1	-3	11
x_{11}	1	0	0	7
x_{10}	0	-1	0	6
x_{12}	0	0	-1	4
x_{15}	0	0	-1	3
x_3	0	-1	1	2
x_{16}	0	-1	0	4
x_{19}	0	0	-1	2
x_{18}	1	0	0	1
x_5	-1	0	-3	6
x_{17}	0	-1	0	3
x_{20}	-1	-1	-3	8
x_{23}	0	0	-1	1
x_{21}	0	-1	0	2
x_{24}	0	-1	0	1
x_6	0	5	-7	2
$\bar{c}_j$	$-1/2$	$-1/2$	$-9/2$	$F_1 = 7$
$\bar{d}_j$	0	3	$-8/3$	$F_2 = 11$

TABLE 11 –

B	x_4	x_{21}	x_{23}	RHS
x_{20}	-1	-1	-5	1
x_2	0	0	1	1
x_9	0	0	-1	4
x_7	$1/2$	$1/2$	$11/2$	17
x_{14}	1	0	2	8
x_1	1	0	2	12
x_8	0	-3	$8/3$	6
x_{13}	-1	-1	-5	4
x_{11}	1	0	2	9
x_{10}	0	-1	0	4
x_{12}	0	0	-1	3
x_{15}	0	0	-1	2
x_3	0	-1	1	1
x_{16}	0	-1	0	2
x_{19}	0	0	-1	1
x_{18}	1	0	2	3
x_6	0	5	-7	5
x_{17}	0	-1	0	1
x_{22}	1	0	2	1
x_5	-1	0	-5	1
$\bar{c}_j$	$-1/2$	$-1/2$	$-11/2$	$F_1 = 27/2$
$\bar{d}_j$	0	3	$-8/3$	$F_2 = 23/3$

TABLE 12 –

B	x_4	x_{23}	x_{24}	RHS
x_{20}	0	-5	-1	2
x_2	0	1	0	1
x_9	0	-1	0	4
x_7	0	11/2	1/2	17
x_{14}	1	2	0	8
x_1	1	2	0	12
x_8	3	8/3	-3	9
x_{13}	0	-5	-1	5
x_{11}	1	2	0	9
x_{10}	1	0	-1	5
x_{12}	0	-1	0	3
x_{15}	0	-1	0	2
x_3	1	1	-1	2
x_{16}	1	0	-1	3
x_{19}	0	-1	0	1
x_{18}	1	2	0	3
x_6	-5	-7	5	0
x_{17}	1	0	-1	2
x_{22}	1	2	0	1
x_5	-1	-5	0	1
x_{21}	1	0	-1	1
$\bar{c}_j$	0	-11/2	-1/2	$F_1 = 13$
$\bar{d}_j$	-3	-8/3	3	$F_2 = 32/3$

L'ensemble des solutions dominées $\bar{\mathcal{D}}_E = \{(2, 4, 0), (2, 3, 0), (2, 3, 1), (2, 2, 1), (2, 1, 1), (2, 1, 2), (2, 0, 2), (3, 4, 0), (3, 3, 0), (3, 3, 1), (3, 2, 1), (3, 1, 1), (3, 1, 2), (3, 0, 2), (8, 2, 1), (8, 1, 1), (8, 1, 2), (8, 0, 2), (9, 2, 1), (10, 1, 1), (11, 1, 2), \}$.

La solution efficace z^* qui maximise $F(z)$ la valeur de la fonction objectif du problème $UIQP_0$ est $(12, 1, 2)$, où $F(z^*) = 416/3$, alors la solution optimale du problème $UIQP_0$ est $(12, 1, 2)$. $\mathcal{D}_E^0 = \mathcal{D}_E$ et $\bar{\mathcal{D}}_E^0 = \bar{\mathcal{D}}_E$, aller à l'étape 3.

Etape 3 : Soit $z^{*(0)} = (12, 1, 2)$ la solution optimale du problème $UIIQP_0$. Pour $x_1 = x_1^{*(0)}$ résoudre le problème du second niveau (5.27) par la même technique citée dans l'algorithme 2 ci-dessus, le tableau optimal indique que le problème du second niveau (5.27) est irréalisable, aller à l'étape 4.

Etape 4 : Mise à jour de $\mathcal{D}_E^0$ et l'ensemble $\bar{\mathcal{D}}_E^0$.

– $\mathcal{D}_E^1 = \mathcal{D}_E^0 \setminus \{z^{*(0)}\}$ et $\bar{\mathcal{D}}_E^1 = \bar{\mathcal{D}}_E^0$.

– Il existe une solution $(11, 1, 2) \in \bar{\mathcal{D}}_E^0$ tel que le vecteur $(25/2, 32/3)$ est non dominé, alors $\mathcal{D}_E^1 = \mathcal{D}_E^1 \cup \{(11, 1, 2)\}$ et $\bar{\mathcal{D}}_E^1 = \bar{\mathcal{D}}_E^1 \setminus \{(11, 1, 2)\}$, aller à l'étape 5.

Etape 5 : $N_0 = \{4, 23, 24\}$, la coupe $x_4 + x_{23} + x_{24} \geq 1$ est ajoutée au domaine, on obtient le problème $UIIQP_1$, $\mathcal{D}_E = \mathcal{D}_E^1$, $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E^1$, $F_{opt} = 416/3$, $S = S_0$, $k = 1$, aller à l'étape 2.

Etape 2 : Résoudre le problème $UIIQP_1$, en utilisant l'algorithme 2. La solution optimale déterminée est $(11, 1, 1)$, une coupe efficace est ajoutée $x_{23} + x_{24} \geq 1$ fournit une solution efficace entière $(11, 1, 2)$ comme indiqué dans le tableau 13. :

le vecteur critère est $(25/2, 32/3)$. L'ensemble efficace est $\mathcal{D}_E^1 = \{(1, 5, 0), (4, 4, 0), (7, 3, 0), (7, 3, 1), (10, 2, 1), (10, 0, 2), (12, 1, 1), (11, 1, 2)\}$. L'ensemble des solution dominées est $\bar{\mathcal{D}}_E^1 = \{(2, 4, 0), (2, 3, 0), (2, 3, 1), (2, 2, 1), (2, 1, 1), (2, 1, 2), (2, 0, 2), (3, 4, 0), (3, 3, 0), (3, 3, 1), (3, 2, 1), (3, 1, 1), (3, 1, 2), (3, 0, 2), (8, 2, 1), (8, 1, 1), (8, 1, 2), (8, 0, 2), (9, 2, 1), (10, 1, 1), (11, 1, 1)\}$, la solution optimale du problème $UIIQP_1$ est $(11, 1, 2)$, aller à l'étape 3.

Etape 3 : $z^{*(1)} = (11, 1, 2)$, put $x_1 = x_1^{*(1)}$ et résoudre le problème du second niveau (5.27) par la même technique citée dans l'algorithme 2, le problème est irréalisable, aller à l'étape 4.

Etape 4 : $\mathcal{D}_E^2 = \mathcal{D}_E^1 \setminus \{z^{*(1)}\}$ et $\bar{\mathcal{D}}_E^2 = \bar{\mathcal{D}}_E^1$. Il existe une solution $(10, 1, 2) \in \bar{\mathcal{D}}_E^2$ tel que le vecteur $(12, 21/2)$ est non dominé, alors

$\mathcal{D}_E^2 = \mathcal{D}_E^2 \cup \{(10, 1, 2)\}$ et $\bar{\mathcal{D}}_E^2 = \bar{\mathcal{D}}_E^2 \setminus \{(10, 1, 2)\}$, aller à l'étape 5.

Etape 5 : $N_1 = \{6, 23, 26\}$, la coupe $x_6 + x_{23} + x_{25} \geq 1$ est ajoutée au domaine correspondant à la solution entière $z^{*(1)}$, utiliser l'algorithme 2 pour résoudre le nouveau problème $UIIQP_2$, $\mathcal{D}_E = \mathcal{D}_E^2$, $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E^2$, $F_{opt} = 800/6$, $S = S_1$, $k = 2$, aller à l'étape 2.

Etape 2 : Le problème $UIIQP_2$ est irréalisable, le noeud est sondé,
 $k = k + 1$, $\mathcal{D}_E^3 = \mathcal{D}_E^2$, $\bar{\mathcal{D}}_E^3 = \bar{\mathcal{D}}_E^2$, aller à l'étape 3.

Etape 3 : La nouvelle solution optimale entière dans $\mathcal{D}_E^3$ est $z^{*(3)} = (10, 1, 2)$, est indiquée dans la Table 14 :

TABLE 13 –

B	x_6	x_{23}	x_{26}	RHS
x_{20}	0	-4	-1	3
x_2	0	1	0	1
x_9	0	-1	0	4
x_7	0	5	1/2	17
x_{14}	1/5	-2/5	1	7
x_1	1/5	-2/5	1	11
x_8	3/5	-23/15	0	9
x_{13}	0	-4	-1	6
x_{11}	1/5	-2/5	1	8
x_{10}	1/5	-7/5	0	5
x_{12}	0	-1	0	3
x_{15}	0	-1	0	2
x_3	1/5	-2/5	0	2
x_{16}	1/5	-7/5	0	3
x_{19}	0	-1	0	1
x_{18}	1/5	-2/5	1	2
x_{25}	-1/5	12/5	-2	1
x_{17}	1/5	-7/5	0	2
x_{22}	1/5	-2/5	1	0
x_5	-1/5	-13/5	-1	2
x_{21}	1/5	-7/5	0	1
x_4	-1/5	12/5	-1	1
x_{24}	0	1	-1	1
$\bar{c}_j$	0	-5	-1/2	$F_1 = 25/2$
$\bar{d}_j$	-3/5	23/15	0	$F_2 = 32/3$

$x_1 = x_1^{*(3)}$ résoudre le problème du second niveau (5.27) utiliser l'algorithme 2.
La solution optimale est $(\tilde{x}_2, \tilde{x}_3) = (1, 1) \neq (1, 2) = (x_2^3, x_3^3)$, alors $z^{*(3)} = (10, 1, 2)$, n'est pas optimale pour le problème principal (5.2), aller à l'étape 4.

Etape 4 : $\mathcal{D}_E^4 = \mathcal{D}_E^3 \setminus \{z^{*(3)}\}$ et $\bar{\mathcal{D}}_E^4 = \bar{\mathcal{D}}_E^3$. Aller à l'étape 5.

Etape 5 : la coupe $x_{22} + x_{23} + x_{24} \geq 1$ est ajoutée au domaine correspondant à la

TABLE 14 –

B	x_{22}	x_{23}	x_{24}	RHS
x_4	-1	2	0	2
x_2	0	1	0	1
x_9	0	-1	0	4
x_7	1/2	9/2	1/2	16
x_{14}	1	0	0	6
x_1	1	0	0	10
x_8	0	8/3	-3	9
x_{13}	-1	-3	-1	7
x_{11}	1	0	0	7
x_{10}	0	0	-1	5
x_{12}	0	-1	0	3
x_{15}	0	-1	0	2
x_3	0	1	-1	2
x_{16}	0	0	-1	3
x_{19}	0	-1	0	1
x_{18}	1	0	0	1
x_5	-1	-3	0	3
x_{17}	0	0	-1	2
x_{20}	-1	-3	-1	4
x_6	0	-7	5	0
x_{21}	0	0	-1	1
$\bar{c}_j$	-1/2	-9/2	-1/2	$F_1 = 12$
$\bar{d}_j$	0	-8/3	3	$F_2 = 32/3$

solution entière $z^{*(3)}$, on obtient le nouveau problème quadratique indéfini en nombres entiers ($UIIQP_4$), $\mathcal{D}_E = \mathcal{D}_E^4$, $\bar{\mathcal{D}}_E = \bar{\mathcal{D}}_E^4$, $F_{opt} = 384/3$, $S = S_3$, $k = 4$. Aller à l'étape 2.

Etape 2 : Résoudre le problème $UIIQP_4$, on utilise l'algorithme 2. La solution optimale obtenue est $(9, 1, 2)$, le vecteur critère correspondant est $(23/2, 32/3)$. L'ensemble des solutions efficaces est $\mathcal{D}_E^4 = \{(1, 5, 0), (4, 4, 0), (7, 3, 0), (7, 3, 1), (10, 2, 1), (10, 0, 2), (12, 1, 1), (9, 1, 2)\}$. L'ensemble des solutions domonées est $\bar{\mathcal{D}}_E^4 = \{(2, 4, 0), (2, 3, 0), (2, 3, 1), (2, 2, 1), (2, 1, 1), (2, 1, 2), (2, 0, 2), (3, 4, 0),$

$(3, 3, 0), (3, 3, 1), (3, 2, 1), (3, 1, 1), (3, 1, 2), (3, 0, 2), (8, 2, 1), (8, 1, 1), (8, 1, 2), (8, 0, 2),$
 $(9, 2, 1), (10, 1, 1), (11, 1, 1), (9, 0, 2)\}$, aller à l'étape 3.

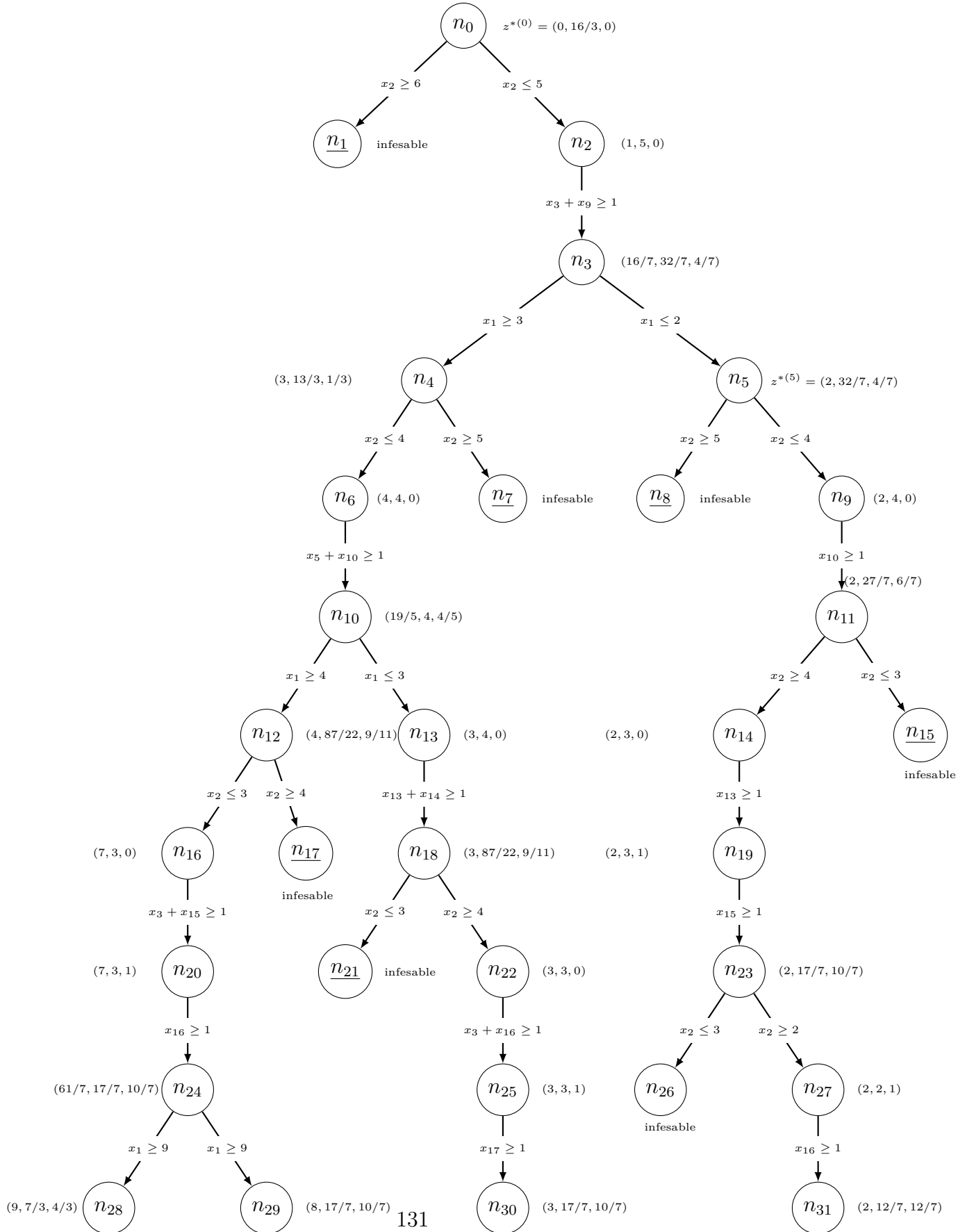
Etape 3 : La solution optimale entière du problème $UIIQP_4$ est
 $z^{*(4)} = (9, 1, 2)$ donnée dans la Table 15 :

TABLE 15 –

B	x_6	x_{23}	x_{25}	RHS
x_4	$-1/5$	$22/5$	-1	3
x_2	0	1	0	1
x_9	0	-1	0	4
x_7	0	4	$1/2$	16
x_{14}	$1/5$	$-12/5$	1	5
x_1	$1/5$	$-12/5$	1	9
x_8	$3/5$	$-23/5$	0	9
x_{13}	0	-2	-1	8
x_{11}	$1/5$	$-12/5$	1	6
x_{10}	$1/5$	$-7/5$	0	5
x_{12}	0	-1	0	3
x_{15}	0	-1	0	2
x_3	$1/5$	$-2/5$	0	2
x_{16}	$1/5$	$-7/5$	0	3
x_{19}	0	-1	0	1
x_{18}	$1/5$	$-12/5$	1	0
x_5	$-1/5$	$-3/5$	-1	4
x_{17}	$1/5$	$-7/5$	0	2
x_{20}	0	-2	-1	5
x_{22}	$-1/5$	$12/5$	-1	1
x_{21}	$1/5$	$-7/5$	0	1
x_{24}	$1/5$	$-7/5$	0	0
$\bar{c}_j$	0	-4	$-1/2$	$F_1 = 23/2$
$\bar{d}_j$	$-3/5$	$23/15$	0	$F_2 = 32/3$

Pour $x_1 = x_1^{(4)}$ résoudre le problème du second niveau (5.27) en utilisant l'algorithme 2, la solution optimale est $(\tilde{x}_2, \tilde{x}_3) = (1, 2) = (1, 2) = (x_2^4, x_3^4)$, alors $z^{*(4)} = (9, 1, 2)$ est la solution optimale entière du problème principal (5.2) avec $F(z^{*(4)}) = 368/3$ et $f(z^{*(4)}) = 11$.*

La méthode Branch & Cut pour la résolution d'un problème quadratique indéfini à
deux niveaux discret



La méthode Branch & Cut pour la résolution d'un problème quadratique indéfini à
deux niveaux discret

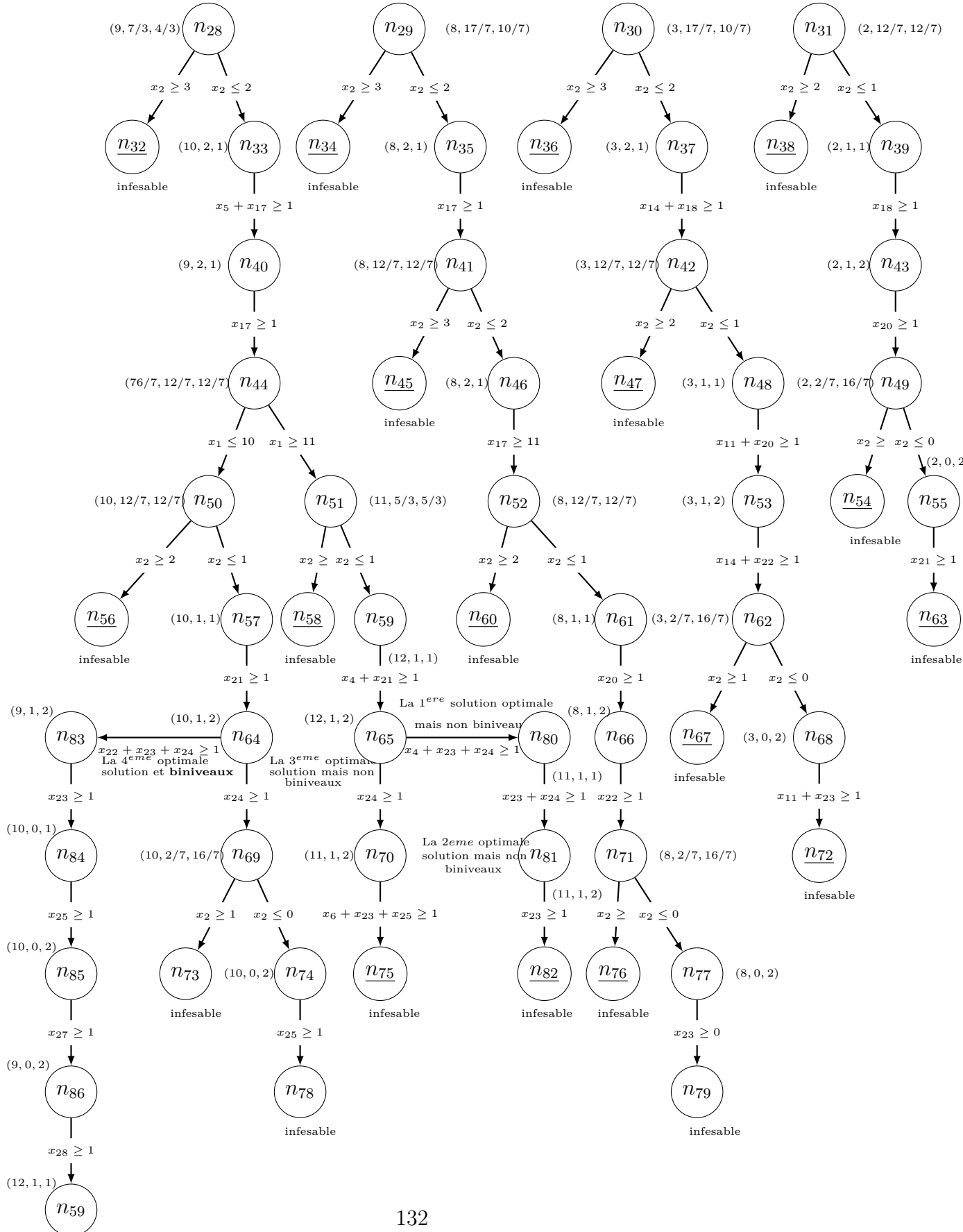


FIGURE 5.3 – Arbre représentant les états des nœuds dans l’algorithme Branch and Cut pour résoudre le problème principal

5.6 Résultats de calcul

La méthode décrite a été implémentée sous l'environnement Matlab et testée sur des problèmes *IIQBP* générés aléatoirement. Les données sont générées aléatoirement par une distribution uniforme discrète dans l'intervalle $[1, 30]$ pour les coefficients des contraintes. La valeur du côté droit b est régénérée aléatoirement dans l'intervalle $[50, 100]$. Les vecteurs $c^1, c^2, d^1, d^2, p^1, p^2, q^1, q^2$ et les scalaires $\alpha_1, \alpha_2, \beta_1, \beta_2$ sont générés dans $[1, 10]$ pour assurer la condition nécessaire de positivité pour $(c^1x + c^2y + \alpha_1)$, $(d^1x + d^2y + \beta_2)$, $(p^1x + p^2y + \alpha_2)$ et $(q^1x + q^2y + \beta_2)$. Pour chaque instance (n_1, n_2, m) (n_1 est le nombre de variables du premier niveau, n_2 est le nombre de variables du second niveau et m le nombre de contraintes), une série de 10 problèmes ont été résolus. Les expériences de calcul ont été réalisées sur un ordinateur FUJITSU de 2.4 GHZ, processeur Intel(R) Core(TM)i5-4210U et 4 GB de mémoire. Les performances obtenues de l'algorithme 3 sont résumées dans le tableau 16 où sont indiqués les nombres (moyen et maximal) de temps CPU (en secondes). Comme on peut le constater, l'algorithme 3 proposé est efficace pour les petites et moyennes dimensions en termes de temps d'exécution (CPU (secondes)).

TABLE 16 – Résultats de calcul.

n_1	n_2	m	$CPU(seconde)$	
			Moyenne	Max
10	5	5	0.8302	2.086
20	10	10	32.5516	85.208
30	15	20	165.9574	348.922
40	20	30	719.9926	1209.944
50	20	40	972.8664	1940.202
80	20	70	3224.783	4267.85

Le temps de calcul (en secondes) des deux exemples est de 0,011 et 0,067, respectivement.

5.7 Conclusion

*Dans ce chapitre, nous avons développé un nouvel algorithme pour résoudre le problème quadratique indéfini à deux niveaux en nombres entiers **IIQBP**. L'algorithme est basé sur la méthode de "branch & cut". L'algorithme proposé est une méthode exacte, qui détermine la solution optimale si elle existe. La solution optimale entière du problème **IIQBP** est obtenue en un nombre fini d'itérations. Afin de mesurer l'efficacité de notre algorithme, nous avons construit un tableau de valeurs pour déterminer le temps de calcul pour deux exemples et pour d'autres exemples de plus grandes dimensions. Le temps d'exécution était raisonnable. Notre proposition peut être appliquée aux problèmes combinatoires quadratiques indéfinis à deux niveaux puisqu'ils sont des cas particuliers de ce programme mathématique.*

Conclusion générale

Cette thèse porte sur l'optimisation non linéaire à deux niveaux. Les principaux résultats obtenus se scindent en deux parties :

Nous avons d'abord développé une méthode pour résoudre un problème fractionnaire quadratique/quadratique à deux niveaux (Maachou N et Moulaï M [88]). La méthode proposée est basée sur les conditions de Karush-Kuhn-Tucker, pour laquelle le problème du second niveau est remplacé par ses conditions d'optimalité de Karush-Kuhn-Tucker (KKT). Il en résulte un programme mathématique non linéaire à un seul niveau (PNL), où la résolution de ce problème non linéaire s'appuie essentiellement sur des résultats trouvés dans un de nos travaux cités dans la thèse (Maachou N et Moulaï M[89]), le travail consiste à résoudre un problème fractionnaire quadratique discret, la méthode est basée sur la résolution d'un problème fractionnaire linéaire, des conditions d'optimalités sont déterminées afin de résoudre le problème (PNL).

La seconde partie de notre contribution consiste à mettre au point une nouvelle méthode exacte pour résoudre le problème quadratique indéfini à deux niveaux en nombres entiers, dit IIQBP. En effet, on résout le problème quadratique indéfini du premier niveau, dont la solution optimale appartient à l'ensemble des solutions efficaces du problème bicritère associé. L'ensemble des solutions efficaces est déterminé par la méthode Branch & Cut. La méthode est basée sur la procédure Branch & Bound pour trouver une solution entière, une coupe efficace est utilisée afin d'éliminer les solutions dominées du domaine réalisable. La solution optimale entière obtenue du premier niveau est ensuite testée pour l'optimalité du problème principal en résolvant le problème du second niveau, l'algorithme exploité pour la résolution du premier niveau est retenu pour résoudre le second niveau. Si cette solution n'est pas optimale pour le problème principal, on l'élimine du domaine, en ajoutant la coupe de Dantzig au problème du premier niveau et on continue l'exploration. L'algorithme proposé est une méthode exacte, qui fournit la solution optimale si elle existe. La solution optimale entière du problème IIQBP est obtenue en un nombre fini d'itérations.

Afin de mesurer l'efficacité de l'algorithme, nous avons construit un tableau de valeurs pour déterminer le temps de calcul pour deux exemples et pour d'autres exemples de plus grandes dimensions. Le temps d'exécution était raisonnable. Un programme informatique a été développé sous environnement Matlab.

Parmi les travaux qui peuvent présenter des perspectives et qu'on souhaite aborder pour l'avenir on y trouve :

- utilisation des bornes dans la méthode de Branch & Bound dans le souci de résoudre le problème du premier niveau en un temps plus court.*
- La résolution d'un problème à deux niveaux où les facteurs sont non linéaires.*

Bibliographie

- [1] Abbaci L, Moulaï M. (2010), *L'optimisation sur l'ensemble efficient d'un problème stochastique multi-objectif discret*, thèse de Magister en Recherche Opérationnelle, USTHB, Algérie.
- [2] Abbas M, Chaabane D. (2002), *An algorithm for solving multiple objective integer linear programming problem*, *RAIRO Operations Research* 36, pp. 351-364.
- [3] Abbas M. and Moulaï M. (2002), *Integer Linear Fractional Programming with Multiple Objective*, *Recerca Operativa*, Vol. 32(103/104), pp. 51-70 .
- [4] Abbas M, Moulaï M. (1997), *Penalties Method for Integer Linear Fractional Programs*, *Belgian Journal of Operations Research, Statistics and computer sciences*, Vol. 37/4, 41-51.
- [5] Abbas M, Moulaï M. (1999), *An algorithm for mixed integer linear fractional programming problem*, *Belgian Journal of Operations Research, Statistics and computer sciences*, Vol. 39/1, 21-30.
- [6] Abbas M, Moulaï M. (1999), *Solving multiple objective integer linear programming*, *Journal of the Italian Operations Research Society (Ricerca Operativa)* 29, pp. 15-38, 1999.
- [7] Aiyoshi E., et Shimizu K. (1984), *A solution method for the static constrained Stackelberg problem via penalty method*, *IEEE Transactions on Automatic Control*, Vol. 29, pp. 1111-1114.
- [8] Anand P. (1972), *Decomposition principle for indefinite quadratic programme*, *Trabajos de Estadística de Investigación Operativa* 23, 61-71, 1972.
- [9] Anandalingam G., et White D. j. (1990), *A solution Method for the Linear Stackelberg Problem Using Penalty Functions*. *IEEE Trans. Automatic Control*, Vol. 35, No. 10.
- [10] Arora S. R., and Gupta R. (2009), *Interactive fuzzy goal programming approach for bilevel programming problem*, *European Journal of Operational Research*, Vol. 194(2), pp. 368-376.

-
- [11] Arora S. R., Alemayehu G and Narang R. (2004), *On The Indefinite Quadratic Bilevel Programming Problem*, *Operational Research Society of India*, Vol. 41(4) .
 - [12] Audet C., Hansen P., Jaumard B., Savard G. (1997), *Links between linear bi-level and mixed 0-1 programming problems*, *Journal of optimization Theory and Applications* Vol. 93, pp. 273-300.
 - [13] Axehill D. (2008), *Applications of integer quadratic programming in control and communication*, Linköping (Sweden), Linköping University, [Dissertation No.1158].
 - [14] Bard J. F. and Falk J. E. (1982), *An explicit Solution to the Multi-Level Programming Problem*, *Computers and Operation Research*, Vol. 9.
 - [15] Bard J. F. (1988), *Convex two-level optimization*, *Mathematical Programming*, Vol.40, pp 15-27.
 - [16] Bosco J. Etoa. Etoa (2007), *Optimisation hiérarchique : théorie, algorithmes et applications*, books.google.com, books.
 - [17] Bazaraa M.S., Sherali H.D. and Shetty C.M. (1992), *Nonlinear Programming : Theory and Algorithms*. J. Wiley and Sons, New York.
 - [18] Bector C.R. (1972), *Indefinite quadratic fractional functional programming*, *Metrica*, Vol. 18(1), pp 21-30.
 - [19] Bector C.R, Dahl M. (1974), *Simplex type finite iteration technique and duality for a special type of pseudo-concave quadratic program*, *Cahiers du Centre d'Etudes de Recherche Operationnelle* 16, 207-222.
 - [20] Bector C.R. (1972), *Indefinite quadratic fractional functional programming*. *Metrica* ; 18(1) :21-30.
 - [21] Bector C.R., Chandra S., Singh C. (1990), *Duality in Multiobjective Fractional Programming*. *Lecture Notes in Economics and Mathematical Systems*, vol 345. Springer, Berlin, Heidelberg.
 - [22] Belkeziz K, Metrane A. (2004), *Optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces d'un problème multicritère quadratique convexe*, *Laboratoire de mathématiques de l'université Blaise-Pascal , France*, V11, n 1, p.19-33.
 - [23] Ben-Ayed O. and Blair C. (1982), *Computational difficulties of bilevel linear programming*, *Operation Research*, Vol. 38, pp 556-560.
 - [24] Benson H.P. (1978), *Existence of Efficient Solutions for Vector Maximization Problems*, *Journal of Optimization Theory and Applications* 26, No4, 569-580.

- [25] Benson H.P. (1984), *Optimization over the efficient set*. *J Math Ana App*, 98 :562–580.
- [26] Benson H.P. (1991), *An All-Linear Programming Relaxation Algorithm for Optimizing over the Efficient Set*, *Journal of Global Optimization*, Vol.1, 83-104.
- [27] Benson H.P. (1992), *A finite Nonadjacent Extreme point Search Algorithm over the efficient Set*, *Journal of Optimization Theory and Applications*, Vol.73, 47-64.
- [28] Benson H.P. (1993), *A bisection-extreme point search algorithm for optimizing over the efficient set in the linear dependence case*, *Journal of Global Optimization* 3, 95-111.
- [29] Benson H.P., Sayin S. (1994), *Optimizing over the Efficient Set : Four Special Cases*, *Journal of optimization Theory and Applications*, Vol.80, No1 ; 3-17.
- [30] Bereanu B. (1965), *Decision regions and minimum risk solutions in linear programming*, in : A. Prekopa, Ed., *Colloquium on applications of mathematics to economics*, Budapest, Publ. house of the Hungarian academy of sciences, Budapest, 37-42.
- [31] Bitran G.R., Novaes G. (1976), *Linear Programming with a fractional objective function*, *Operations Research*, 24, 675-699.
- [32] Calvete H. I. and Galé C. (2004(b)), *Solving linear fractional bilevel programs*. *Operations Research Letters*, 32(2), 143–151.
- [33] Calvete H.I. and Galé C. (2004), *Optimality conditions for the linear fractional/Quadratic bilevel problem*, *Monografias del Seminario Garcia de Geldeano*, Vol. 31, pp. 285-294.
- [34] Cambini A., Martein L. (1986), *A modified version of Martos's algorithm for the linear fractional problem*, *Methods of Oper. Res.* 53, pp. 33-44.
- [35] Cambini A., Martein L. (1992), *Equivalence in Linear Fractional Programming*, *Optimization* 23, 41-51
- [36] Chaabane D., Pirlot M. (2010), *A method for optimizing over the integer efficient set*, *J. Ind. Man. Opt*, 6/4 : 811–823. DOI :10.3934/jimo.2010.6.811.
- [37] Chandra S., Chandramohan M. (1980), *A Branch and Bound Method for Integer Nonlinear Fractional Programs*, *Z. A. M. M.*, 60, pp. 735-737.
- [38] Charnes A., Cooper W.W. (1962), *Programming with linear fractional functionals*, *Naval Res. Logist. Quart.* 9, 181-186.
- [39] Collette Y., Siarry P. (2002), *Optimisation multiobjectif*, Editions Eyrolles.

- [40] Choo E.U. (1984), *Proper efficiency and the linear fractional vector maximum problem*, *Operational Research* 32/1, pp. 216-220.
- [41] Cherfaoui Y. and Moulai M. (2021), *Generating the efficient set of MultiObjective Integer Linear plus Linear Fractional Programming Problems*, *Annals of Operations Research*, Springer. vol. 296(1), pp. 735-753.
- [42] Candler W. and Townsley R. (1982), *A linear bilevel programming problem*, *Computers and Operations Research*, Vol. 9(1), pp. 59-76.
- [43] Chergui M. E-A. and Moulai M. (2008), *An exact method for a discrete multiobjective linear fractional optimization*, *J. Appl. Math. Decis. Sci.*, 12pp. doi : 10.1155/2008/760191 Google Scholar.
- [44] Chinchuluun A., Pardalos P. M. (2007), *A survey of recent developments in multiobjective optimization*. *Annals of Operations Research*, 154(1), 29-50. DOI : 10.1007/s10479-007-0186-0
- [45] Cottle R.W. (1975) *Three remarks about two papers on quadratic forms*, *Zeitschrift für Operations Research* 19, 123-124, 1975.
- [46] Crema A., Sylva J. (2003), *A method for finding the set of nondominated vectors for multiple objective integer linear programs*, *European Journal of Operational Research*, in press.
- [47] Dantzig G.B. (1963), *Linear programming and extention*, Princeton University Press.
- [48] Dantzig G.B. (1951), *Maximization of linear function of variables subject to linear inequalities : in t.c*, Koopmans ed. *Activity Analysis of*. John Wiley Sons, New York, 339-347.
- [49] Datta N., Bhatia D. (1984), *Algorithm to determine an initial efficient basic solution for a linear fractional multiple objective transportation problem*. *Cahiers Centre Etudes Rech. Oper*, Kaiser-slauternol, 26/ 1-2 : 127-136.
- [50] Dempe S. (2002), *Foundations of Bilevel Programming, Nonconvex Optimization and its Applications*. Kluwer Academic Publishers, Dordrecht. .
- [51] Dempe S., Mefo Kue F. (2018), Mehlitz P., *Optimality conditions for mixed discrete bilevel optimization problems*. *Optimization*, 67(6), 737-756.
- [52] Dempe S., Kalashnikov V. (2005), Rios-Mercado R.Z., *Discrete bilevel programming : application to a natural gas cash-out problem*. *Eur. J. Oper. Res.*, 166(2), 469-488.

-
- [53] Dinkelbach W. (1967), *On nonlinear fractional programming*, *Management Science*, 13 :492-498.
- [54] Dhaenens-Flipo C. (2005), *Optimisation Combinatoire Mult-Objectif : Apport des Méthodes Coopératives et Contribution à l'Extraction de Connaissances*, thèse d'Habilitation à diriger des Recherches de l'U.S.T.L, Lille.
- [55] Dorfman R. (1951), *Application of linear programming to the theory of the firm*, University of California Press.
- [56] Ecker J.G., Song J.H. (1994), *Optimizing a linear function over an efficient set*, *Journal of Optimization Theory and Applications* Vol.83, No3; 541-563 .
- [57] Ecker J, Kouada I. (1975), *Finding efficient points for linear multiple objective programs*. *Math Prog*, 8 :375-377.
- [58] Eckardt J. (1998), *Zwei-Ebenen-Optimierung mit diskreten Aufgaben in der unteren Ebene.*, Master's thesis, TU Bergakademie Freiberg, Fakultät für Mathematik und Informatik.
- [59] Ehrgott M, Hamacher HW., Klamroth K., Nickel S., Schobel A., Wiecek M.M. (1997), *A note on the equivalence of balance points and Pareto solutions in multiple-objective programming*. *Journal of Optimization Theory and Applications*, 92/1 :209-212.
- [60] Ehrgott M. (2000), *Multicriteria Optimization*, *Lecture Notes in Economics and Mathematical Systems*, No 491, Springer Verlag, Berlin, Heidelberg.
- [61] Erenguc SS, Benson H. (1991), *An algorithm for indefinite integer quadratic programming*. *Comput Math Applic*, 21,6/7 : 99 – 106.
- [62] Fortuni-Amat J. and McCarl B. (1981), *A representation and economic interpretation of a two-level, programming problem*, *Journal of Operational Research Society*, Vol. 32(9), pp. 783-792.
- [63] Geoffrion A.M. (1968) *Proper Efficiency and the Theory of Vector Maximization*, *Journal of Mathematical Analysis and Applications* 22, No-3, 618-630, 1968.
- [64] Gupta R., Malhotra R.(1992), *Multi-criteria integer linear programming problem*, *Cahiers de CERO* 34, (1992).
- [65] Granot D., Granot F. (1977), *On integer and mixed integer fractional programming problems*, *Ann. Discrete Math*, 1, 221-231.
- [66] Grunspan M. (1971), *Fractional programming : A survey*, Technical Report 50, Department of Industrial and Systems Engineering, University of Florida.

-
- [67] Gotoh J.Y. and Konno H. (2001), *Maximization of the Ratio of Two Convex Quadratic Functions over a Polytope*, *Computational Optimization and Applications*, Vol.20(1), pp. 43-60.
 - [68] Gupta R., Puri M. C. (1994), *Extreme point quadratic fractional programming problem*, *Optimization*, Vol. 30(3), pp. 205-214.
 - [69] Henderson J.M. and Quandt R.E. (1971), *Microeconomic Theory : A Mathematical Approach*, New York : McGraw-Hill.
 - [70] Júdice J., and Faustino A. (1994), *The linear-quadratic bilevel programming problem*, *INFOR*, Vol. 32(2), pp. 87-98.
 - [71] Gotoh, J.Y., Konno, H. (2001), *Maximization of the Ratio of Two Convex Quadratic Functions over a Polytope*. *Computational Optimization and Applications*, 20, 1, pp 43–60.
 - [72] Hadley M. (1962), *Linear Programming*. Reading Mass : Addison Wesley.
 - [73] Hansen P., Jaumard B., Savard G. (2012), *New branch and bound rules for linear bilevel programming*, *SIAM Journal on scientific and statistic computer* , Volume 13, 1194-1217.
 - [74] Hasan M.B. (2012), *A technique for solving special type quadratic programming problems*, *Dhaka university j. sci.*, Volume 60(2), 209-215.
 - [75] Henderson J.M., Quandt R.E. (1971), *Microeconomic theory*. New York (NY) : McGraw-Hill.
 - [76] Isermann H. (1974), *Proper efficiency and the linear vector maximization problem*. *Op Res.* ; 22 :189–191.
 - [77] Isermann H., Steuer R.E. (1987), *Computational Experience Concerning Payoff Tables and Minimum Criterion Values over the Efficient Set*. *European Journal of Operational Research*, Vol.33, 91-97.
 - [78] Jeroslow R.G (1985), *The polynomial hierarchy adn simle model for competitive analysis*, *Math. Programming*, Vol. 32, pp. 146-164.
 - [79] Jorge J.M. (2009), *An algorithm for optimizing a linear function over an integer efficient set*. *Eur J Oper Res.*, 195 :98–103.
 - [80] Konno H. and Kuno T. (1992), *Linear multiplicative programming*, *Mathematical Programming*, Vol. 56, pp. 51-64.
 - [81] Karmarkar N.K. (1984), *A new polynomial-time algorithm for linear programming*, *Combinatorica* 373-395.

-
- [82] Klein D, Hannan E. (1992), *An algorithm for multiple objective integer linear programming problem*, *European Journal of Operational Research* 9, pp. 378-385.
 - [83] Koopmans T.C. (1971), *Analysis and Production as an Efficient Combination of Activities*, *Activity Analysis of Production and Allocation*, Yale University Press, New Haven, London, 33-97 (originally published in 1951).
 - [84] Kuhn H.W, Tucker A.W. (1951), *Nonlinear Programming*, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, Los Angeles, 481-492.
 - [85] Li J.L., Wang P., Jian J. (2015), *A New Algorithm Based on Lagrangian Relaxation for Solving Indefinite Quadratic Integer Programming*, *J. COMPUT. ANAL. APPL.*, 19/6 pp. 1038-1048.
 - [86] Luenberger D.G, Ye Y. (2008), *Linear and Nonlinear Programming*, Third Edition, Springer.
 - [87] Lokman B., Köksalan M., Korhonen P.J., Wallenius J. (2016), *An interactive algorithm to find the most preferred solution of multi-objective integer programs*. *Annals of operations research*, 245(1-2), 67-95.
 - [88] Maachou N. and Moulaï M. (2015), *Bilevel Quadratic Fractional/Quadratic Problem*, Le Thi H., Pham Dinh T., Nguyen N. (eds) *Modelling, Computation and Optimization in Information Systems and Management Sciences. Advances in Intelligent Systems and Computing* Springer, Cham., Vol. 359, pp.381-389.
 - [89] Maachou N., Moulaï M. (2008), *An Exact Method for a Discrete Quadratic Fractional Maximum Problem*, Le Thi H.A., Bouvry P., Pham Dinh T. (eds) *Modelling, Computation and Optimization in Information Systems and Management Sciences. Communications in Computer and Information Science*, vol 14. Springer, Berlin, Heidelberg. <https://doi.org/10.1007/978-3-540-87477-5-22>.
 - [90] Mahdi S., Chaabane D. (2015), *A linear fractional optimization over an integer efficient set*. *RAIRO-Oper. Res.*, 49 : 265-278. DOI : 10.1051/ro/2014036.
 - [91] Marcotte P., and Savard G., *Bilevel programming : A combinatorial perspective, graph theory and combinatorial optimization*, Eds. D. Avis, A. Hertz, O. Marcotte, Springer
 - [92] Marchi E. (2008), *When is the product of two concave functions concave ?*, *IMA Preprint Series* #2204, Institute for Mathematics and its Applications, University of Minnesota, pp. 1-8.
 - [93] Markowitz H. (1959), *Portfolio selection : Efficient diversification of investment*, John Wiley and Sons, New York.

-
- [94] Martos B. (1964), *Hyperbolic programming*, *Naval Res. Logist. Quart.*, 11, pp. 135-155.
 - [95] Martos B. (1975), *Nonlinear programming, Theory and Methods*. North-Holland, Amsterdam.
 - [96] Miettinen K. (1984), *On the Methodology of Multiobjective Optimization Problems*, *Journal of Optimization Theory and Applications* 42, No4, 499-524.
 - [97] Miettinen K. (1999), *Nonlinear Multiobjective Optimization*, Kluwer Academic Publishers, Boston.
 - [98] Miettinen K. (2001), *Some Methods for Nonlinear Multi-objective Optimization*, In : Zitzler E., Thiele L., Deb K., Coello Coello C.A., Corne D. (eds) *Evolutionary Multi-Criterion Optimization. EMO 2001. Lecture Notes in Computer Science*, vol 1993. Springer, Berlin, Heidelberg
 - [99] Moulaï M., Drici W. (2018), *An indefinite quadratic optimization over an integer efficient set*, *Optimization*, 67/8, pp. 1143-1156. Available at <https://doi.org/10.1080/02331934.2018.1456539>.
 - [100] Murt K. G. and Chung S. J.(1992), *Extreme Point Enumeration*. College of engineering, University of Michigan. Technical Report 92-21.
 - [101] Moulaï M. and Drici, W. (2018), *An indefinite quadratic optimization over an integer efficient set*, *Optimization*, Vol. 64, No. 4, pp. 135-155 . Available at <https://doi.org/10.1080/02331934.2018.1456539>.
 - [102] Mishra, S., Ghosh A. (2006), *Interactive fuzzy programming approach to Bi-level quadratic fractional programming problems*. *Ann. Oper. Res.* 143, 251--263.
 - [103] Nagih A., Plateau G. (1999), *Problèmes fractionnaires : Tour d'horizon sur les applications et méthodes de résolution*, *RAIRO Oper. Res.* 33, 383-419.
 - [104] Narang R. and Arora S. R. (2009), *Indefinite quadratic integer bilevel programming problem with bounded variable*, *Journal of operational research society of India (OPSEARCH)*, Vol.46(4), pp. 428-448.
 - [105] Nash O.F. (1950), *The bargaining problem*, *Econometrica*, Vol. 18, pp. 155-162.
 - [106] Nemhauser G.L., Wolsey L.A. (1988), *Integer and combinatorial optimization*, Wiley, Chichester.
 - [107] Nguyen N.C. (1992), *An Algorithm for Optimizing a Linear Function over the Integer Efficient Set*, *Konrad-Zuse-zentrum fur Informationstechnik Berlin*.
 - [108] Ozlen M., Azizoglu M., Burton BA. (2013), *Optimising a nonlinear utility function in multi-objective integer programming*. *J Global Optim.* ; 56(1) :93-102.

-
- [109] Philip J. (1972), *Algorithms for the vector maximization problem*, *Mathematical programming* 2, 207-229.
 - [110] Padberg M. (2005), *Classical cuts for mixed-integer programming and branch-and-cut*. *Annals of Operations Research*, 139(1), 321-352.
 - [111] Savard G. and Gauvin J. (1994), *The steepest descent direction for the nonlinear bilevel programming problem*, *Operation Research letters*, Vol. 15, pp. 265-272 .
 - [112] Sakarovitch M. (1984), *Optimisation combinatoire, Graphes et programmation linéaire*, Hermann.
 - [113] Sayin S. (2000), *Optimizing over the efficient set using a topdown search of faces*, *Operations Research* 48, 65-72.
 - [114] Schaible S.(1976), *Duality in fractional programming : a unified approach*, *Operations Research*, 24, No-3 :452-461.
 - [115] Schaible S.(1981), *Fractional programming : applications and algorithms*, *European Journal of Operational Research* 7, 111-120.
 - [116] Schaible S.(1981), *Fractional programming*, *Zeitschrift für, Operations Research*, 27 :39-54.
 - [117] Schaible S.(1981), *Fractional programming*, in R. Horst and P.M. Pardalos (eds.), *Handbook of Global Optimization*, Kluwer Academic Publishers, 495- 608.
 - [118] Schrijver A. (1998), *Theory of linear and integer programming*, Wiley - *Inter-science Series in Discrete Mathematics and Optimization*.
 - [119] Sharma S.D. (1994), *Non-linear and dynamic Programming*, Kedar Nath Ram Nath & CO, Meerut, India.
 - [120] Shimizu K., et Lu M. (1995), *A Global Optimization Method for the Stackelberg Problem with Convex Functions via Problem Transformation and Concave Programming*, *IEEE Trans. Systems, Man and Cybernetics* Vol. SMC-25, No. 12.
 - [121] Shimuzu K., Ishizuka Y., Bard J. F. (1997), *Nondifferentiable and two level mathematical programming*, Kluwer Academic Publishers.
 - [122] Shigeno M, Takahashi I, Yamamoto Y. (2003), *Minimum maximal flow problem An optimization over the efficient set*. *Journal of Global Optimization*, 22 : 425-443.
 - [123] Simonard M. (1972), *Programmation linéaire*, 2^{ème} édition, tomes 1 et 2, Dunod.
 - [124] Sinha S. (2003), *Fuzzy programming approach to multi-level programming problems*, *Fuzzy Sets and Systems*, Vol. 136(2), pp. 189-202.

- [125] Stancu-Minasian I.M. (1997), *Fractional Programming : Theory, Methods and Applications*, Kluwer Academic Publishers, 45, 1-4 :343-367.
- [126] Stancu-Minasian I.M. (1999), *A fifth bibliography of fractional programming*, *Optimization* 45, 1-4 :343-367.
- [127] Stancu-Minasian I.M. (2006), *A sixth bibliography of fractional programming*, *Optimization* 55, 4, 405–428.
- [128] Steuer R.E. (1985), *Multiple Criteria Optimization : Theory, Computation and Applications*, John Wiley and Sons, New-York.
- [129] Stancu-Minasian I.M. (2006), *A sixth bibliography of fractional programming*. *Optimization* 55/4, 405–428.
- [130] Sun X.L., Li J.L., Luo H.Z. (2010) *Convex Relaxation and Lagrangian Decomposition for Indefinite Quadratic Integer Programming*, *Optimization*, 59 ,pp. 627–641.
- [131] Swarup K. (1966), *Quadratic programming*. *Cahier du Centre d'Etudes et de Recherche Opérationnelle*, 8(4) : 223 – 234.
- [132] Swarup K. (1966), *Indefinite quadratic programming*, *Faculty of Mathematics, University of Delhi, Delhi-7. Lecturer in Operational Research*, Vol. 8, pp. 217-222 .
- [133] Teghem J. (1998), *Programmation linéaire*, Ellipses.
- [134] Teghem J., Kunsh P.L. (1986), *A Survey Of Techniques For Finding Efficient Solutions To Multi-Objective Integer Linear Programming*, *Asia-Pacific Journal of Operational Research* 3 95-108.
- [135] Thirwani D. and Arora S. R. (1998), *An algorithm for quadratic bilevel programming problem*, *International Journal of Management and System*, Vol. 14(2), pp. 89-98 .
- [136] Vicente L., Savard G. and Jùdice J. (1994), *Descent approaches for quadratic bilevel programming*, *Journal of Optimization Theory and Applications*, Vol. 81(2), pp. 379-399.
- [137] Wang S. and Wang Q. (1994), *Optimality conditions and an algorithm for linear-quadratic bilevel programs*, *Optimization*, Vol. 31(2), pp. 127-139.
- [138] Wendell R.E, Lee D.N. (1977), *Efficiency in Multiple Objective Optimization Problems*, *Mathematical Programming* 12, No-3, 406-414.
- [139] Yamamoto Y. (2002) *Optimization over the efficient set : Overview*. *J Glo Opt.*, 22/1-4 :285-317.

- [140] Zellner A. (1961), *Linear regression with inequality constraints on the coefficients : An application of quadratic programming and linear decision rules*. *Econ Inst Netherlands School of Economics*, 6109.
- [141] Zerdani O, Moulaï M. (2011), *Optimization over an integer efficient set of a multiple objective linear fractional problem*. *App Mat Sci.*, 5/50 :2451-2466.
- [142] Zerdani O. (2013), *L'Optimisation non Linéaire Multiobjectif*, thèse de doctorat en mathématiques, option Recherche Opérationnelle, USTHB, Algérie.