

UNIVERSITÉ DE MONTRÉAL

MODÉLISATION, ÉNUMÉRATION ET RAFFINEMENT
DES ÉQUILIBRES EN THÉORIE DES JEUX

SLIM BELHAIZA
DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION
DU DIPLÔME DE PHILOSOPHIAE DOCTOR (Ph.D.)
(MATHÉMATIQUES DE L'INGÉNIEUR)

JUIN 2008

©Slim Belhaiza, 2008.



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-46090-0
Our file *Notre référence*
ISBN: 978-0-494-46090-0

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

■ ■ ■
Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

MODÉLISATION, ÉNUMÉRATION ET RAFFINEMENT
DES ÉQUILIBRES EN THÉORIE DES JEUX

présentée par : BELHAIZA Slim

en vue de l'obtention du diplôme de : Philosophiæ Doctor

a été dûment acceptée par le jury d'examen constitué de :

M. DESAULNIERS Guy, Ph.D., président

M. AUDET Charles, Ph.D., membre et directeur de recherche

M. HANSEN Pierre, D.Agr., membre et codirecteur de recherche

M. ZACCOUR Georges, Ph.D., membre

M. PLATEAU Gérard, Doctorat, membre externe

REMERCIEMENTS

Mes louanges à Allah (Dieu) qui nous a tant donné.

Je dédie ce travail à ma mère, mon père, mon frère, mes soeurs, à mon épouse, à notre enfant et à toute ma famille.

Je remercie Charles pour son soutien et sa patience. Je remercie Pierre pour son aide et ses conseils.

Je remercie aussi tous mes amis à Montréal, Tunis et ailleurs, qui m'ont aidé à prendre les devants et réduire mon ignorance.

RÉSUMÉ

Cette thèse de doctorat propose plusieurs formulations mathématiques linéaires et non linéaires, dans le but d'énumérer des équilibres de Nash extrêmes et de les raffiner en théorie des jeux. Parmi les différentes formes de jeux généralement formulées dans la littérature, nous étudions plus particulièrement les jeux bimatriciels, polymatriciels et la forme séquentielle des jeux étendus à deux joueurs.

Notre premier objectif est d'utiliser la linéarisation des conditions de complémentarité afin de concevoir un algorithme d'énumération des équilibres extrêmes pour les jeux bimatriciels et polymatriciels.

Dans le premier chapitre, nous effectuons une synthèse des principales formulations mathématiques pour la recherche d'équilibres de Nash pour les jeux bimatriciels. Nous exposons l'algorithme de Lemke et Howson (37) pour la recherche d'équilibre de Nash. Ensuite, nous présentons l'algorithme *EEE* (6) pour l'énumération des équilibres de Nash extrêmes. Nous montrons aussi comment le problème d'énumération des sous-ensembles de Nash maximaux peut être résolu en utilisant un algorithme d'énumération des cliques maximales en théorie des graphes. Finalement, nous linéarisons les conditions de complémentarité en utilisant la méthode de Júdice et Mitra (29) et en introduisant des variables binaires. Cette linéarisation permettra de proposer un programme linéaire mixte 0 – 1 dont les solutions extrêmes sont des équilibres de Nash extrêmes du jeu bimatriciel.

Le deuxième chapitre est consacré en premier lieu à une synthèse des formulations mathématiques pour la recherche d'équilibres de Nash pour les jeux polymatriciels. Nous exposons ensuite l'algorithme des pivots complémentaires de recherche d'équilibre de Nash pour les jeux polymatriciels (70). En troisième lieu, nous approfondissons la méthode d'élimination des stratégies dominées pour les jeux bimatriciels et polymatriciels à trois joueurs. Finalement, nous linéarisons les conditions de complémentarité en introduisant des variables binaires. Cette nouvelle modélisation permettra de proposer un

programme linéaire mixte 0 – 1 dont les solutions extrêmes sont des équilibres de Nash extrêmes du jeu polymatriciel.

Notre deuxième objectif est d'étudier l'énumération des équilibres de Nash extrêmes pour la forme séquentielle d'un jeu étendu à deux joueurs. Nous présentons dans le troisième chapitre une revue de la littérature sur les différents concepts utilisés pour les jeux étendus. Ensuite, nous étudions les différentes représentations d'un jeu étendu à deux joueurs. Finalement, nous proposons un programme linéaire mixte 0 – 1 dont les solutions extrêmes permettent, par projection sur l'espace des variables des stratégies mixtes, d'énumérer les équilibres de Nash extrêmes de la représentation séquentielle d'un jeu étendu à deux joueurs.

Notre troisième objectif est de concevoir en premier lieu un algorithme permettant d'énumérer tous les équilibres de Nash extrêmes pour les jeux bimatriciels, polymatriciels et les formes séquentielles des jeux étendus à deux joueurs et d'implanter ce nouvel algorithme ainsi que l'algorithme *EEE* en utilisant une librairie d'arithmétique exacte. Nous commençons le quatrième chapitre en présentant l'algorithme *E χ MIP* pour l'énumération des équilibres extrêmes en théorie des jeux. En première étape, nous implantons cet algorithme en utilisant les bibliothèques du logiciel d'optimisation *Cplex*.

Nous proposons dans le quatrième chapitre une formulation biniveau permettant d'étendre l'algorithme *EEE* aux formes séquentielles des jeux étendus à deux joueurs. Nous avons pu implémenter les algorithmes *EEE* et *E χ MIP*, en *C* et *C++*, en faisant appel à la librairie d'arithmétique exacte de McCutchen (44). Nous décrivons dans ce chapitre les détails techniques de ces implantations. En outre, nous y exposons le logiciel *XGame Solver* qui regroupe ces implantations sous une application à interfaces en *QT* facile à manipuler et mise gratuitement à la disposition de la communauté scientifique.

Notre quatrième objectif est de définir des méthodes automatiques pour le raffinement des équilibres en théorie des jeux. Dans le cinquième chapitre, nous exposons les différents raffinements dans la littérature de théorie des jeux. Nous proposons une paire de programmes linéaires dont la résolution permet d'affirmer si un équilibre de Nash est

parfait ou non. Nous obtenons aussi une série de propositions concernant les équilibres propres. Ces propositions nous conduiront à énoncer un théorème qui définit un programme quadratique mixte 0 – 1 dont la résolution permet de conclure si un équilibre n'est pas propre.

ABSTRACT

This thesis proposes a number of linear and non-linear mathematic formulations in order to enumerate and refine extreme Nash equilibria in Game Theory. We study in particular Bimatrix, Extensic and Polymatrix games.

Our first goal is to linearise complementarity conditions in order to design an algorithm for the enumeration of extreme Nash equilibria of Bimatrix, Polymatrix and Sequential forms of two-persons extensive games.

In the first chapter of this thesis, we provide a review of Nash equilibria mathematic formulations in Bimatrix games. We also present the Lemke and Howson (37) and the *EEE* (6) algorithms. We show how enumeration of all maximal Nash-subsets could be performed using an algorithm for maximal cliques enumeration from Graph Theory. We finally, present a linearisation of complementarity conditions, using binary variables, in order to formulate a mixed 0–1 program. The set of all extreme solutions of this program defines the set of extreme Nash equilibria of a Bimatrix Game.

The second chapter provides a review of Nash equilibria mathematic formulations in Polymatrix games. We present the complementary pivoting algorithm to find Nash equilibria. We also focus on the elimination of dominated strategies in Bimatrix and Polymatrix games. We finally present a linearisation of complementarity conditions, using binary variables, in order to formulate a mixed 0 – 1 program. The set of all extreme solutions of this program defines the set of extreme Nash equilibria of a Polymatrix Game.

Our second goal is to study the enumeration of extreme Nash equilibria for the sequential form of a two-persons extensive game. We present in the third chapter a review of the main concepts used for extensive games. We study the different representations of two-persons extensive games. We finally propose a mixed 0 – 1 program. The set of all extreme solutions of this program defines the set of extreme Nash equilibria of the sequential form a two-persons extensive game.

Our third goal is to design an algorithm in order to enumerate all extreme Nash equilibria of Bimatrix, Polymatrix and Sequential forms of two-persons extensive games. We present the *EXMIP* algorithm for the enumeration of all extreme Nash equilibria in Game Theory. We also present the reformulation of the *EEE* algorithm (6) on sequential forms of two-persons extensive games. Both algorithms are implemented using *Cplex* callable libraries and exact arithmetics libraries (44). We finally present our *XGame Solver* application which regroups our algorithms implementations using *QT* interfaces. This application is freely available for the use of the scientific community.

Our fourth goal is to define a set of automatic equilibria refinement methods in Game Theory. In the fifth chapter, we present different refinements used in game theory. We propose a pair of linear programs in order to check the perfectness of a Nash equilibrium. We also obtain a set of propositions on proper equilibria. Finally we provide a theorem defining a quadratic mixed 0 – 1 program in order to check if an equilibrium is proper or not.

TABLE DES MATIÈRES

| | |
|--|------|
| REMERCIEMENTS | iv |
| RÉSUMÉ | v |
| ABSTRACT | viii |
| TABLE DES MATIÈRES | x |
| LISTE DES TABLEAUX | xiii |
| LISTE DES FIGURES | xiv |
| INTRODUCTION | 1 |
| CHAPITRE 1 LES JEUX BIMATRICIELS | 5 |
| 1.1 Revue de la littérature | 9 |
| 1.2 Algorithme de Lemke et Howson | 13 |
| 1.3 Algorithme d'énumération des équilibres extrêmes <i>EEE</i> : | 15 |
| 1.4 Énumération des sous-ensembles de Nash maximaux | 17 |
| 1.5 Modélisation et formulation linéaire mixte d'un jeu bimatriciel | 20 |
| 1.6 Discussion | 25 |
| CHAPITRE 2 LES JEUX POLYMATRICIELS | 26 |
| 2.1 Revue de la littérature | 30 |
| 2.2 Algorithme des pivots complémentaires | 31 |
| 2.3 Élimination des stratégies dominées | 32 |

| | | |
|---|--|-----------|
| 2.3.1 | Les stratégies fortement dominées | 32 |
| 2.3.2 | Les stratégies faiblement dominées | 36 |
| 2.4 | Modélisation et formulation linéaire mixte d'un jeu polymatriciel | 37 |
| 2.5 | Discussion | 41 |
| CHAPITRE 3 LES JEUX SOUS FORME ÉTENDUE | | 43 |
| 3.1 | Revue de la littérature | 44 |
| 3.1.1 | Les jeux étendus | 44 |
| 3.1.2 | Jeux coopératifs et jeux non-coopératifs | 46 |
| 3.1.3 | Jeux à information complète et jeux à information incomplète | 48 |
| 3.1.4 | Jeux à information parfaite et jeux à information imparfaite | 49 |
| 3.2 | Les représentations d'un jeu étendu | 49 |
| 3.2.1 | Représentation sous forme stratégique | 49 |
| 3.2.2 | Représentation sous forme séquentielle | 52 |
| 3.2.3 | Représentation multi-agents | 53 |
| 3.3 | Formulation linéaire mixte 0 — 1 | 55 |
| 3.4 | Discussion | 61 |
| CHAPITRE 4 ALGORITHMES DE RÉOLUTION | | 62 |
| 4.1 | Présentation de l'algorithme $E\chi MIP$ | 62 |
| 4.2 | Algorithme $E\chi MIP$ | 64 |
| 4.3 | Exemples illustrés | 67 |
| 4.3.1 | Jeux bimatriciels | 67 |
| 4.3.2 | Jeux polymatriciels | 75 |
| 4.3.3 | Les jeux étendus à deux joueurs | 78 |
| 4.4 | L'algorithme EEE revisité | 82 |

| | | |
|--|--|-----|
| 4.5 | Implantation en arithmétique exacte et le logiciel <i>XGame Solver</i> | 85 |
| 4.5.1 | Implantation en arithmétique exacte | 86 |
| 4.5.2 | La librairie <i>BigInteger</i> | 86 |
| 4.5.3 | La classe <i>Rational</i> | 87 |
| 4.5.4 | La classe <i>Simplex</i> | 87 |
| 4.5.5 | La classe <i>Node</i> | 88 |
| 4.5.6 | Résultats numériques | 88 |
| 4.5.7 | Description du logiciel <i>XGame Solver</i> | 91 |
| 4.5.8 | Interface principale | 91 |
| 4.5.9 | Créer un nouveau jeu | 93 |
| 4.5.10 | Les icônes d'édition | 97 |
| 4.5.11 | La fenêtre de sortie | 97 |
| 4.6 | Discussion | 97 |
| CHAPITRE 5 RAFFINEMENT DES ÉQUILIBRES | | 100 |
| 5.1 | Équilibre essentiel | 101 |
| 5.2 | Équilibre régulier | 102 |
| 5.2.1 | Équilibre quasi-fort | 103 |
| 5.2.2 | Équilibre isolé | 104 |
| 5.3 | Équilibre parfait | 106 |
| 5.4 | Équilibre propre | 113 |
| 5.5 | Discussion | 127 |
| CHAPITRE 6 CONTRIBUTIONS ET PERSPECTIVES | | 128 |
| 6.1 | Contributions | 128 |
| 6.2 | Perspectives | 131 |
| RÉFÉRENCES | | 133 |

LISTE DES TABLEAUX

| | | |
|-------------|--|-----|
| Tableau 1.1 | Équilibres de Nash extrêmes pour l'exemple de Borm <i>et al.</i> (14) | 19 |
| Tableau 3.1 | Dilemme du prisonnier | 47 |
| Tableau 4.1 | Équilibre de Nash extrêmes de l'exemple (4.3.1) (3×3) | 71 |
| Tableau 4.2 | Équilibres de Nash extrêmes du jeu bimatriciel de Winkels (68) . . . | 75 |
| Tableau 4.3 | Équilibres de Nash extrêmes de l'exemple (4.3.3) ($3 \times 3 \times 3$) . . . | 78 |
| Tableau 4.4 | Équilibres extrêmes de la forme séquentielle du jeu de von Stengel <i>et al.</i> (64) | 78 |
| Tableau 4.5 | Équilibres extrêmes de la forme stratégique du jeu de Kohlberg et Mertens (31) | 80 |
| Tableau 4.6 | Équilibres extrêmes de la forme séquentielle du jeu étendu de Kohlberg et Mertens (31) | 81 |
| Tableau 4.7 | Jeux bimatriciels avec <i>EXMIP</i> vs <i>EEE</i> | 89 |
| Tableau 4.8 | Jeux polymatriciels à trois joueurs avec <i>EXMIP</i> ; $m_1 = m_2 = m_3$ | 90 |
| Tableau 5.1 | Jeu bimatriciel avec un seul équilibre en stratégies mixtes . . . | 101 |
| Tableau 5.2 | Ajout des stratégies d'un équilibre instable | 102 |
| Tableau 5.3 | Équilibres de Nash extrêmes pour Myerson (49) | 104 |
| Tableau 5.4 | Forme stratégique réduite (23) | 107 |
| Tableau 5.5 | Raffinement des équilibres de Nash extrêmes pour l'exemple de Borm <i>et al.</i> (14) | 126 |
| Tableau 5.6 | Équilibres de Nash extrêmes pour le jeu bimatriciel (5×5) . . | 126 |
| Tableau 5.7 | Exemple (5×5) | 127 |

LISTE DES FIGURES

| | | |
|-------------|--|-----|
| Figure 1.1 | Graphe d'interchangeabilité $G = (V, E)$ | 19 |
| Figure 1.2 | Sous-ensemble de Nash maximal T_1 | 20 |
| Figure 3.1 | Jeu sous forme étendue à deux joueurs de von Stengel (62) | 51 |
| Figure 4.1 | Arbre d'exploration avec $E\chi MIP$ sur l'exemple (4.1) : (3×3) | 68 |
| Figure 4.2 | Arbre d'exploration avec $E\chi MIP$ sur le jeu bimatriciel de Winkels (68) | 73 |
| Figure 4.3 | Arbre d'exploration avec $E\chi MIP$ sur l'exemple (4.3.3) : $(3 \times 3 \times 3)$ | 76 |
| Figure 4.4 | Jeu étendu de Kohlberg et Mertens (31) | 79 |
| Figure 4.5 | Interface principale de <i>XGame Solver</i> | 92 |
| Figure 4.6 | Interface Nouveau jeu bimatriciel | 93 |
| Figure 4.7 | Interface Nouveau jeu séquentiel | 94 |
| Figure 4.8 | Interface Nouveau jeu polymatriciel | 95 |
| Figure 4.9 | Interface de Transformation | 96 |
| Figure 4.10 | Fenêtre de sortie | 98 |
| Figure 5.1 | Jeu étendu d'Harsanyi et Selten (23) | 106 |
| Figure 5.2 | Jeu étendu à deux joueurs de Myerson (49) | 109 |
| Figure 5.3 | Courbe de $f(\sigma) = \min \epsilon$, pour l'équilibre 3 | 125 |

INTRODUCTION

La globalisation des conflits sociaux, économiques, commerciaux et diplomatiques, confère une importance croissante aux contributions des scientifiques dans la modélisation et la résolution de ces problèmes des temps modernes. Parmi les confrontations les plus intenses, les conflits entre groupes de pression et les organisations à intérêts privés et publics, sont des plus passionnants.

Les impacts des manifestations pacifistes et non-pacifistes sur les législations dans plusieurs pays industrialisés et non-industrialisés ont de plus en plus de répercussions sur notre mode de vie. Par ailleurs, les firmes privées même les plus monopolistiques deviennent beaucoup plus sensibles aux attentes des consommateurs en matière d'écologie et d'environnement. Les spécialistes en théorie des jeux contribuent à la gestion de ces conflits en employant des modélisations sous plusieurs formes et en alternant les approches de résolution.

La théorie des jeux est une branche des mathématiques qui s'intéresse à l'étude et à l'analyse des situations conflictuelles dans plusieurs domaines. La théorie des jeux permet de modéliser les confrontations stratégiques entre divers intervenants en leur proposant une aide substantielle dans leur prise de décisions. La description et la détection des situations d'équilibre pour plusieurs classes de jeux est d'un intérêt majeur pour les chercheurs en mathématiques, en économie et en sciences sociales.

Ce travail est une étude portant sur les jeux bimatriciels et polymatriciels ainsi que sur la forme séquentielle des jeux sous forme étendue à deux joueurs. Les jeux sous forme stratégique forment une classe particulière en théorie des jeux. Ils sont statiques et simultanés. Les prises de décisions y sont uniques, par opposition aux jeux sous forme étendue ou dynamique. L'énumération complète des équilibres de Nash extrêmes pour

cette classe de jeux permet d'identifier les différents ensembles d'équilibres qui peuvent être obtenus.

Le premier chapitre de cette thèse est une étude du problème d'énumération des équilibres extrêmes dans les jeux bimatriciels. Après une définition des principales notions techniques utilisées, une revue de la littérature pour le problème d'énumération des équilibres extrêmes dans un jeu bimatriciel et une description des algorithmes de Lemke et Howson (37) et d'Audet *et al.* (6) sont présentées. L'énumération des ensembles maximaux de Nash est aussi étudiée comme un problème d'énumération de cliques maximales dans un graphe. À travers des propositions soutenues par des preuves, la modélisation et la formulation linéaire mixte 0 – 1 d'un jeu bimatriciel sont ensuite proposées. Un exemple d'un programme linéaire mixte 0 – 1, pour un jeu bimatriciel, est ensuite exposé.

Le deuxième chapitre est une étude du problème d'énumération des équilibres extrêmes dans les jeux polymatriciels. Une revue de la littérature et des principales notions relatives au problème de recherche d'équilibre dans un jeu polymatriciel ainsi qu'une description de l'algorithme des pivots complémentaires sont présentées. Les notions de stratégies *fortement dominées* et *faiblement dominées* sont ensuite abordées. Le concept d'élimination des stratégies fortement dominées, dans un jeu bimatriciel et dans un jeu polymatriciel, est illustré à travers des exemples numériques. Une proposition relative au nombre maximal d'itérations nécessaires à la détection récursive des stratégies fortement dominées dans un jeu polymatriciel est ensuite énoncée et démontrée. La modélisation d'un jeu polymatriciel sous forme d'un programme linéaire mixte est enfin proposée et démontrée. Un exemple d'un programme linéaire mixte 0 – 1, pour un jeu polymatriciel à trois joueurs, est ensuite exposé.

Le troisième chapitre est une étude du problème d'énumération des équilibres de Nash extrêmes pour les formes stratégiques et séquentielles des jeux étendus à deux joueurs.

Les différents concepts utilisés en théorie des jeux sont présentés. La modélisation de la forme séquentielle d'un jeu étendu à deux joueurs sous forme d'un programme linéaire mixte est ensuite proposée et démontrée. Quelques exemples de programmes linéaires mixtes 0 – 1 obtenus pour différents exemples sont ensuite exposés.

Le quatrième chapitre présente l'algorithme $E\chi MIP$ pour l'énumération complète des équilibres extrêmes dans un jeu bimatriciel, polymatriciel et de la forme séquentielle d'un jeu étendu à deux joueurs. Un théorème relatif à l'aptitude de l'algorithme à énumérer tous les équilibres de Nash extrêmes de tels jeux est énoncé et démontré. Des exemples numériques sur différents jeux sont illustrés afin d'expliquer l'exécution de l'algorithme $E\chi MIP$. L'implantation de l'algorithme EEE pour la forme séquentielle d'un jeu étendu à deux joueurs est ensuite présentée. Ce chapitre contient aussi une section consacrée à l'implantation en arithmétique exacte des deux algorithmes. Ce chapitre décrit en outre le logiciel *XGame Solver* que nous avons réalisé au cours de ce projet de doctorat. Le logiciel regroupe nos différentes implantations en arithmétique exacte des algorithmes EEE et $E\chi Mip$ ainsi que les quelques méthodes de raffinement automatique que nous avons proposées ultérieurement. Nous discuterons aussi de l'intérêt d'offrir à la communauté scientifique un logiciel tel que *XGame Solver* et de l'enrichir à travers l'implantation d'autres algorithmes de recherche d'équilibre. Nous envisagerons aussi l'élargissement de la gamme des jeux que le logiciel pourrait résoudre.

Le cinquième et dernier chapitre est une étude des différents raffinements des équilibres de Nash extrêmes. En premier lieu, les raffinements essentiel, quasi-fort, isolé et régulier sont présentés. Pour le raffinement parfait, une définition alternative est présentée sous la forme d'une paire de programmes linéaires dont la résolution permet de déterminer en pratique si un équilibre est parfait. Le raffinement propre est ensuite étudié en proposant un programme paramétré, quadratique et mixte 0 – 1 dont l'analyse de la réalisabilité permet de conclure, d'un point de vue théorique, si un équilibre est propre ou non.

La dernière partie de cette thèse est une conclusion générale sur les réalisations de cette thèse et sur les différentes possibilités d'étendre ce travail et de l'appliquer à plusieurs problèmes de confrontations stratégiques modernes.

Les codes sources des algorithmes $E\chi MIP$ et EEE , pour l'implantation en arithmétique exacte et pour l'implantation utilisant *Cplex*, sont publiés sur le lien internet <http://www.gerad.ca/~slimb>. Le code source du logiciel *XGame Solver* y sera aussi disponible et pourrait être compilé par les utilisateurs sous les systèmes d'exploitation Windows, Linux et Mac OS.

CHAPITRE 1

LES JEUX BIMATRICIELS

Un jeu bimatriciel naît de la rencontre de deux joueurs I et II, pouvant représenter des intervenants politiques, économiques ou sociaux. Chaque joueur possède une panoplie d'opportunités stratégiques dénombrables, couramment nommée ensemble de stratégies pures. Le joueur I a le choix entre n stratégies pures, alors que le joueur II dispose de m stratégies pures.

Un jeu bimatriciel peut être résumé par une paire de matrices de gains A et B de l'ensemble des matrices, de n lignes et m colonnes, $M_{n \times m}$. L'élément a_{ij} de la matrice A représente le gain immédiat du joueur I, si les deux joueurs choisissent respectivement leur $i^{\text{ème}}$ et $j^{\text{ème}}$ stratégie au même moment. Réciproquement, l'élément b_{ij} de la matrice B représente le gain immédiat du joueur II, si les deux joueurs choisissent respectivement leur $i^{\text{ème}}$ et $j^{\text{ème}}$ stratégie au même moment.

Dans un jeu bimatriciel, chaque intervenant désire adopter la (les) stratégie(s) lui permettant de maximiser son gain à l'issue d'un jeu sous forme normale (stratégique). Un jeu sous forme normale est un jeu où chaque participant a le droit d'effectuer son choix stratégique une seule fois, sachant que tous les choix sont formulés simultanément. Un jeu sous forme normale peut aussi être la représentation simplifiée d'un jeu sous forme étendue, où les choix stratégiques des divers intervenants sont formulés à plusieurs reprises de manière non simultanée.

Chaque joueur est doté, dans un jeu bimatriciel, d'une aptitude à combiner en un seul jeu une variété de stratégies pures. Cette combinaison de stratégies pures, nommée *stratégie mixte*, permet au joueur de formuler ses préférences sous forme d'un vecteur x ou y

précisant la probabilité ou la fréquence avec laquelle chacune de ses stratégies serait jouée, si la même situation se répétait.

Le joueur I vise alors à maximiser son gain $x^t Ay$, en choisissant le vecteur de probabilité x dans \mathbb{R}^n , tandis que le joueur II tente de maximiser $x^t By$ en fixant son vecteur de probabilité y dans \mathbb{R}^m .

Un *équilibre* est une situation dans laquelle chaque intervenant a adopté un choix stratégique lui permettant de maximiser son gain étant donné le choix de l'autre. Aucun d'entre eux n'a donc intérêt à modifier unilatéralement son choix stratégique, puisqu'il en sortirait perdant.

Dans un jeu bimatriciel, un point d'équilibre est une paire de stratégies (\hat{x}, \hat{y}) où :

si deux ensembles $X(\hat{y})$ et $Y(\hat{x})$ sont définis tels que

$$\begin{aligned} X(\hat{y}) = \operatorname{argmax}_x x^t A \hat{y} & \qquad Y(\hat{x}) = \operatorname{argmax}_y \hat{x}^t B y \\ \text{s.à. } x^t \mathbf{1}_n = 1, & \qquad \text{et} \qquad \text{s.à. } \mathbf{1}_m^t y = 1, \\ x \geq 0, & \qquad \qquad \qquad y \geq 0, \end{aligned}$$

avec $\mathbf{1}_n$ et $\mathbf{1}_m$ deux vecteurs de dimensions respectives $(n \times 1)$ et $(m \times 1)$ et composés uniquement de 1, alors $\hat{x} \in X(\hat{y})$ et $\hat{y} \in Y(\hat{x})$.

Une stratégie est dite *dominée* lorsque, pour un joueur donné, tous les gains qu'elle pourrait rapporter sont inférieurs ou égaux aux gains apportés par une combinaison linéaire des autres stratégies (Myerson (49)). Une stratégie est dite *dominante*, par rapport à une stratégie *dominée*, si tous les gains qu'elle pourrait rapporter sont supérieurs ou égaux aux gains apportés par la stratégie *dominée*.

La formulation d'un jeu matriciel sous la forme d'un programme linéaire fût introduite par Gale, Kuhn et Tucker (1948). Bien que la question eut été abordée en premier lieu

par Von Neumann (1928), la première preuve de l'existence d'un point d'équilibre dans tout jeu fini à plusieurs joueurs, et par suite dans le cas restreint d'un jeu bimatriciel, fût l'oeuvre de Nash (50). Ce théorème a inspiré les chercheurs qui ont abordé le sujet par la suite et explique l'appellation courante "*équilibre de Nash*" définissant cet équilibre.

Mills (46), puis Mangasarian et Stone (41) ont étudié les conditions d'optimalité du système précédent dans le but de définir les conditions nécessaires et suffisantes d'équilibre. Ils ont alors introduit deux scalaires α et β , tels que $(\alpha, \beta) \in \mathbb{R}^2$, et reformulé le système à partir des problèmes duaux de la manière suivante :

$$\begin{array}{ll} \min_{\alpha} & \alpha \\ \text{s.à.} & \mathbf{1}_n \alpha \geq A\hat{y}, \end{array} \quad \text{et} \quad \begin{array}{ll} \min_{\beta} & \beta \\ \text{s.à.} & \beta \mathbf{1}_m^t \geq \hat{x}^t B. \end{array}$$

Une fois les contraintes de réalisabilité primales et duales combinées, il est possible d'affirmer qu'une paire (\hat{x}, \hat{y}) est une stratégie d'équilibre si et seulement si il existe deux scalaires $\hat{\alpha}$ et $\hat{\beta}$ tels que :

$$(\hat{x}, \hat{\beta}) \in X \equiv \{(x, \beta) \in \mathbb{R}^{n+1} : x^t B \leq \beta \mathbf{1}_m^t, x^t \mathbf{1}_n = 1, x \geq 0\},$$

$$(\hat{y}, \hat{\alpha}) \in Y \equiv \{(y, \alpha) \in \mathbb{R}^{m+1} : Ay \leq \mathbf{1}_n \alpha, \mathbf{1}_m^t y = 1, y \geq 0\}.$$

Les deux variables duales $\hat{\alpha}$ et $\hat{\beta}$ sont alors égales aux gains respectifs des joueurs I et II, lorsque ces conditions sont satisfaites, vu que les valeurs des objectifs des problèmes primaux et duaux sont égales :

$$\hat{x}^t A \hat{y} = \hat{\alpha} \quad \text{et} \quad \hat{x}^t B \hat{y} = \hat{\beta}.$$

Ces conditions, telles que Mills (46) et Mangasarian et Stone (41) l'ont montré, peuvent

être fusionnées en une seule contrainte bilinéaire :

$$\hat{x}^t(A + B)\hat{y} = \hat{\alpha} + \hat{\beta}.$$

Il est donc possible de définir un problème d'optimisation globale dont la solution optimale est un point d'équilibre :

$$\begin{aligned} \max_{x,y,\alpha,\beta} \quad & x^tAy + x^tBy - \alpha - \beta \\ \text{s.à.} \quad & \\ & Ay \leq \alpha \mathbf{1}_n, \\ & x^tB \leq \beta \mathbf{1}_m^t, \\ & x^t \mathbf{1}_n = 1, \\ & \mathbf{1}_m^t y = 1, \\ & x \geq 0, \\ & y \geq 0. \end{aligned}$$

En un point d'équilibre, la valeur de la fonction objectif est nulle puisque $(\hat{x}, \hat{y}, \hat{\alpha}, \hat{\beta})$ sont tels que :

$$\hat{x}^t(A + B)\hat{y} - (\hat{\alpha} + \hat{\beta}) = 0.$$

Mills (46) apporte aussi la preuve qu'un point d'équilibre dans un jeu bimatriciel est la solution d'un système d'inéquations linéaires, où certains vecteurs u et v doivent être binaires :

$$\begin{aligned} 0 \leq \alpha \mathbf{1}_n - Ay & \leq \mathbf{1}_n - u \leq \mathbf{1}_n - x, \\ 0 \leq \beta \mathbf{1}_m - xB & \leq \mathbf{1}_m - v \leq \mathbf{1}_m - y, \\ x \in X, \quad y \in Y, \quad u \in \{0, 1\}^n, \quad v \in \{0, 1\}^m. \end{aligned}$$

Exemple 1.0.1 Soit un jeu bimatriciel de dimension (4×3) dont les éléments des ma-

trices A et B prennent pour valeurs :

$$A = \begin{pmatrix} 3 & 2.5 & 5 \\ 4 & 0 & 2 \\ 2 & 3.5 & 1.5 \\ 4.5 & 0.5 & 5.5 \end{pmatrix} \quad \text{et} \quad B = \begin{pmatrix} 5 & 4.5 & 2.5 \\ 2 & 3 & 1 \\ 1 & 1.5 & 3.5 \\ 2.5 & 3.5 & 4 \end{pmatrix}.$$

Si I choisissait sa 4^{ème} stratégie et II choisissait sa 3^{ème} stratégie, $\hat{x}^t = (0, 0, 0, 1)$ et $\hat{y}^t = (0, 0, 1)$, les deux joueurs seraient en une situation d'équilibre et leurs gains respectifs seraient : $\hat{\alpha} = 5.5$ et $\hat{\beta} = 4$.

Pour le joueur I , la 2^{ème} stratégie est dominée par la 4^{ème} stratégie. Il est donc inutile de la considérer car I ne la choisira jamais.

1.1 Revue de la littérature

Un équilibre de Nash extrême (\hat{x}, \hat{y}) est tel que \hat{x} est un point extrême de l'ensemble $X(\hat{y})$ des meilleures répliques à la stratégie \hat{y} , et \hat{y} est un sommet de l'ensemble $Y(\hat{x})$ des meilleures répliques à la stratégie \hat{x} :

$$\hat{x} \in \text{ext}(X(\hat{y})) \quad \text{et} \quad \hat{y} \in \text{ext}(Y(\hat{x})),$$

où ext dénote l'ensemble des points extrêmes. Les équilibres d'un jeu bimatriciel forment un ensemble E qui est l'union d'un nombre fini de polytopes, parfois disjoints (45), chacun étant appelé un *sous-ensemble de Nash maximal*. Un équilibre de Nash extrême est un point extrême de l'un de ces polytopes.

Soit $\Sigma_x \subset \mathbb{R}^n$, un ensemble ayant comme éléments quelques stratégies extrêmes du

premier joueur. Vorob'ev (65) a prouvé que toute paire de stratégies (\hat{x}, \hat{y}) pour laquelle \hat{x} appartient à l'enveloppe convexe de Σ_x et où \hat{y} appartient à $Y(x)$ pour toute stratégie extrême x de Σ_x est un équilibre. L'ensemble E est ainsi l'union de ces ensembles de paires sur tous les sous-ensembles de stratégies extrêmes du premier joueur :

$$E = \bigcup_{\Sigma_x \subseteq \{x: (x,y) \in \text{ext}(E)\}} \{ \text{conv}(\Sigma_x) \times \bigcap_{x \in \Sigma_x} Y(x) \}.$$

Keiding (30) a montré que dans un jeu bimatriciel non-dégénéré, de dimensions $(m \times n)$, il existe au plus $K = \min\{\Phi(m, n + m), \Phi(n, m + n)\} - 1$ équilibres extrêmes avec :

$$\Phi(d, h) = \binom{h - \lfloor \frac{d-1}{2} \rfloor - 1}{\lfloor \frac{d}{2} \rfloor} + \binom{h - \lfloor \frac{d}{2} \rfloor - 1}{\lfloor \frac{d-1}{2} \rfloor},$$

$\Phi(d, h)$ étant le nombre maximal de sommets dans un polyèdre d -dimensionnel à h facettes.

Il faut noter cependant que cette borne supérieure sur le nombre maximal d'équilibres extrêmes n'est probablement pas serrée.

Pour le cas particulier d'un jeu bimatriciel non-dégénéré, de dimension $n \times n$, von Stengel (62) a démontré que le nombre maximal d'équilibres extrêmes était borné inférieurement par $(\psi(n) - 1)$ et supérieurement par $(\omega(n) - 1)$ avec :

$$\psi(n) = 0.949 \frac{2.414^n}{\sqrt{n}} \quad \text{et} \quad \omega(n) = 0.921 \frac{2.5981^n}{\sqrt{n}}.$$

L'énumération de tous les points extrêmes permet d'identifier complètement l'ensemble

E (Vorob'ev (65)) puisque tout point d'équilibre peut s'écrire comme une combinaison convexe de points d'équilibres extrêmes (Mangasarian (40)).

Les algorithmes énumérant tous les équilibres extrêmes d'un jeu bimatriciel ou polymatriciel sont très rares. Vorob'ev (65) fut le premier à en confectionner, ce qui donna ultérieurement lieu à une simplification de la part de Kuhn (36). L'idée de base était de générer à partir de la matrice B toutes les sous-matrices indépendantes et de vérifier si en ajoutant à chacune une colonne de 1, la stratégie correspondante engendrait un point d'équilibre.

Lemke et Howson (38) développèrent un algorithme pour la recherche d'équilibres de Nash. Cet algorithme requiert deux ensembles disjoints de stratégies pures (N et M). Aggarwal (1) a démontré que l'algorithme proposé par Lemke et Howson (38) ne pouvait être modifié afin d'énumérer tous les équilibres extrêmes, certains restant inaccessibles. Aggarwal (1) explique cette pathologie par le fait que l'union des chemins ayant $(n + m - k)$ ($k \in N \times M$) attributs pouvait être non-connexe.

Mangasarian (40) et Winkels (68) proposent aussi deux approches semblables dont l'idée est d'énumérer tous les sommets de X et de Y . Alors que le premier utilise la condition d'optimalité $x^t(A + B)y = \alpha + \beta$ pour vérifier si la paire (x, y) est un équilibre, le deuxième exploite les conditions des écarts complémentaires associées aux programmes primal et dual.

Mukhamediev (47) adopte quant à lui une approche s'inspirant de la formulation bilinéaire proposée par Mills (46) et Mangasarian et Stone (41). Il aborde la problématique en utilisant l'optimisation globale de la fonction :

$$\max_{(x,\beta) \in X, (y,\alpha) \in Y} x^t A y + x^t B y - \alpha - \beta$$

\Leftrightarrow

$$\max_{(x,\beta) \in X} -\beta + \left(\max_{(y,\alpha) \in Y} x^t(A+B)y - \alpha \right).$$

D'après les définitions de X et Y , on a :

$$\left\{ \begin{array}{l} Ay \leq \alpha \mathbf{1}_n, \\ x^t B \leq \beta \mathbf{1}_m^t, \\ x^t \mathbf{1}_n = 1, \\ \mathbf{1}_m^t y = 1, \\ x \geq 0, \\ y \geq 0, \end{array} \right. \Rightarrow \left\{ \begin{array}{l} x^t Ay \leq x^t \mathbf{1}_n \alpha, \\ x^t By \leq \beta \mathbf{1}_m^t y, \\ x^t \mathbf{1}_n = 1, \\ \mathbf{1}_m^t y = 1, \end{array} \right. \Rightarrow \left\{ \begin{array}{l} x^t Ay \leq \alpha, \\ x^t By \leq \beta, \end{array} \right.$$

$$\text{d'où } x^t Ay + x^t By - \alpha - \beta \leq 0.$$

En un point d'équilibre (\hat{x}, \hat{y}) , la valeur de cette fonction objectif sera donc nulle puisque les gains respectifs des deux joueurs seront : $\hat{\alpha} = \hat{x}^t A \hat{y}$ et $\hat{\beta} = \hat{x}^t B \hat{y}$.

Il existe dans un jeu bimatriciel $(2^n - 1) \times (2^m - 1)$ *supports* distincts. Un *support* est défini comme un ensemble de stratégies pures auxquelles sont assignées des probabilités strictement positives. Un *support* peut alors engendrer un ensemble vide de points d'équilibre, comme il pourrait en engendrer un seul ou parfois même un polytope.

Dickhaut et Kaplan (20) et McKelvey et McLennan (42) ont adopté une approche basée sur la vérification des conditions d'équilibre pour chacun des supports possibles et ont résolu des problèmes de dimension allant jusqu'à 8×8 , avec la possibilité de résoudre des problèmes de dimension 12×12 .

Le fait que toutes ces méthodes s'articulent, de manière plus ou moins explicite, autour du concept d'énumération de supports, les rend peu pratiques à utiliser lorsque la dimension du jeu n'est pas modeste. La nature combinatoire de cette énumération peut rendre le nombre de sommets à traiter très élevé.

La situation devient encore plus problématique en cas de dégénérescence, ce qui laisse la taille des problèmes susceptibles d'être résolus par ces méthodes très fortement limitée.

Les différentes méthodes linéaires de recherche d'équilibres de Nash ont été exposées par von Stengel (61). D'une part, il a montré comment la maximisation de fonctions de gains linéaires permettait de définir deux polyèdres. D'autre part, il a décrit une procédure algébrique de pivots complémentaires permettant d'obtenir un équilibre et il a clarifié la notion de dégénérescence dans les jeux bimatriciels. Von Stengel a également mentionné que la méthode lexicographique permettait d'étendre les algorithmes de pivots à cette dernière catégorie de jeux.

1.2 Algorithme de Lemke et Howson

Nous présentons dans cette section l'algorithme de Lemke et Howson (37) qui a pour but de trouver un équilibre de Nash dans un jeu bimatriciel. Soit $N = \{1, 2, \dots, n\}$ et $M = \{1, 2, \dots, m\}$. À toute paire de stratégies mixtes (x, y) correspondent des éléments de $N \times M$. Ces attributs dépendent des stratégies pures non-choisies par le joueur et des meilleures répliques pures de son adversaire.

Soient pour $i = 1, \dots, n$ et $j = 1, \dots, m$:

$$\begin{aligned} X(i) &= \{x \in X \mid x_i = 0\}, \\ X(j) &= \{x \in X \mid B_{.j}x \geq B_{.k}, \forall k \in M\}, \\ Y(i) &= \{y \in Y \mid A_i y \geq A_{.k}, \forall k \in N\}, \\ Y(j) &= \{y \in Y \mid y_j = 0\}, \end{aligned}$$

où A_i est la i^{me} ligne de A et $B_{.j}$ est la j^{eme} colonne de B . Ainsi, x a un attribut k si $x \in X(k)$, et y a un attribut k si $y \in Y(k)$, pour $k \in N \cup M$.

Lemke et Howson ont montré qu'une paire de stratégies (x, y) de $X \times Y$ est un équilibre

de Nash si et seulement si pour tout $k \in N \cup M : x \in X(k)$ ou $y \in Y(k)$, ou les deux à la fois.

Soient G_1 et G_2 deux graphes dont les sommets sont respectivement les points x de X ayant n attributs avec un sommet supplémentaire 0 de \mathbb{R}^n , et les points y de Y ayant m attributs avec un sommet supplémentaire 0 de \mathbb{R}^m . Dans G_1 , chaque paire de sommets ayant en commun $n - 1$ attributs est reliée par une arête, et similairement dans G_2 pour chaque paire de sommets ayant en commun $m - 1$ attributs.

L'algorithme commence à partir de l'équilibre artificiel $(x = 0, y = 0)$, suit le chemin sur lequel l'attribut k est manquant et s'arrête au premier équilibre de Nash rencontré.

Assimilables à une série de déplacements le long des arêtes de G_1 et G_2 , les itérations se font de telle manière que, si $(x, y) \in G_1 \times G_2$ est la paire courante de stratégies et si k est un attribut de x , alors y est remplacé par son voisin y' ayant k comme unique attribut non-commun avec lui et $m - 1$ attributs communs. Maintenant que k est à la fois attribut de x et de y , x est remplacé par son voisin x' , ayant k comme seul attribut non-commun avec lui et $n - 1$ attributs communs.

L'algorithme s'arrête lorsque x et y réunissent à eux deux tous les attributs $k \in N \times M$. L'équilibre atteint à la fin diffère selon le choix de l'attribut initial à la première itération.

Lemke et Howson (37) ont montré que l'ensemble des sommets et arêtes de $G_1 \times G_2$ ayant $n + m - k$ attributs ($k \in \{N \cup M\}$) consiste en un ensemble de chemins disjoints et de cycles, et que les points finaux à ces chemins sont les équilibres du jeu et l'équilibre artificiel $(0, 0)$.

Cet algorithme de Lemke et Howson est aussi à l'origine de l'une des premières avancées dans la recherche d'équilibres à un jeu polymatriciel à N joueurs.

1.3 Algorithme d'énumération des équilibres extrêmes *EEE* :

Audet *et al.* (6) proposent une approche énumérative plus sélective en étudiant seulement les sommets qui satisfont les conditions des écarts complémentaires :

$$\begin{cases} x^t A y = x^t \mathbf{1}_n \alpha, \\ x^t B y = \beta \mathbf{1}_m^t y, \end{cases} \Rightarrow \begin{cases} x^t (\mathbf{1}_n \alpha - A y) = 0, \\ (\beta \mathbf{1}_m - x^t B) y = 0. \end{cases}$$

Audet *et al.* (6) considèrent deux programmes linéaires à objectifs paramétrés :

$$P(y) \equiv \max_{(x, \beta) \in X} x^t A y - \beta,$$

$$Q(x) \equiv \max_{(y, \alpha) \in Y} x^t B y - \alpha.$$

L'algorithme *EEE* construit un arbre d'exploration où à chaque noeud est associée une paire de *sous-problèmes* linéaires à fonctions objectifs paramétrées, $\tilde{P}(y)$ et $\tilde{Q}(x)$. À chaque noeud est associée une variable x ou y représentant une stratégie réalisable, pour l'un ou l'autre des sous-problèmes courants.

Les programmes $\tilde{P}(y)$ et $\tilde{Q}(x)$ sont quasi-identiques à $P(y)$ et $Q(x)$, mais avec en plus un ensemble d'inégalités ou de contraintes de non-négativité qui sont mises à égalité.

Ainsi, Audet *et al.* (6) distinguent quatre formes possibles que peuvent prendre les problèmes $\tilde{P}(\cdot)$ et $\tilde{Q}(\cdot)$:

- $P^i(\cdot) \equiv \tilde{P}(\cdot)$ avec une contrainte $x_i = 0$ à la place de $x_i \geq 0$.
- $P_j(\cdot) \equiv \tilde{P}(\cdot)$ avec une contrainte $x^t B_j = \beta$ à la place de $x^t B_j \leq \beta$.
- $Q^j(\cdot) \equiv \tilde{Q}(\cdot)$ avec une contrainte $y_j = 0$ à la place de $y_j \geq 0$.
- $Q_i(\cdot) \equiv \tilde{Q}(\cdot)$ avec une contrainte $A_i y = \alpha$ à la place de $A_i y \leq \alpha$.

À chaque noeud de cet arbre, une optimisation de $\tilde{P}(y)$ et une autre de $\tilde{Q}(x)$ sont lancées

successivement (l'ordre dépend des cas) et trois sortes de résultats différents peuvent être obtenus :

- $\tilde{P}(\cdot)$ ou $\tilde{Q}(\cdot)$ est non-réalisable.
- $\tilde{P}(\cdot)$ et $\tilde{Q}(\cdot)$ sont réalisables mais un manque d'information ne permet pas de déduire un point d'équilibre.
- $\tilde{P}(\cdot)$ et $\tilde{Q}(\cdot)$ sont réalisables et un point d'équilibre peut être déduit à partir des informations réunies.

Mis à part le premier cas où le noeud courant sera éliminé, dans les deux autres il sera remplacé par ses descendants directs. Chaque *fil*s possède une seule différence par rapport à son *père* qui consiste en une contrainte supplémentaire pour $\tilde{P}(\cdot)$ ou $\tilde{Q}(\cdot)$ selon la règle de branchement choisie.

Cette règle de branchement se détermine à partir de deux paramètres τ_i et π_j , tels que

$$\tau_i = \begin{cases} x_i(\alpha - A_i.y), & \text{si } \tilde{P}(\cdot) \text{ n'a pas été changé en } P^i(\cdot) \\ & \text{et } \tilde{Q}(\cdot) \text{ n'a pas été changé en } Q_i(\cdot), \\ -1 & , \text{ sinon,} \end{cases}$$

et

$$\pi_j = \begin{cases} (\beta - x^t B_j)y_j, & \text{si } \tilde{Q}(\cdot) \text{ n'a pas été changé en } Q^j(\cdot) \\ & \text{et } \tilde{P}(\cdot) \text{ n'a pas été changé en } P_j(\cdot), \\ -1 & , \text{ sinon.} \end{cases}$$

En choisissant pour $i \in \{1, \dots, n\}$ et $j \in \{1, \dots, m\}$, les i et j qui maximisent les valeurs de τ_i et π_j , trois cas différents permettent de conclure sur la règle de branchement à adopter :

– Si $-1 \neq \tau_i \geq \pi_j$, le noeud courant aura deux descendants :

$$(y, P^i(\cdot), \tilde{Q}(\cdot)) \quad \text{et} \quad (x, Q_i(\cdot), \tilde{P}(\cdot)).$$

– Si $\tau_i < \pi_j$, le noeud courant aura deux descendants :

$$(x, Q^j(\cdot), \tilde{P}(\cdot)) \quad \text{et} \quad (y, P_j(\cdot), \tilde{Q}(\cdot)).$$

– Si $\tau_i = \pi_j = -1$, (x, y) est un équilibre extrême et le noeud courant aura quatre descendants :

$$\{(y, P^i(\cdot), \tilde{Q}(\cdot)) : x_i > 0\}, \quad \{(x, Q_i(\cdot), \tilde{P}(\cdot)) : A_i \cdot y < \alpha\},$$

$$\{(x, Q^j(\cdot), \tilde{P}(\cdot)) : y_j > 0\} \quad \text{et} \quad \{(y, P_j(\cdot), \tilde{Q}(\cdot)) : x^t B_j < \beta\}.$$

Quand la profondeur du noeud courant est supérieure à $n + m$, la solution (x, y) est un équilibre extrême car chacune des conditions des écarts complémentaires est satisfaite.

Audet *et al.* (6) démontrent que l'algorithme *EEE* énumère en un temps fini tous les équilibres extrêmes d'un jeu bimatriciel. Ils résolvent des problèmes aléatoirement générés, de taille allant jusqu'à (29×29) lorsque les dimensions sont égales et (700×5) lorsque la seconde dimension est fixée à 5.

1.4 Énumération des sous-ensembles de Nash maximaux

Soit NE est l'ensemble des équilibres de Nash extrêmes d'un jeu bimatriciel. Un sous-ensemble $T \subset NE$ est un sous-ensemble de Nash si et seulement si toute paire d'éléments T est interchangeable. Ceci veut dire que si $(x_1, y_1), (x_2, y_2) \in T$ alors $(x_1, y_2) \in T$ et $(x_2, y_1) \in T$. Un sous-ensemble de Nash T est dit maximal si il n'est pas contenu

dans un autre sous-ensemble de Nash (14).

En théorie des jeux, il n'existe aucun algorithme dont le but est d'énumérer spécifiquement les sous-ensembles de Nash maximaux. Cependant, un rapport technique de von Stengel (63) évoque la possibilité d'utiliser un algorithme d'énumération des cliques maximales dans un graphe G afin d'énumérer les sous-ensembles de Nash maximaux. Ceci étant dit, nous avons abouti au même résultat indépendamment de ce travail non-publié de von Stengel.

Suite à l'analyse de l'ensemble des équilibres de Nash extrêmes NE , nous obtenons un graphe $G = (V, E)$. Les équilibres de Nash extrêmes éléments de NE définissent l'ensemble V des noeuds de G . L'ensemble des arêtes E de G est tel que toute arête $e \in E$ est une connexion entre deux noeuds (équilibres de Nash extrêmes) de G , $E_{q_1} = (x_1, y_1) \in V$ et $E_{q_2} = (x_2, y_2) \in V$, si et seulement si E_{q_1} et E_{q_2} sont interchangeable.

Toute clique maximale du graphe G est un sous-graphe complet de G . Toute clique maximale de G correspond alors à un sous-ensemble d'équilibres de Nash extrêmes où tout équilibre de Nash extrême est interchangeable avec tous les autres. Ainsi, tout sous-ensemble de Nash maximal T correspond donc à une clique maximale de G .

Plusieurs articles en théorie des graphes ont étudié et résolu le problème d'énumération des cliques maximales dans un graphe. Nous avons choisi d'implanter une version en $C++$ de l'algorithme très connu de Bron et Kerbosch (15).

Exemple 1.4.1 Soit A et B les matrices des gains du jeu bimatriciel tiré d'un article de Borm et al. (14) :

$$A = \begin{pmatrix} -3 & -3 & -3 & -3 \\ 3 & 3 & 3 & -6 \\ -3 & -3 & -3 & -3 \end{pmatrix} \quad B = \begin{pmatrix} 3 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 2 \end{pmatrix}.$$

Le graphe G obtenu à partir de NE est représenté à la Figure 1.1.

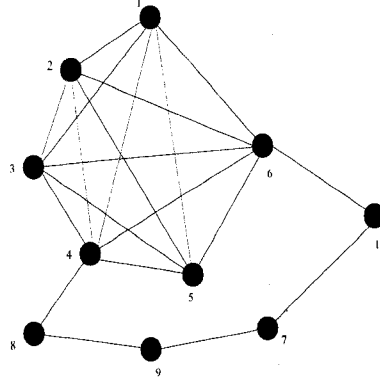


Figure 1.1 Graphe d'interchangeabilité $G = (V, E)$

En utilisant l'algorithme EEE (6), nous énumérons dix équilibres de Nash extrêmes (Tableau 1.1). Ainsi, ce jeu contient six sous-ensembles de Nash maximaux $T_1 = \{1, 2, 3, 4, 5, 6\}$, $T_2 = \{4, 8\}$, $T_3 = \{6, 10\}$, $T_4 = \{7, 9\}$, $T_5 = \{7, 10\}$ et $T_6 = \{8, 9\}$.

Le sous-ensemble T_1 est représenté à la Figure 1.2.

Tableau 1.1 Équilibres de Nash extrêmes pour l'exemple de Borm et al. (14)

| Eq. | x_1 | | | x_2 | | | | α | β |
|-----|-------|---|-----|-------|-----|-----|-----|----------|---------|
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 3 | 0 |
| 2 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 3 | 0 |
| 3 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 3 | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 | 1/3 | 2/3 | -3 | 0 |
| 5 | 0 | 1 | 0 | 0 | 1/3 | 0 | 2/3 | -3 | 0 |
| 6 | 0 | 1 | 0 | 1/3 | 0 | 0 | 2/3 | -3 | 0 |
| 7 | 2/3 | 0 | 1/3 | 0 | 0 | 0 | 1 | -3 | 2 |
| 8 | 1/3 | 0 | 2/3 | 0 | 0 | 1/3 | 2/3 | -3 | 2 |
| 9 | 1/3 | 0 | 2/3 | 0 | 0 | 0 | 1 | -3 | 2 |
| 10 | 2/3 | 0 | 1/3 | 1/3 | 0 | 0 | 2/3 | -3 | 2 |

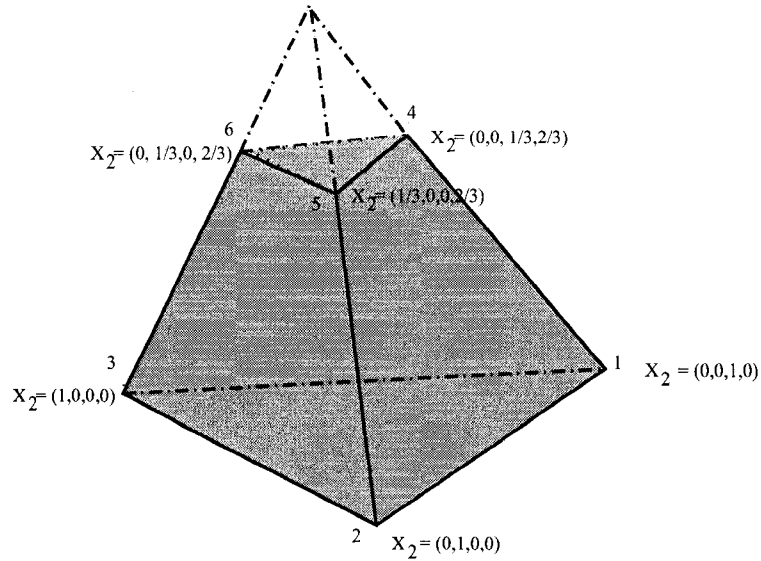


Figure 1.2 Sous-ensemble de Nash maximal T_1

1.5 Modélisation et formulation linéaire mixte d'un jeu bimatriciel

Cette section récapitule le travail de modélisation entrepris dans (4) et énonce la formulation d'un programme linéaire mixte 0 – 1, dont la résolution permet d'accomplir l'énumération des équilibres de Nash extrêmes d'un jeu bimatriciel.

La modélisation sous la forme d'un problème linéaire mixte d'un jeu bimatriciel passe par la transformation des contraintes de complémentarité en contraintes linéaires. Júdice et Mitra (29), puis Audet *et al.* (8), ont mis en place une manière efficace de transformer

les conditions d'écart complémentaires en introduisant des variables binaires :

$$\begin{cases} x^t A y = x^t \mathbf{1}_n \alpha, \\ x^t B y = \beta \mathbf{1}_m^t y, \end{cases} \Leftrightarrow \begin{cases} x^t (\mathbf{1}_n \alpha - A y) = 0, \\ (\beta \mathbf{1}_m - x^t B) y = 0, \end{cases} \Rightarrow \begin{cases} (\mathbf{1}_n \alpha - A y) \leq L_1 u, \\ (\beta \mathbf{1}_m - B^t x) \leq L_2 v, \\ x \leq \mathbf{1}_n - u, \\ y \leq \mathbf{1}_m - v, \end{cases}$$

où u et v sont deux vecteurs de variables binaires de dimensions respectives n et m , et L_1 et L_2 sont des constantes positives suffisamment grandes. Nous avons ainsi :

- Si $u_i = 0$, alors : $x_i \leq 1$ et $(\alpha - A_i y) = 0$, donc $x_i(\alpha - A_i y) = 0$.
- Si $v_j = 0$, alors : $y_j \leq 1$ et $(\beta - (B_j)^t x) = 0$, donc $y_j(\beta - (B_j)^t x) = 0$.
- Si $u_i = 1$, alors : $x_i = 0$ et $(\alpha - A_i y) \leq L_1$, donc $x_i(\alpha - A_i y) = 0$.
- Si $v_j = 1$, alors : $y_j = 0$ et $(\beta - (B_j)^t x) \leq L_2$, donc $y_j(\beta - (B_j)^t x) = 0$.

Où A_i et $(B_j)^t$ sont respectivement les $i^{\text{ème}}$ et $j^{\text{ème}}$ lignes de A et B^t .

Proposition 1.5.1 Choisir L_1 et L_2 de la manière suivante permet de satisfaire toutes les conditions de complémentarité données :

$$L_1 \geq \Gamma_1 \quad \text{et} \quad L_2 \geq \Gamma_2,$$

$\Gamma_1 = a_{\max} - a_{\min}$ et $\Gamma_2 = b_{\max} - b_{\min}$, étant les différences respectives entre les plus grands et les plus petits éléments de A et de B :

$$\begin{aligned} a_{\max} &= \max_{i \in N, j \in M} a_{ij}, & a_{\min} &= \min_{i \in N, j \in M} a_{ij}, \\ b_{\max} &= \max_{i \in N, j \in M} b_{ij}, & b_{\min} &= \min_{i \in N, j \in M} b_{ij}. \end{aligned}$$

Preuve 1.5.1 *Puisque nous avons :*

$$\begin{aligned} a_{\min} &\leq \alpha \leq a_{\max}, \\ b_{\min} &\leq \beta \leq b_{\max}, \\ a_{\min} &\leq A_i y \leq a_{\max}, \text{ pour } i = 1, \dots, n, \\ b_{\min} &\leq (B_j)^t x \leq b_{\max}, \text{ pour } j = 1, \dots, m, \end{aligned}$$

nous pouvons déduire les inégalités suivantes :

$$\begin{aligned} \alpha - A_i y &\leq a_{\max} - a_{\min}, \text{ pour } i = 1, \dots, n, \\ \beta - (B_j)^t x &\leq b_{\max} - b_{\min}, \text{ pour } j = 1, \dots, m. \end{aligned}$$

Nous obtenons ainsi : $(\mathbf{I}_n \alpha - Ay) \leq \Gamma_1 u$ et $(\beta \mathbf{I}_m - B^t x) \leq \Gamma_2 v$. En conclusion, choisir L_1 et L_2 de la manière suivante permet de transformer les conditions de complémentarité en contraintes linéaires : $L_1 \geq \Gamma_1$ et $L_2 \geq \Gamma_2$.

Toutes les conditions vérifiées par un équilibre de Nash peuvent donc s'exprimer de façon linéaire.

Proposition 1.5.2 *L'ensemble des équilibres de Nash d'un jeu bimatriciel est l'ensemble des paires de choix stratégiques $(x, y) \in \mathbb{R}^n \times \mathbb{R}^m$ telles que :*

$$\begin{aligned} x^t \mathbf{I}_n &= 1, & \mathbf{I}_m^t y &= 1, \\ x^t B - \beta \mathbf{I}_n &\leq 0, & Ay - \alpha \mathbf{I}_m &\leq 0, \\ x + u &\leq \mathbf{I}_n, & y + v &\leq \mathbf{I}_m, \\ (\mathbf{I}_n \alpha - Ay) - L_1 u &\leq 0, & (\beta \mathbf{I}_m - B^t x) - L_2 v &\leq 0, \\ x &\geq 0, & y &\geq 0, \\ u &\in \{0, 1\}^n, & v &\in \{0, 1\}^m. \end{aligned}$$

Preuve 1.5.2 *Un équilibre de Nash est défini comme un point $(\hat{x}, \hat{y}, \hat{\alpha}, \hat{\beta})$ tel que :*

$$(\hat{x}, \hat{\beta}) \in X \equiv \{(x, \beta) \in \mathbb{R}^{n+1} : x^t B \leq \beta \mathbf{I}_m, x^t \mathbf{I}_n = 1, x \geq 0\},$$

$$(\hat{y}, \hat{\alpha}) \in Y \equiv \{(y, \alpha) \in \mathbb{R}^{m+1} : Ay \leq \mathbf{I}_n \alpha, \mathbf{I}_m^t y = 1, y \geq 0\}.$$

Un équilibre de Nash extrême doit par ailleurs satisfaire les conditions des écarts complémentaires. Il suffit alors qu'un point (x, y, α, β) réponde aux conditions suivantes pour être reconnu comme équilibre de Nash extrême :

$$\begin{aligned} x^t \mathbf{I}_n &= 1, & \mathbf{I}_m^t y &= 1, \\ x^t B - \beta \mathbf{I}_n &\leq 0, & Ay - \alpha \mathbf{I}_m &\leq 0, \\ x + u &\leq \mathbf{I}_n, & y + v &\leq \mathbf{I}_m, \\ (\mathbf{I}_n \alpha - Ay) &\leq L_1 u, & (\beta \mathbf{I}_m - B^t x) &\leq L_2 v, \\ x &\geq 0, & y &\geq 0, \\ u &\in \{0, 1\}^n, & v &\in \{0, 1\}^m. \end{aligned}$$

Afin d'énumérer tous les équilibres de Nash extrêmes, il faudrait énumérer toutes les solutions extrêmes d'un programme linéaire mixte 0 – 1.

Corollaire 1.5.1 *L'énumération de tous les équilibres de Nash extrêmes dans un jeu bimatriciel est équivalente à l'énumération de toutes les solutions extrêmes d'un programme linéaire mixte 0 – 1, sujet aux contraintes précédentes, et ayant un objectif linéaire, par exemple :*

$$\min_{\alpha, \beta, x, y, u, v} f(\alpha, \beta) = \alpha + \beta$$

ou encore :

$$\min_{\alpha, \beta, x, y, u, v} f(\alpha, \beta) = 0.$$

Peu importe l'expression linéaire de l'objectif du programme, l'algorithme $E\chi MIP$ présenté au quatrième chapitre permettra d'énumérer tous les équilibres de Nash extrêmes. Il aura pour objet d'identifier toutes les solutions extrêmes satisfaisant les contraintes du programme mixte 0–1. Ces contraintes sont au nombre de $(2+3(n+m))$, sans compter les contraintes de non-négativité.

Exemple 1.5.1 Soit le jeu bimatriciel suivant :

$$A = \begin{pmatrix} 3 & 2.5 & 5 \\ 4 & 0 & 2 \\ 2 & 3.5 & 1.5 \\ 4.5 & 0.5 & 5.5 \end{pmatrix} \quad \text{et} \quad B = \begin{pmatrix} 5 & 4.5 & 2.5 \\ 2 & 3 & 1 \\ 1 & 1.5 & 3.5 \\ 2.5 & 3.5 & 4 \end{pmatrix}.$$

Selon la proposition (1.5.2), l'expression du programme linéaire mixte 0–1 correspondant à ce jeu est :

$$\begin{aligned} & \min_{\alpha, \beta, x, y, u, v} f(\alpha, \beta) \\ & \text{s.à.} \\ & x_1 + x_2 + x_3 + x_4 = 1, & y_1 + y_2 + y_3 = 1, \\ & x_1 + u_1 \leq 1, & y_1 + v_1 \leq 1, \\ & x_2 + u_2 \leq 1, & y_2 + v_2 \leq 1, \\ & x_3 + u_3 \leq 1, & y_3 + v_3 \leq 1, \\ & x_4 + u_4 \leq 1, \\ & \alpha - 3y_1 - 2.5y_2 - 5y_3 - 5.5u_1 \leq 0, & \beta - 5x_1 - 2x_2 - x_3 - 2.5x_4 - 4v_1 \leq 0, \\ & \alpha - 4y_1 - 2y_3 - 5.5u_2 \leq 0, & \beta - 4.5x_1 - 3x_2 - 1.5x_3 - 3.5x_4 - 4v_2 \leq 0, \\ & \alpha - 2y_1 - 3.5y_2 - 1.5y_3 - 5.5u_3 \leq 0, & \beta - 2.5x_1 - x_2 - 3.5x_3 - 4x_4 - 4v_3 \leq 0, \\ & \alpha - 4.5y_1 - 0.5y_2 - 5.5y_3 - 5.5u_4 \leq 0, \\ & -\alpha + 3y_1 + 2.5y_2 + 5y_3 \leq 0, & -\beta + 5x_1 + 2x_2 + x_3 + 2.5x_4 \leq 0, \\ & -\alpha + 4y_1 + 2y_3 \leq 0, & -\beta + 4.5x_1 + 3x_2 + 1.5x_3 + 3.5x_4 \leq 0, \\ & -\alpha + 2y_1 + 3.5y_2 + 1.5y_3 \leq 0, & -\beta + 2.5x_1 + x_2 + 3.5x_3 + 4x_4 \leq 0, \\ & -\alpha + 4.5y_1 + 0.5y_2 + 5.5y_3 \leq 0, \\ & x \geq 0, & y \geq 0, \\ & u \in \{0, 1\}^4, & v \in \{0, 1\}^3. \end{aligned}$$

1.6 Discussion

Dans ce chapitre, nous avons effectué une synthèse des principales formulations mathématiques d'un jeu bimatriciel. Nous avons exposé les deux algorithmes de Lemke et Howson (37) et *EEE* (6), respectivement pour la recherche d'un équilibre de Nash et l'énumération de tous les équilibres de Nash extrêmes d'un jeu bimatriciel. Ensuite, nous avons montré que le problème d'énumération des sous-ensembles de Nash maximaux peut être résolu avec un algorithme d'énumération des cliques maximales d'un graphe représentant les équilibres de Nash extrêmes du jeu bimatriciel. Nous avons finalement linéarisé les conditions de complémentarité en utilisant la méthode de Júdice et Mitra (29) et nous avons proposé un programme linéaire mixte 0 – 1 dont les solutions extrêmes sont des équilibres de Nash extrêmes du jeu bimatriciel. Nous présenterons au quatrième chapitre l'algorithme *E χ Mip* qui utilise cette formulation linéaire mixte 0 – 1 afin d'énumérer les équilibres de Nash extrêmes d'un jeu bimatriciel.

CHAPITRE 2

LES JEUX POLYMATRICIELS

Un jeu *polymatriciel* est une rencontre de n joueurs ($n \geq 2$), dans un contexte non-coopératif et sous forme normale (stratégique). Contrairement à un simple jeu stratégique à n joueurs, un jeu polymatriciel est doté de $n(n - 1)$ matrices des gains partiels ; soit deux matrices pour chaque paire de joueurs. Soit $N = \{1, \dots, n\}$ l'ensemble des joueurs où chaque joueur i dispose d'un ensemble fini de stratégies pures S_i : $S_i = \{s_i^1, \dots, s_i^{m_i}\}$ avec $|S_i| = m_i$.

Si le joueur i choisit une de ses stratégies s_i et le joueur j choisit une de ses stratégies s_j , il est possible d'assigner un gain $a_{ij}(s_i, s_j)$ tel que pour n'importe quel choix de stratégies pures (s_1, \dots, s_n) par les n joueurs, le gain total du joueur i à l'issue du jeu est donné par :

$$A_i(s_1, \dots, s_n) = \sum_{j \neq i} a_{ij}(s_i, s_j).$$

Il y a lieu alors de définir la matrice $A_{ij} = (a_{ij}^{kl})$, de dimension $m_i \times m_j$, qui n'est autre que la matrice des gains partiels du joueur i , résultant des choix stratégiques purs des joueurs i et j . L'influence relative du choix stratégique de n'importe quel participant au jeu sur n'importe quel autre est toujours la même, peu importe ce que les autres joueurs choisissent.

Comme mentionné au premier chapitre, une stratégie mixte pour un joueur i est une distribution de probabilités ou de fréquences sur ses stratégies pures. Une stratégie mixte est un vecteur $(X_i)^t = (x_i^1, \dots, x_i^{m_i})$, tel que pour tout $k \in \{1, \dots, m_i\}$, x_i^k est la probabilité que le joueur i joue sa stratégie $s_i^k \in S_i$.

Toutes les stratégies mixtes possibles du joueur i définissent l'ensemble \tilde{S}_i tel que : $\tilde{S}_i = \{x_i : \mathbf{1}_{m_i}^t x_i = 1 \text{ et } x_i \geq 0\}$. Le gain que pourrait tirer le joueur i à l'issue du jeu s'écrit :

$$R_i(X) = (X_i)^T \sum_{j \neq i} A_{ij} X_j = \sum_{j \neq i} \sum_{k=1}^{m_i} \sum_{l=1}^{m_j} a_{ij}^{kl} x_i^k x_j^l.$$

Dans un jeu polymatriciel, un n -tuple $X^* = (X_1^*, \dots, X_n^*)$ est un équilibre de Nash si et seulement si pour n'importe quel autre ensemble $X = (X_1, \dots, X_n)$, l'inégalité suivante est observée :

$$(X_i^*)^T \sum_{j \neq i} A_{ij} X_j \geq (X_i)^T \sum_{j \neq i} A_{ij} X_j, \quad \text{pour } i = 1, \dots, n. \quad (2.0.1)$$

Cela équivaut aussi à la condition suivante :

$$R_i(X^*) \geq R_i(X_{N-\{i\}}^*, x_i), \quad \forall x_i \in \tilde{S}_i \quad \text{et pour } i = 1, \dots, n,$$

$$\text{avec } (X_{N-\{i\}}^*, x_i) = (x_1^*, \dots, x_{i-1}^*, x_i, x_{i+1}^*, \dots, x_n^*).$$

Ainsi, x_i^* n'est considéré comme la meilleure réplique de i par rapport à X , que lorsque :

$$R_i(X_{N-\{i\}}^*, x_i^*) = \max_{x_i \in \tilde{S}_i} R_i(X_{N-\{i\}}^*, x_i).$$

Pour un ensemble de stratégies mixtes X_1, \dots, X_n et pour tout joueur i , soit :

$$\alpha_i = (X_i^*)^T \sum_{j \neq i} A_{ij} X_j. \quad (2.0.2)$$

Soit aussi e_r^i , une colonne de dimension $(m_i \times 1)$, dont toutes les composantes sont nulles sauf la $r^{\text{ème}}$ égale à 1. L'inégalité (2.0.1) est valide pour tout X_i , y compris pour $X_i = e_r^i$ avec $(r = 1, \dots, m_i)$. Howson (24) a prouvé que, dans ce cas, l'équation (2.0.2) n'est

valide que si :

$$\alpha_i \mathbf{1}_{m_i} \geq \sum_{j \neq i} A_{ij} X_j, \quad \text{pour } i = 1, \dots, n, \quad (2.0.3)$$

où $\mathbf{1}_{m_i}$ est une colonne de dimension $(m_i \times 1)$, dont tous les éléments sont égaux à 1.

Ceci permet d'affirmer que :

$$(X_i^*)^T \alpha_i \mathbf{1}_{m_i} \geq (X_i^*)^T \sum_{j \neq i} A_{ij} X_j \iff (X_i^*)^T \left(\sum_{j \neq i} A_{ij} X_j - \alpha_i \mathbf{1}_{m_i} \right) = 0. \quad (2.0.4)$$

Ainsi, un ensemble X^* associé à un ensemble de scalaires $\alpha_1, \dots, \alpha_n$ permet de satisfaire les deux conditions (2.0.3) et (2.0.4) est un équilibre, puisque la condition nécessaire et suffisante est satisfaite :

$$(X_i)^T \alpha_i \mathbf{1}_{m_i} \geq (X_i)^T \sum_{j \neq i} A_{ij} X_j \iff (X_i^*)^T \sum_{j \neq i} A_{ij} X_j \geq (X_i)^T \sum_{j \neq i} A_{ij} X_j.$$

Ce premier résultat permet de conclure que les réels α_i ($i \in N$) ne sont autres que les gains totaux de chacun des joueurs à un équilibre de Nash (Quintas (54)). La relation (2.0.4) est une première condition de complémentarité. De la même manière, soient :

$$\begin{aligned} Y_i &= \alpha_i \mathbf{1}_{m_i} - \sum_{j \neq i} A_{ij} X_j, & \text{pour } i = 1, \dots, n, \\ \mu_i &= \mathbf{1}_{m_i}^t X_i^* - 1, & \text{pour } i = 1, \dots, n. \end{aligned}$$

En utilisant alors les résultats (2.0.2), (2.0.3) et (2.0.4), ainsi que le fait que X_i^* est un vecteur de probabilité, les conditions suivantes peuvent être formulées :

$$\begin{aligned} X_i &\geq 0, Y_i \geq 0, (X_i^*)^T Y_i = 0, & \text{pour } i = 1, \dots, n, \\ \mu_i &\geq 0, \alpha_i \geq 0, \mu_i \alpha_i = 0, & \text{pour } i = 1, \dots, n. \end{aligned}$$

La recherche d'un équilibre de Nash dans un jeu polymatriciel est équivalente, d'après

ces conditions de complémentarité, à la recherche de solution au *Problème de Complémentarité Linéaire (PCL)* (18; 70). Il s'agit de trouver un vecteur Z tel que : $Z \geq 0$, $W = Q + MZ \geq 0$, $Z^T W = 0$, où Q et M sont choisis convenablement.

Exemple 2.0.1 Soit un jeu polymatriciel à trois joueurs, de dimension $(2 \times 2 \times 2)$, où A , B et C sont les matrices des gains respectifs des joueurs I, II et III.

$$A = \left(\begin{array}{cc|cc} 2 & 0 & 3 & 5 \\ 3 & 4 & 1 & 4 \end{array} \right)$$

$$B = \left(\begin{array}{cc|cc} 0 & 5 & 3 & 2 \\ 1 & 4 & 6 & 7 \end{array} \right)$$

$$C = \left(\begin{array}{cc|cc} 4 & 1 & 6 & 0 \\ 1 & 2 & 5 & 3 \end{array} \right)$$

Le point : $\hat{x}^t = (0, 1)$, $\hat{y}^t = (0, 1)$ et $\hat{z}^t = (0, 1)$ est un équilibre de Nash extrême.

Les gains respectifs, $\hat{\alpha}$, $\hat{\beta}$ et $\hat{\gamma}$, des trois joueurs sont en ce point :

$$\hat{\alpha} = \hat{x}^t \times A \times \begin{pmatrix} \hat{y} \\ \hat{z} \end{pmatrix} = 8$$

$$\hat{\beta} = \hat{y}^t \times B \times \begin{pmatrix} \hat{x} \\ \hat{z} \end{pmatrix} = 11$$

$$\hat{\gamma} = \hat{z}^t \times C \times \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = 5.$$

Pour le joueur II, la 1^{ère} stratégie est dominée par la 2^{ème} stratégie. Une explication plus détaillée est donnée dans la troisième section de ce chapitre.

2.1 Revue de la littérature

Le problème de complémentarité linéaire a été résolu, pour certaines classes de matrices de données, par Cottle et Dantzig (17; 18) et Lemke et Howson (37; 38). Les jeux polymatriciels n'ont cependant pas les mêmes classes de données que celles des problèmes qui ont déjà été résolus.

Yanovskaya (70) a abordé le problème de recherche d'un équilibre d'un jeu polymatriciel par la résolution d'un *Problème de Complémentarité Linéaire* (29; 54; 24). La méthode des *pivots complémentaires* a été utilisée pour résoudre le (*PCL*), aussi bien par Yanovskaya (70), que par Howson (24), Eaves (21) et Howson et Rosenthal (25).

Wilson (67) a démontré que l'algorithme de Lemke et Howson (37) pouvait être directement étendu dans le but de construire un équilibre d'un jeu polymatriciel. L'approche de Wilson (67) découle d'une généralisation aux jeux polymatriciels de la règle selon laquelle un chemin quasi-complémentaire (réunissant toutes les conditions de complémentarité sauf une) conduit impérativement à un équilibre.

Le problème de recherche d'un point situé sur un chemin quasi-complémentaire dans un jeu polymatriciel à n joueurs serait ainsi résolu, une fois qu'un équilibre dans un jeu polymatriciel à $(n - 1)$ joueurs aura été identifié. D'après Wilson (67), construire un équilibre à un jeu à n joueurs revient alors à construire successivement une série d'équilibres à des jeux à k joueurs, $k \in \{1, 2, \dots, n - 1\}$.

2.2 Algorithme des pivots complémentaires

L'idée d'utiliser la méthode des pivots complémentaires dans le but de trouver un équilibre à un jeu polymatriciel découle de la formulation algébrique du problème de traversée d'un polyèdre le long de ses arêtes.

Les contraintes définissant le polyèdre sont représentées par des équations linéaires, où certaines variables sont non-négatives. Dans le cas d'un jeu bimatriciel, ces contraintes pourraient s'écrire :

$$\begin{aligned} Ay' + r &= \mathbf{1}_n, \\ B^t x' + s &= \mathbf{1}_m, \end{aligned} \tag{2.2.5}$$

où $x', y', r, s \geq 0$, et $r \in \mathbb{R}^n$ et $s \in \mathbb{R}^m$ sont des vecteurs de variables d'écarts.

Par extension de la formulation (2.2.5) aux jeux polymatriciels, le lien entre la méthode des pivots complémentaires et le problème d'énumération d'équilibres devient plus clair.

Ce système est de la forme : $Cz = q$, pour une matrice C , une partie droite q et un vecteur z de variables non-négatives.

Pour le *Problème de Complémentarité Linéaire*, si une solution donnée (W, Z) à l'équation $W = Q + MZ$, avec $Z \geq 0$ et $W \geq 0$, possède λ valeurs de i pour lesquelles $z_i w_i > 0$, alors λ est le niveau de non-complémentarité de la solution. Par son adaptation de la méthode des pivots complémentaires au (*PCL*), Howson (24) avait pour objectif de diminuer graduellement λ jusqu'à zéro, grâce à une série de pivots par blocs. Ainsi, le point atteint lorsque λ est égal à 0 est un équilibre.

L'algorithme des pivots complémentaire, tel que défini par Howson (24), se résume en une série finie de pivots par blocs dans le but d'accroître à chaque itération, en mettant des variables hors-base et d'autres dans la base, le nombre de conditions de complé-

mentarité satisfaites jusqu'à ce que toutes ces conditions soient réunies en un seul point. Dans ce cas, le point (X_1, \dots, X_n) atteint est un équilibre.

Howson (24) donne une justification complète de la bonne marche de la procédure, ainsi qu'une preuve quant à l'existence d'au moins un équilibre dans un jeu polymatriciel.

Par ailleurs, von Stengel (62) donne une version pour les jeux bimatriels de l'algorithme des pivots complémentaires accompagnée d'un exemple simple qui peut aider à se familiariser avec la procédure.

2.3 Élimination des stratégies dominées

Dans un jeu bimatriel ou polymatriciel, une stratégie est dite *dominée* (49), pour un joueur donné, si le gain qu'elle pourrait rapporter est inférieur ou égal au gain rapporté par une autre de ses stratégies ou une combinaison linéaire de toutes ses stratégies. Il est intéressant de distinguer entre les stratégies fortement dominées et les stratégies faiblement dominées.

2.3.1 Les stratégies fortement dominées

Il est utile de procéder à une élimination des stratégies fortement dominées avant de se lancer dans l'énumération complète des équilibres de Nash extrêmes d'un jeu bimatriel ou polymatriciel.

Les stratégies fortement dominées portent dans ce contexte des informations sans intérêt, qui ne peuvent qu'encombrer la démarche de résolution. L'élimination des stratégies fortement dominées est similaire à un filtrage d'information.

Pour un joueur donné, une stratégie est dite *fortement dominée* si, dans aucun cas, elle

ne pourrait représenter une meilleure réplique aux choix stratégiques des autres joueurs.

Dans le contexte d'un jeu polymatriciel, si la l stratégie du joueur i est fortement dominée, alors il existe un vecteur de scalaires positifs $\lambda = (\lambda_1, \dots, \lambda_{l-1}, \lambda_{l+1}, \dots, \lambda_{m_i})^t$, tel que pour toute combinaison $S - \{S_i\}$ de choix stratégiques des autres joueurs :

$$\sum_{q=1, q \neq l}^{m_i} \lambda_q = 1, \quad \text{et} \quad \sum_{q=1, q \neq l}^{m_i} \left(\sum_{j=1, j \neq i}^n \lambda_q a_{ij}^{qs_j^k} \right) > \sum_{j=1, j \neq i}^n a_{ij}^{ls_j^k},$$

$$\forall s_j^k \in S_j \text{ et } \forall S_j \in S - \{S_i\}.$$

Ces conditions à respecter résument dans le contexte d'énumération des équilibres extrêmes, la définition donnée par Myerson (49) d'une stratégie fortement dominée dans un jeu à n joueurs.

Exemple 2.3.1 Dans le jeu bimatriciel présenté comme exemple au premier chapitre :

$$A = \begin{pmatrix} 3 & 2.5 & 5 \\ 4 & 0 & 2 \\ 2 & 3.5 & 1.5 \\ 4.5 & 0.5 & 5.5 \end{pmatrix} \quad \text{et} \quad B = \begin{pmatrix} 5 & 4.5 & 2.5 \\ 2 & 3 & 1 \\ 1 & 1.5 & 3.5 \\ 2.5 & 3.5 & 4 \end{pmatrix}.$$

Pour le joueur I , soit le vecteur $\lambda = (\lambda_1, \lambda_3, \lambda_4)^t = (0, 0, 1)^t$. Nous avons alors :

$$\lambda_1 + \lambda_3 + \lambda_4 = 1$$

et

$$3\lambda_1 + 2\lambda_3 + 4.5\lambda_4 = 4.5 > 4,$$

$$2.5\lambda_1 + 3.5\lambda_3 + 0.5\lambda_4 = 0.5 > 0,$$

$$5\lambda_1 + 1.5\lambda_3 + 5.5\lambda_4 = 5.5 > 2.$$

La 2^{ème} stratégie du joueur I est donc fortement dominée par la 4^{ème} stratégie.

L'élimination de cette stratégie permet d'obtenir un jeu de taille (3×3) :

$$A' = \begin{pmatrix} 3 & 2.5 & 5 \\ 2 & 3.5 & 1.5 \\ 4.5 & 0.5 & 5.5 \end{pmatrix} \quad \text{et} \quad B' = \begin{pmatrix} 5 & 4.5 & 2.5 \\ 1 & 1.5 & 3.5 \\ 2.5 & 3.5 & 4 \end{pmatrix}.$$

Exemple 2.3.2 Dans le jeu polymatriciel à trois joueurs présenté comme à l'exemple (2.0.1) :

$$A = \left(\begin{array}{cc|cc} 2 & 0 & 3 & 5 \\ 3 & 4 & 1 & 4 \end{array} \right) \quad B = \left(\begin{array}{cc|cc} 0 & 5 & 3 & 2 \\ 1 & 4 & 6 & 7 \end{array} \right) \quad C = \left(\begin{array}{cc|cc} 4 & 1 & 6 & 0 \\ 1 & 2 & 5 & 3 \end{array} \right)$$

Pour le joueur II, soit le réel $\lambda_2 = 1$. Nous avons alors :

$$\begin{aligned} \lambda_2 \times (1 + 6) &= 7 > 0 + 3, \\ \lambda_2 \times (1 + 7) &= 8 > 0 + 2, \\ \lambda_2 \times (4 + 6) &= 10 > 5 + 3, \\ \lambda_2 \times (4 + 7) &= 11 > 5 + 2. \end{aligned}$$

La 1^{ère} stratégie du joueur II est donc fortement dominée par sa 2^{ème} stratégie. L'élimination de cette stratégie permet d'obtenir un jeu de taille $(2 \times 1 \times 2)$:

$$A' = \left(\begin{array}{c|cc} 0 & 3 & 5 \\ 4 & 1 & 4 \end{array} \right) \quad B' = \left(\begin{array}{c|cc} 1 & 4 & 6 \\ 1 & 4 & 7 \end{array} \right) \quad C' = \left(\begin{array}{c|cc} 4 & 1 & 0 \\ 1 & 2 & 3 \end{array} \right).$$

Récursivement pour le joueur I, soit $\lambda'_2 = 1$. Nous avons alors :

$$\lambda'_2 \times (4 + 1) = 5 > 0 + 3,$$

$$\lambda'_2 \times (4 + 4) = 8 > 0 + 5.$$

La 1^{ère} stratégie du joueur I est donc aussi fortement dominée par sa 2^{ème} stratégie.
L'élimination de cette stratégie permet d'obtenir un jeu de taille $(1 \times 1 \times 2)$:

$$A'' = \left(4 \mid 1 \quad 4 \right) \quad B'' = \left(4 \mid 6 \quad 7 \right) \quad C'' = \left(\begin{array}{c|c} 1 & 0 \\ \hline 2 & 3 \end{array} \right)$$

Enfin pour le joueur III, soit $\lambda''_2 = 1$. Nous avons alors :

$$\lambda'_2 \times (2 + 3) = 5 > 1 + 0,$$

La 1^{ère} stratégie du joueur III est donc aussi fortement dominée par sa 2^{ème} stratégie.
L'élimination de cette stratégie permet d'obtenir un jeu de taille $(1 \times 1 \times 1)$:

$$A''' = \left(4 \mid 4 \right) \quad B''' = \left(4 \mid 7 \right) \quad C''' = \left(2 \mid 3 \right)$$

Il devient évident que le seul déroulement possible de ce jeu donne un équilibre de Nash extrême. Il est très utile de noter que les stratégies éliminées en un deuxième et troisième temps n'étaient pas dominées de manière apparente dans le jeu initial.

L'élimination des stratégies fortement dominées permet donc d'alléger le jeu bimatriciel ou polymatriciel et de faciliter l'énumération de tous les équilibres de Nash extrêmes.

Proposition 2.3.1 *Pour un jeu polymatriciel à n joueurs, l'identification récursive des stratégies fortement dominées peut être réalisée en un nombre d'itérations maximal de l'ordre de :* $M^2 = \left(\sum_{i=1}^n m_i \right)^2$.

Preuve 2.3.1 *Le nombre maximal d'itérations nécessaire à l'identification récursive de toutes les stratégies fortement dominées est égal à la somme des itérations effectuées. Si à chaque passage complet par toutes les stratégies, une seule stratégie fortement dominée est identifiée, le nombre maximal d'itérations est égal à la somme des termes d'une suite arithmétique :*

$$M + (M - 1) + (M - 2) + \dots + N \simeq M(M + 1)/2.$$

Le nombre maximal d'itérations nécessaire à l'identification récursive de toutes les stratégies fortement dominées dans un jeu polymatriciel est donc de l'ordre de :

$$M^2 = \left(\sum_{i=1}^n m_i \right)^2.$$

Tous les équilibres de Nash extrêmes énumérables dans le jeu original seront énumérés dans le jeu résiduel (sans stratégies dominées), car une stratégie n'est éliminée que si elle est fortement dominée par une combinaison linéaire des autres stratégies du même joueur (Myerson(49)). Un joueur rationnel n'aurait donc jamais intérêt à choisir une telle stratégie, puisque, dans aucun cas, elle ne pourrait représenter une meilleure réplique au choix stratégique de l'autre ou des autres joueurs.

2.3.2 Les stratégies faiblement dominées

Dans le contexte d'un jeu polymatriciel à n joueurs, si la $l^{\text{ème}}$ stratégie du joueur i est faiblement dominée, alors il existe un vecteur de scalaires positifs :

$$\lambda = (\lambda_1, \dots, \lambda_{l-1}, \lambda_{l+1}, \dots, \lambda_{m_i})^t,$$

tel que pour toute combinaison $S - \{S_i\}$ de choix stratégiques des autres joueurs :

$$\sum_{q=1, q \neq i}^{m_i} \lambda_q = 1, \quad \text{et} \quad \sum_{q=1, q \neq i}^{m_i} \left(\sum_{j=1, j \neq i}^n \lambda_q a_{ij}^{qs_j^k} \right) \geq \sum_{j=1, j \neq i}^n a_{ij}^{ls_j^k},$$

$$\forall s_j^k \in S_j \text{ et } \forall S_j \in S - \{S_i\}.$$

Ces conditions à respecter résument aussi, dans notre contexte d'énumération des équilibres de Nash extrêmes, la définition donnée par Myerson (49) d'une stratégie faiblement dominée dans un jeu à n joueurs.

La détection des stratégies faiblement dominées permet de filtrer le flot informationnel contenu dans un jeu bimatriciel ou polymatriciel mais n'offre aucune garantie quant à l'énumération complète et efficace de tous les équilibres de Nash extrêmes.

Un joueur rationnel serait indifférent entre le choix d'une stratégie faiblement dominée et le choix parmi ses autres stratégies de la combinaison linéaire qui la domine (Myerson (49)). Il serait donc préjudiciable de procéder à une élimination des stratégies faiblement dominées avant de se lancer dans une énumération complète des équilibres de Nash extrêmes dans un jeu bimatriciel ou polymatriciel.

2.4 Modélisation et formulation linéaire mixte d'un jeu polymatriciel

La transformation des conditions de complémentarité (2.0.4) en contraintes linéaires, grâce à des variables binaires, permet d'aboutir à une nouvelle formulation d'un jeu polymatriciel sous la forme d'un programme linéaire mixte 0 – 1.

Pour $i = 1, \dots, n$, les conditions de complémentarité s'écrivent :

$$(X_i)^T \left(\alpha_i \mathbf{1}_{m_i} - \sum_{j=1, j \neq i}^n A_{ij} X_j \right) = 0 \iff \begin{cases} \alpha_i \mathbf{1}_{m_i} - \sum_{j=1, j \neq i}^n A_{ij} X_j - L_i U_i \leq 0, \\ X_i + U_i \leq \mathbf{1}_{m_i}. \end{cases}$$

Le choix des paramètres L_i adéquats est défini par la proposition suivante. En effet, ces paramètres doivent être plus grands ou égaux à la somme des différences entre les plus grands et les plus petits gains respectifs des joueurs.

Proposition 2.4.1 Choisir L_i de la manière suivante, permet de satisfaire toutes les conditions de complémentarité originales :

$$L_i \geq \sum_{j=1, j \neq i}^n \Gamma_{ij}, \quad \text{pour } i = 1, \dots, n,$$

où $\Gamma_{ij} = a_{ij}^{max} - a_{ij}^{min}$, est la différence entre le plus grand et le plus petit élément de A_{ij} :

$$\begin{aligned} a_{ij}^{max} &= \max_{k \in m_i, l \in m_j} a_{ij}^{kl}, \\ a_{ij}^{min} &= \min_{k \in m_i, l \in m_j} a_{ij}^{kl}. \end{aligned}$$

Preuve 2.4.1 Puisque pour $i = 1, \dots, n$, nous avons :

$$\begin{aligned} \alpha_i &\geq \sum_{j=1, j \neq i}^n a_{i,j}^{min} \quad \text{et} \quad \sum_{j=1, j \neq i}^n A_{ij} X_j \geq \sum_{j=1, j \neq i}^n a_{i,j}^{min}, \\ \alpha_i &\leq \sum_{j=1, j \neq i}^n a_{i,j}^{max} \quad \text{et} \quad \sum_{j=1, j \neq i}^n A_{ij} X_j \leq \sum_{j=1, j \neq i}^n a_{i,j}^{max}, \end{aligned}$$

alors l'inégalité suivante peut être formulée :

$$\left(\alpha_i \mathbf{1}_{m_i} - \sum_{j=1, j \neq i}^n A_{ij} X_j \right) \leq \left(\sum_{j=1, j \neq i}^n \Gamma_{i,j} \right) U_i, \quad \text{pour } i = 1, \dots, n.$$

En conclusion, choisir L_i de la manière suivante permet de transformer les conditions de complémentarité en contraintes linéaires :

$$L_i \geq \sum_{j=1, j \neq i}^n \Gamma_{i,j}.$$

En rassemblant tous les résultats précédents, nous pouvons énumérer une série de conditions linéaires mixtes 0 – 1 que chaque équilibre de Nash d'un jeu polymatriciel doit satisfaire. La proposition suivante permet d'énoncer un résultat important dans le but de concevoir un algorithme d'énumération des équilibres de Nash extrêmes d'un jeu polymatriciel.

Proposition 2.4.2 *L'ensemble des équilibres de Nash d'un jeu polymatriciel est l'ensemble des vecteurs de choix stratégiques $(X_1, X_2, \dots, X_n) \in \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \times \dots \times \mathbb{R}^{m_n}$, tels que :*

$$\begin{aligned} X_i^t I_{m_i} &= 1, & \text{pour } i = 1, \dots, n. \\ \alpha_i I_{m_i} - \sum_{j=1, j \neq i}^n A_{ij} X_j &\geq 0, & \text{pour } i = 1, \dots, n. \\ X_i + U_i &\leq I_{m_i}, & \text{pour } i = 1, \dots, n. \\ \alpha_i I_{m_i} - \sum_{j=1, j \neq i}^n A_{ij} X_j - L_i U_i &\leq 0, & \text{pour } i = 1, \dots, n, \\ X_i &\geq 0, & \text{pour } i = 1, \dots, n, \\ U_i &\in \{0, 1\}^{m_i}, & \text{pour } i = 1, \dots, n. \end{aligned}$$

La preuve de cette proposition est assez facile puisqu'elle ne fait que rappeler les conditions que doit satisfaire un équilibre de Nash.

Preuve 2.4.2 *Un équilibre extrême (X, α) dans un jeu polymatriciel est défini comme*

un point tel que, pour tout joueur $i \in N$, les conditions suivantes sont satisfaites :

$$\begin{aligned} X_i + U_i &\leq \mathbf{I}_{m_i}, \\ \alpha_i \mathbf{I}_{m_i} - \sum_{j=1, j \neq i}^n A_{ij} X_j - L_i U_i &\leq 0, \\ (X_i, \alpha_i) \in (X, \alpha) &\equiv \{(X_i, \alpha_i) \in \mathbb{R}^{m_i+1} : X_i^t \mathbf{I}_{m_i} = 1, \alpha_i \mathbf{I}_{m_i} \geq \sum_{j=1, j \neq i}^n A_{ij} X_j, X_i \geq 0\}. \end{aligned}$$

Cette proposition permet donc d'énoncer un programme linéaire mixte 0 – 1 dont les solutions extrêmes sont les équilibres de Nash extrêmes d'un jeu polymatriciel.

Corollaire 2.4.1 *L'énumération des équilibres de Nash extrêmes d'un jeu polymatriciel est équivalente à l'énumération de toutes les solutions extrêmes d'un programme linéaire mixte 0 – 1, sujet aux contraintes précédentes et ayant une fonction objectif linéaire, par exemple :*

$$\min_{\alpha, X, U} f(\alpha_1, \alpha_2, \dots, \alpha_n) = \sum_{i=1}^n \alpha_i,$$

ou encore : $\min_{\alpha, X, U} f(\alpha_1, \alpha_2, \dots, \alpha_n) = 0$.

Ainsi pour chaque joueur i , un vecteur de variables binaires U_i est associé au vecteur des choix stratégiques X_i . Tous les deux ont la même dimension m_i , nombre de stratégies pures dont le joueur i dispose.

Pour un jeu polymatriciel, où N est le nombre de joueurs et $M = \sum_{i=1}^N m_i$, la formulation linéaire mixte 0 – 1 présentée ci-dessus compte, en plus des contraintes de non-négativité et d'intégralité, $(N + 3M)$ contraintes.

Exemple 2.4.1 *Soit le jeu polymatriciel à trois joueurs suivant :*

$$A = \left(\begin{array}{cc|cc} 2 & 0 & 3 & 5 \\ 3 & 4 & 1 & 4 \end{array} \right) \quad B = \left(\begin{array}{cc|cc} 0 & 5 & 3 & 2 \\ 1 & 4 & 6 & 7 \end{array} \right) \quad C = \left(\begin{array}{cc|cc} 4 & 1 & 6 & 0 \\ 1 & 2 & 5 & 3 \end{array} \right).$$

Le programme linéaire mixte 0–1 dont les solutions extrêmes sont les équilibres de Nash extrêmes de ce jeu, s'écrit selon la proposition (2.4.2) et son corollaire de la manière suivante :

$$\begin{aligned}
 & \min_{\alpha, \beta, \gamma, x, y, z, u, v, w} && f(\alpha, \beta, \gamma) \\
 & \text{s.à.} \\
 & x_1 + x_2 = 1, & y_1 + y_2 = 1, & z_1 + z_2 = 1, \\
 & x_1 + u_1 \leq 1, & y_1 + v_1 \leq 1, & z_1 + w_1 \leq 1, \\
 & x_2 + u_2 \leq 1, & y_2 + v_2 \leq 1, & z_2 + w_2 \leq 1, \\
 \\
 & \alpha - 2y_1 - 3z_1 - 5z_2 - 8u_1 \leq 0, & \alpha - 3y_1 - 4y_2 - z_1 - 4z_2 - 8u_2 \leq 0, \\
 & -\alpha + 2y_1 + 3z_1 + 5z_2 \leq 0, & -\alpha + 3y_1 + 4y_2 + z_1 + 4z_2 \leq 0, \\
 & \beta - 5x_2 - 3z_1 - 2z_2 - 10v_1 \leq 0, & \beta - x_1 - 4x_2 - 6z_1 - 7z_2 - 10v_2 \leq 0, \\
 & -\beta + 5x_2 + 3z_1 + 2z_2 \leq 0, & -\beta + x_1 + 4x_2 + 6z_1 + 7z_2 \leq 0, \\
 & \gamma - 4x_1 - x_2 - 6y_1 - 9w_1 \leq 0, & \gamma - x_1 - 2x_2 - 5y_1 - 3y_2 - 9w_2 \leq 0, \\
 & -\gamma + 4x_1 + x_2 + 6y_1 \leq 0, & -\gamma + x_1 + 2x_2 + 5y_1 + 3y_2 \leq 0, \\
 \\
 & x \geq 0, & y \geq 0, & z \geq 0, \\
 & u \in \{0, 1\}^2, & v \in \{0, 1\}^2 & w \in \{0, 1\}^2.
 \end{aligned}$$

2.5 Discussion

Dans ce chapitre, nous avons fait une synthèse des formulations mathématiques d'un jeu polymatriciel. Nous avons ensuite exposé l'algorithme des pivots complémentaires de recherche d'équilibres de Nash pour les jeux polymatriciels (70). Nous avons aussi approfondi la méthode d'élimination des stratégies dominées pour les jeux bimatriciels et polymatriciels à trois joueurs. Nous avons finalement linéarisé les conditions de complémentarité en introduisant des variables binaires. Cette nouvelle formulation a permis de définir un programme linéaire mixte 0 – 1 dont les solutions extrêmes sont des équi-

libres de Nash extrêmes du jeu polymatriciel. Nous présenterons au quatrième chapitre l'algorithme $E\chi MIP$ qui utilise ce programme linéaire mixte 0 – 1 afin d'énumérer les équilibres de Nash extrêmes d'un jeu polymatriciel.

CHAPITRE 3

LES JEUX SOUS FORME ÉTENDUE

Différentes formes ont été utilisées pour représenter les jeux. Myerson (49) décrit la forme *étendue* et la forme *normale* (stratégique) comme les formes de jeux les plus importantes.

Les spécialistes de la théorie des jeux ont pour objectif d'essayer de prévoir la manière avec laquelle un jeu se déroulera en cherchant et/ou en énumérant son ou ses équilibre(s). Nous avons étudié dans le premier et le deuxième chapitre le problème d'énumération des équilibres de Nash extrêmes pour les jeux bimatriciels et polymatriciels.

Dans un contexte où les choix stratégiques des joueurs ne sont pas tous formulés au même instant, la représentation étendue du jeu est reconnue comme étant la manière la plus complète de le décrire. La définition usitée d'un jeu étendu est due à Kuhn (36). Un tel jeu est représenté habituellement par un arbre avec des branches illustrant les mouvements possibles des joueurs.

L'objectif de ce chapitre est de confectionner une méthode permettant d'énumérer tous les équilibres de Nash extrêmes d'un jeu étendu à deux joueurs. La première section est un survol des principales formes selon lesquelles un jeu peut être présenté. Les différentes représentations d'un jeu étendu y sont aussi exposées. Ensuite nous présentons une formulation linéaire mixte 0 – 1 basée sur la forme séquentielle d'un jeu étendu à deux joueurs.

3.1 Revue de la littérature

La recherche des équilibres dans un jeu étendu a généralement été abordée à travers sa conversion en un jeu sous forme stratégique. Ainsi, chaque combinaison des mouvements successifs possibles d'un joueur est représentée par une stratégie. Afin d'éviter une croissance exponentielle de la taille du jeu suite à cette conversion, Wilson (66) et Koller et Megiddo (34) proposent des mécanismes de recherche qui font appel aux stratégies mixtes avec de petits supports. Romanovskii (56), Selten (58), Koller et Megiddo (33) et von Stengel (60) utilisent une approche séquentielle en remplaçant les stratégies pures par des séquences de mouvements.

3.1.1 Les jeux étendus

Dans un jeu étendu, les états auxquels les joueurs peuvent se retrouver sont représentés par des noeuds. Un noeud racine indique le début du jeu et chaque feuille (noeud terminal) indique la fin du jeu. À la fin du jeu, chaque joueur reçoit un gain selon les mouvements des joueurs durant le jeu. Ainsi, mis à part les feuilles terminales, tous les noeuds de l'arbre admettent des mouvements (décisions) associés aux arcs sortants. Certains noeuds sont déterminés par un facteur externe indépendant de la volonté des joueurs. Chaque séquence de mouvements possibles est représentée par un chemin joignant le noeud racine à l'un des noeuds terminaux. Durant le déroulement du jeu, le chemin représentant la séquence de mouvements des joueurs est appelé *chemin de jeu*.

Kuhn (36) a introduit un partitionnement des noeuds de décision selon l'ensemble informationnel auquel ils appartiennent. Tous les noeuds d'un même ensemble informationnel appartiennent à un même joueur et ont les mêmes mouvements possibles. Un joueur qui doit prendre une décision sait exactement à quel ensemble informationnel il est, mais ne sait pas à quel noeud il est.

Dans un jeu étendu à n joueurs, chaque noeud intermédiaire (non-terminal) possède un indice appartenant à l'ensemble $\{0, 1, 2, \dots, n\}$, où 0 est l'indice des noeuds dus à des facteurs externes. Ainsi, les noeuds contrôlés par le joueur i définissent l'ensemble des noeuds contrôlés par le joueur i . Chaque noeud possède aussi un deuxième indice indiquant l'ensemble informationnel que le joueur doit considérer si le jeu atteint ce noeud. Chaque mouvement possible à un noeud contrôlé par un joueur a un indice représentant le mouvement à effectuer.

Pour un joueur i , une stratégie est *séquentiellement rationnelle* à un état informationnel h_i , si le joueur i appliquait cette stratégie dans le cas où l'état informationnel h_i se réalisait.

Une stratégie de comportement pour un joueur i peut être alors redéfinie en prenant en compte les probabilités de réalisation des états informationnels de ce joueur. Pour l'ensemble des états informationnels de chaque joueur i , il y a lieu de définir alors un *vecteur de probabilité* $p_{i,s}$. Ce vecteur donne une distribution de probabilité sur les noeuds où le joueur i est à l'état s . Un vecteur de probabilité p peut alors être défini pour regrouper les différents vecteurs de probabilité $p_{i,s}$ de chaque joueur i à chacun de ses états informationnels s .

Définition 3.1.1 *Un équilibre séquentiel est toute paire (σ, p) telle que étant donné le vecteur de probabilité p , le choix stratégique σ est séquentiellement rationnel, pour tout joueur, à chaque état informationnel.*

Une stratégie de comportement σ est un équilibre séquentiel, si et seulement si il existe un vecteur de probabilités p tel que (σ, p) est un équilibre séquentiel.

Puisqu'il est généralement admis que tous les joueurs ont une *mémoire parfaite* dans un jeu étendu, nous allons considérer uniquement cette hypothèse. Cette condition signifie

que chaque joueur se rappelle toutes les informations dont il a eu connaissance pendant le déroulement du jeu, y compris ses propres mouvements antérieurs. L'exclusion de toute hypothèse de *mémoire imparfaite* est due au fait qu'un joueur rationnel ne devrait jamais oublier ses mouvements précédents. Cependant, il devient important de considérer l'hypothèse d'une mémoire imparfaite lorsque certains joueurs sont des équipes.

Un *sous-jeu* est défini comme un sous-arbre incluant tous les ensembles informationnels contenant un noeud du sous-arbre. Ainsi, les équilibres d'un jeu sous forme étendue peuvent être obtenus récursivement en considérant les sous-jeux. Dans cette optique, une distinction doit être faite entre les *jeux coopératifs* et les *jeux non-coopératifs*, les jeux à *information complète* et les jeux à *information incomplète* et les jeux à *information parfaite* et les jeux à *information imparfaite*.

3.1.2 Jeux coopératifs et jeux non-coopératifs

En théorie des jeux, les jeux coopératifs et les jeux non-coopératifs sont abordés de manière différente. Une distinction entre ces deux classes de jeux est fondamentale. Nash (50; ?), qui fût le premier à faire cette distinction, a défini les jeux coopératifs comme des jeux permettant la communication libre et les accords forcés entre les joueurs. Les jeux non-coopératifs n'autorisent quant à eux ni la communication ni les accords forcés entre les joueurs. Cependant, la libre communication entre les joueurs apparaît, selon Harsanyi et Selten (23), comme un critère moins important dans cette distinction.

Afin d'illustrer cette distinction, l'exemple classique du *dilemme du prisonnier* (Luce et Raiffa (39)) paraît le plus approprié (Tableau 3.1).

Tableau 3.1 Dilemme du prisonnier

| | | |
|-------|----------|----------|
| | C^{**} | N^{**} |
| C^* | 10, 10 | -10, 11 |
| N^* | 11, -10 | 1, 1 |

Ce jeu bimatriciel admet donc les matrices des gains suivantes :

$$A = \begin{pmatrix} 10 & -10 \\ 11 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 10 & 11 \\ -10 & 1 \end{pmatrix}.$$

Dans cet exemple, les deux joueurs ont les mêmes pondérations de gain selon les stratégies qu'ils peuvent choisir puisque le jeu est symétrique. Ainsi, il est normal de s'attendre à ce que les joueurs se mettent d'accord sur une solution qui leur donnerait des gains égaux. Deux solutions sont alors possibles, $C = (C^*, C^{**})$ qui donnerait les gains (10, 10), et $N = (N^*, N^{**})$ qui donnerait les gains (1, 1). Si le jeu est coopératif les joueurs n'auront aucune peine à se mettre d'accord pour choisir la paire de stratégies C qui leur procurerait des gains supérieurs. Ainsi, $C = (C^*, C^{**})$ peut être vue comme la *solution coopérative* au jeu. D'autre part, si le jeu est non-coopératif, les joueurs ne pourront faire mieux que de choisir la paire $N = (N^*, N^{**})$ qui sera vue comme étant la *solution non-coopérative* au jeu.

Ce dernier résultat est dû au fait que même si les joueurs arrivaient à conclure un accord, aucun d'eux ne serait obligé de le respecter. Ainsi, si le premier joueur pense que le deuxième joueur va choisir sa stratégie C^{**} , il aura intérêt à choisir sa stratégie N^* , puisque celle-ci lui procurerait le meilleur gain. De même, si le deuxième joueur pense que le premier joueur va choisir sa stratégie C^* , il aura intérêt à choisir sa stratégie N^{**} , vu que celle-ci lui procurerait le meilleur gain. La paire C est donc déstabilisante pour elle-même, puisque N^* est la meilleure réplique à C^{**} et N^{**} est la meilleure réplique à C^* . Par contre, la paire N est stabilisante pour elle-même.

Plusieurs points importants se dégagent à partir de l'exemple précédent. Pour les jeux non-coopératifs, Nash (50; ?) a relevé ces mêmes points. Comme il est impossible d'avoir des accords forcés dans un jeu non-coopératif, des joueurs rationnels choisiraient une combinaison stratégique stabilisante pour elle-même. Ainsi, tout joueur choisirait la meilleure réplique aux choix stratégiques des autres. Une combinaison stratégique définie de la manière précédente n'est autre qu'un *équilibre de Nash*. En outre, Nash (50) a montré que tout jeu fini possède au moins un équilibre, même parfois uniquement en *stratégies mixtes*.

3.1.3 Jeux à information complète et jeux à information incomplète

Un jeu est à information complète si tous les joueurs savent comment le jeu peut se dérouler. Les joueurs auraient alors toute l'information sur la forme étendue (l'arbre) ou la forme normale (stratégique) du jeu (la matrice des gains). Un jeu à information complète peut être avec information parfaite ou avec information imparfaite.

Par contre, dans un jeu à information incomplète, les joueurs n'ont pas une information complète sur les stratégies possibles, ni sur les conséquences produites par les combinaisons diverses de ces stratégies, ni sur les préférences des autres joueurs. En plus, chaque joueur peut ne pas avoir d'idée sur les informations dont disposent les autres joueurs quant aux stratégies et aux préférences des autres joueurs.

La théorie des jeux classique a du mal à traiter les jeux avec information incomplète. Cependant, l'utilisation de modèles probabilistes permet de représenter l'information incomplète que les joueurs ont sur les divers paramètres du jeu. Ainsi, un jeu avec information incomplète peut se réduire à un jeu contenant des mouvements avec certaines probabilités sur les gains que ces mouvements pourraient rapporter.

3.1.4 Jeux à information parfaite et jeux à information imparfaite

Dans un jeu à information parfaite, les joueurs ont une idée sur le déroulement du jeu, sa nature et sur tous les mouvements antérieurs à chaque étape du jeu. En d'autres termes, chaque ensemble informationnel est un singleton et chaque noeud est la racine d'un sous-jeu. Dans un contexte de jeu à information imparfaite, les joueurs ont une idée sur le déroulement du jeu mais n'ont qu'une information amoindrie quant aux mouvements antérieurs durant le déroulement jeu.

3.2 Les représentations d'un jeu étendu

En vue d'obtenir le ou les équilibres d'un jeu étendu, trois représentations possibles sont généralement utilisées ; *la représentation sous forme stratégique, la représentation sous forme séquentielle et la représentation multiagent.*

3.2.1 Représentation sous forme stratégique

Pour chaque joueur i , l'ensemble de ses ensembles informationnels est noté H_i . Les ensembles informationnels sont notés par h_i et l'ensemble des mouvements possibles à h est noté C_{h_i} . Les équilibres de Nash d'un jeu étendu sont généralement définis comme étant les équilibres de sa représentation sous forme stratégique. Ceci est vrai en particulier pour les jeux ne contenant pas de sous-jeux. Toutefois, il y a lieu de prendre ce résultat avec beaucoup de précautions. Ainsi, nous distinguerons ultérieurement entre les différents types d'équilibres lorsque le problème du raffinement d'équilibres sera traité.

Dans un jeu étendu, une *stratégie pure* d'un joueur i est définie comme étant un mouvement déterministe à chaque ensemble informationnel. Une stratégie pure est donc un

élément de $\prod_{h_i \in H_i} C_{h_i}$. En plus, les mouvements à des ensembles informationnels qui ne peuvent être atteints à cause d'un mouvement antérieur propre au joueur sont identifiés dans la forme stratégique *réduite* du jeu. Cependant, la forme stratégique réduite est généralement de taille exponentielle par rapport à la taille du jeu original, ce qui a limité l'utilisation des jeux étendus.

Un joueur pourrait aussi avoir des ensembles informationnels parallèles qui ne peuvent être distingués par les mouvements antérieurs propres à ce joueur. Ceci arrive lorsque par exemple un joueur reçoit d'un autre joueur de l'information sur un mouvement antérieur.

En appliquant l'algorithme de Lemke-Howson à la représentation stratégique d'un jeu sous forme étendue, Wilson (66) a montré que le nombre de stratégies pures peut augmenter lors des itérations. Koller et Megiddo (34) ont cependant surmonté ce problème en introduisant un système d'équations linéaires pour les pondérations de réalisabilité des feuilles de l'arbre du jeu. En outre, alors que l'algorithme de Wilson (66) requiert une mémoire parfaite de la part des joueurs, Koller et Megiddo (34) proposent une méthode permettant l'énumération de tous les petits supports, qui peut être étendue aux jeux étendus sans mémoire parfaite.

La Figure 3.1 illustre un jeu étendu à deux joueurs pris de von Stengel (62). Le joueur 1 joue deux fois tandis que le joueur 2 ne joue qu'une fois. Pour son premier mouvement, le joueur 1 a deux alternatives L et R . Si le joueur 1 choisit la décision L à son premier mouvement, le joueur 2 aura deux alternatives l ou r . À ce stade le joueur 2 peut décider de ne pas jouer et chaque joueur recevrait alors 0 comme gain final. Si le joueur 2 choisit la décision l ou r , le joueur 1 aurait deux autres alternatives S et T . Si le joueur 1 choisit la décision R à son premier mouvement, le jeu se terminerait et le joueur 1 recevrait 3 comme gain final alors que le joueur 2 recevrait 4 comme gain final.

Les stratégies pures du joueur 1 sont les combinaisons de mouvements $\langle L, S \rangle$, $\langle L, T \rangle$,

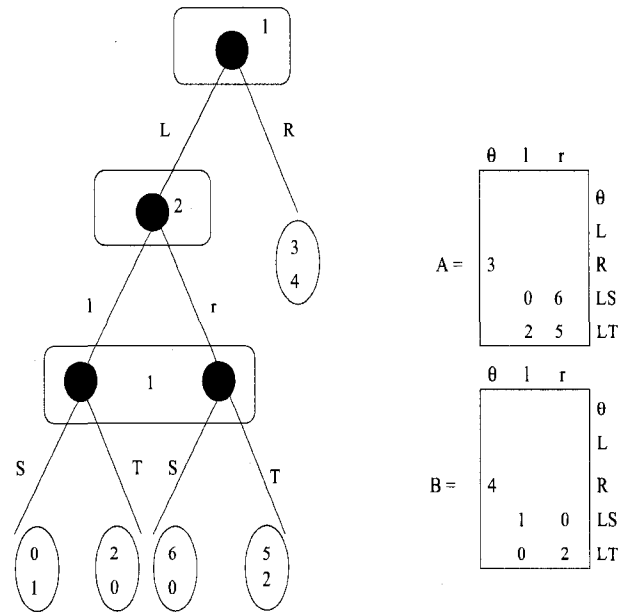


Figure 3.1 Jeu sous forme étendue à deux joueurs de von Stengel (62)

$\langle R, S \rangle$ et $\langle R, T \rangle$. Cependant, il est possible d'identifier certaines combinaisons de stratégies pures qui ne peuvent être considérées dans ce jeu. Ainsi, la forme stratégique réduite de ce jeu est :

$$A = \begin{pmatrix} 3 & 3 \\ 0 & 6 \\ 2 & 5 \end{pmatrix} \quad B = \begin{pmatrix} 4 & 4 \\ 1 & 0 \\ 0 & 2 \end{pmatrix}$$

En appliquant l'algorithme *EEE* (6) à cette forme stratégique réduite, trois équilibres de Nash extrêmes sont énumérés :

$$E_1 : \left[\begin{array}{cc} \alpha_1 = 3 & \alpha_2 = 4 \\ x_1 = [1, 0, 0] & x_2 = [\frac{2}{3}, \frac{1}{3}] \end{array} \right] \quad E_2 : \left[\begin{array}{cc} \alpha_1 = 3 & \alpha_2 = 4 \\ x_1 = [1, 0, 0] & x_2 = [1, 0] \end{array} \right]$$

$$E_3 : \left[\begin{array}{cc} \alpha_1 = 4 & \alpha_2 = \frac{2}{3} \\ x_1 = [0, \frac{2}{3}, \frac{1}{3}] & x_2 = [\frac{1}{3}, \frac{2}{3}] \end{array} \right]$$

avec α_1 et α_2 les gains respectifs des deux joueurs et x_1 et x_2 leurs vecteurs de choix de probabilité respectifs.

Ce jeu possède deux équilibres extrêmes E_1 et E_2 qui ne sont pas *sous-jeux parfaits*. Ainsi, leurs restrictions au sous-jeu commençant au noeud du joueur 2 ne constituent aucun équilibre à ces sous-jeux, telle est la définition d'un *sous-jeu parfait*. En effet, les restrictions respectives de ces deux équilibres au sous-jeu commençant au noeud du joueur 2 confèrent un gain nul aux deux joueurs.

En outre, l'élimination de toutes les stratégies redondantes permet d'obtenir une représentation stratégique totalement réduite du jeu sous forme étendue. Par ailleurs, l'élimination de toutes les stratégies fortement dominées permet d'obtenir un jeu sous forme stratégique résiduel plus facile à manipuler dans un objectif d'énumération de tous ses équilibres extrêmes, comme montré dans le deuxième chapitre.

3.2.2 Représentation sous forme séquentielle

L'utilisation des *séquences de mouvements* en lieu et place des stratégies pures est aussi possible dans le but de déterminer les différents équilibres d'un jeu sous forme étendue. Comme un jeu sous forme étendue est représenté par un arbre, il existe un chemin unique liant le noeud racine à tout autre noeud d'un tel arbre. Ce chemin représente une séquence de mouvements pour un joueur i . En supposant que chaque joueur i a une mémoire parfaite, il résulte que chaque paire de noeuds d'un ensemble informationnel h de H_i correspond à la même séquence pour le joueur i .

À un ensemble informationnel h_i , une séquence de mouvements est généralement notée

σ_{h_i} et l'ensemble des séquences de mouvements est noté S_i , pour chaque joueur i . Toute séquence de mouvements $\sigma_{h_i} \in S_i$ peut être égale à la séquence vide \emptyset ou bien uniquement obtenue par le dernier mouvement c_i du joueur i à l'ensemble informationnel $h'_i \in H_i$. Cela signifie que $\sigma_{h_i} = \sigma_{h'_i} c_i$.

La définition suivante peut alors être présentée :

$$S_i = \{\emptyset\} \cup \{\sigma_{h'_i} c_i \mid h'_i \in H_i, c_i \in C_{h_i}\}.$$

Dans l'exemple illustré par la Figure 3.1, les deux matrices des gains A et B donnent les différents gains de chaque joueur tout en considérant son ensemble de séquences. Les ensembles de séquences sont donc $S_1 = \{\emptyset, L, R, LS, LT\}$ et $S_2 = \{\emptyset, l, r\}$.

Ainsi, chaque joueur aura un nombre de séquences de jeux possibles qui n'excèdera pas le nombre de noeuds dans l'arbre. La forme séquentielle d'un jeu sous forme étendue est similaire à sa représentation sous forme stratégique réduite. L'unique distinction entre ces deux conversions vient du fait que la forme séquentielle utilise des séquences de mouvements au lieu des stratégies pures, ce qui permet d'obtenir une représentation plus compacte du jeu original.

3.2.3 Représentation multi-agents

En plus de ces deux approches, Myerson (49) propose une *représentation multi-agents* assez intéressante. Cette représentation d'un jeu étendu est toutefois largement inspirée de la définition de Selten (57). Une représentation multi-agents d'un jeu étendu peut être simplement vue comme une représentation où un agent différent est associé à chaque ensemble informationnel possible d'un joueur du jeu original, tout en sachant que ces agents partagent les mêmes préférences que le joueur qu'ils remplacent.

Myerson (49) présente par ailleurs une *stratégie de comportement* comme une distribution de probabilité sur les stratégies de la représentation multi-agents d'un jeu étendu.

Pour un joueur i , une stratégie de comportement β_i est obtenue par les probabilités $\beta(c_i)$ de son mouvement c_i telles que $\beta(c_i) \geq 0$ et $\sum_{c_i \in C_{h_i}} \beta(c_i) = 1$ pour tout $h_i \in H_i$. À travers la formulation suivante, la définition d'une stratégie de comportement peut simplement être étendue aux séquences $\sigma_{h_i} \in S_i$:

$$\beta[c_i] = \prod_{c_i \in \sigma_{h_i}} \beta(c_i).$$

Dans ce contexte, une stratégie pure π pour un joueur donné est un genre de stratégie de comportement avec $\pi(c_i) \in \{0, 1\}$ pour tous les mouvements c_i . Ceci veut dire que $\pi[\sigma_{h_i}] \in \{0, 1\}$ pour tout $\sigma \in S_i$. Une stratégie mixte μ correspond donc à une probabilité $\mu(\pi)$ à toute stratégie pure π du joueur i . Les *probabilités de réalisation* de jouer les séquences $\sigma_{h_i} \in S_i$ sont alors définies comme suit

$$\mu[\sigma_{h_i}] = \sum_{\pi} \mu(\pi) \pi[\sigma_{h_i}].$$

Pour un joueur i , un plan de réalisation de μ est alors dénoté $x(\sigma_{h_i}) = \mu[\sigma_{h_i}]$ pour $\sigma_{h_i} \in S_i$. Koller et Megiddo (33) montrent que pour un joueur i donné, x est le plan de réalisation d'une stratégie mixte si et seulement si les conditions suivantes sont satisfaites :

$$x(\emptyset) = 1,$$

$$\sum_{c_i \in C_{h_i}} x(\sigma_{h_i} c_i) = x(\sigma_{h_i}), \quad h_i \in H_i,$$

$$x(\sigma_{h_i}) \geq 0, \quad \forall \sigma_{h_i} \in S_i.$$

3.3 Formulation linéaire mixte 0 – 1

Au premier chapitre, nous avons proposé une formulation linéaire mixte 0 – 1 d'un jeu bimatriciel. La représentation stratégique d'un jeu étendu à deux joueurs pourrait aussi bénéficier du même type de formulation. En effet, la représentation stratégique d'un jeu étendu à deux joueurs est en fin de compte un jeu bimatriciel.

Nous proposons dans cette section une formulation linéaire mixte 0 – 1 de la forme séquentielle d'un jeu étendu à deux joueurs.

Comme énoncé précédemment, toute séquence de mouvements $\sigma_{h_i} \in S_i$ peut être égale à la séquence vide \emptyset ou bien uniquement obtenue par le dernier mouvement c_i du joueur i à l'ensemble informationnel $h_i \in H_i$. Ainsi, S_i peut être présenté de la manière suivante :

$$S_i = \{\emptyset\} \cup \{\sigma_{h'_i} c_i \mid h'_i \in H_i, c_i \in C_h\}.$$

En dénotant $x_i = (x_\sigma)_{\sigma \in S_i}$, le vecteur des choix des stratégies du joueur i , ces conditions peuvent être reformulées pour chaque joueur i comme suit :

$$x_i \geq 0, \quad E_i x_i = e_i,$$

où E_i est une matrice adéquatement choisie et $e_i = (1, 0, \dots, 0)^t$, tous deux ayant $1 + |H_i|$ lignes.

Dans l'exemple illustré par la Figure 3.1, les deux matrices des gains A et B donnent les différents gains de chaque joueur considérant son ensemble de séquences. Ainsi, les ensembles de séquences sont $S_1 = \{\emptyset, L, R, LS, LT\}$ et $S_2 = \{\emptyset, l, r\}$. Les matrices E_1 , E_2 , e_1 et e_2 s'écrivent donc ainsi :

$$E_1 = \begin{bmatrix} 1 & & & \\ -1 & 1 & 1 & \\ & -1 & & 1 & 1 \end{bmatrix}, \quad e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad E_2 = \begin{bmatrix} 1 & & & \\ -1 & 1 & 1 & \end{bmatrix}, \quad e_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

Les matrices E_1 et E_2 décrivent respectivement les liens entre les différentes séquences de jeu du joueur 1 et 2.

Koller et Megiddo (33) montrent que deux stratégies mixtes μ et μ' d'un joueur i sont équivalentes du point de vue plan de réalisation si $\mu[\sigma] = \mu'[\sigma]$ pour tout $\sigma \in S_i$. En plus, Kuhn (36) montre que pour un joueur avec une mémoire parfaite, toute stratégie mixte est équivalente du point de vue plan de réalisation à une stratégie de comportement.

Romanovskii (56) et von Stengel (60) démontrent que les équilibres d'un jeu étendu à deux joueurs, à somme non nulle et avec mémoire parfaite, sont les solutions simultanées aux programmes linéaires suivants :

$$\begin{array}{ll} \max_{x_1} x_1^t A x_2 & \max_{x_2} x_1^t B x_2 \\ \text{s.à. } E_1 x_1 = e_1, & \text{et} \quad \text{s.à. } E_2 x_2 = e_2, \\ x_1 \geq 0, & x_2 \geq 0. \end{array} \quad (3.3.1)$$

Les formulations duales à ces programmes linéaires s'expriment alors comme suit :

$$\begin{array}{ll} \min_{\alpha_1} e_1^t \alpha_1 & \min_{\alpha_2} e_2^t \alpha_2 \\ \text{s.à. } E_1^t \alpha_1 \geq A x_2, & \text{et} \quad \text{s.à. } \alpha_2^t E_2 \geq x_1^t B. \end{array}$$

Les conditions de complémentarité obtenues sont :

$$x_1^t(E_1^t\alpha_1 - Ax_2) = 0 \quad \text{et} \quad (\alpha_2^tE_2 - x_1^tB)x_2 = 0. \quad (3.3.2)$$

Júdice et Mitra (29), puis Audet *et al.* (8), ont mis en place une manière efficace de transformer les conditions des écarts complémentaires en introduisant des variables binaires. Ainsi, en introduisant deux vecteurs de variables binaires u_1 et u_2 , ces conditions de complémentarité (3.3.2) peuvent être linéarisées de la manière suivante :

$$x_1^t(E_1^t\alpha_1 - Ax_2) = 0 \Leftrightarrow \begin{cases} E_1^t\alpha_1 - Ax_2 \leq L_1u_1, \\ x_1 + u_1 \leq \mathbf{1}_n, \end{cases}$$

et

$$(\alpha_2^tE_2 - x_1^tB)x_2 = 0 \Leftrightarrow \begin{cases} \alpha_2^tE_2 - x_1^tB \leq L_2u_2, \\ x_2 + u_2 \leq \mathbf{1}_m, \end{cases}$$

où $\mathbf{1}_n$ et $\mathbf{1}_m$ sont des vecteurs ayant tous leurs éléments égaux à 1, et L_1 et L_2 sont des scalaires adéquatement choisis comme au premier chapitre. Les paramètres L_1 et L_2 peuvent prendre une valeur égale à la différence entre le plus grand et le plus petit élément des matrices A et B , respectivement.

L'énumération complète de tous les équilibres de Nash extrêmes de la représentation séquentielle d'un jeu étendu à deux joueurs est alors équivalente à l'énumération de tous les points extrêmes de l'ensemble Q des solutions d'un programme linéaire mixte 0 – 1

sujet aux conditions suivantes :

$$\begin{aligned}
E_1 x_1 &= e_1, & E_2 x_2 &= e_2, \\
E_1^t \alpha_1 &\geq A x_2, & \alpha_2^t E_2 &\geq x_1^t B, \\
E_1^t \alpha_1 - A x_2 &\leq L_1 u_1, & \alpha_2^t E_2 - x_1^t B &\leq L_1 u_2, \\
x_1 + u_1 &\leq \mathbf{1}_n, & x_2 + u_2 &\leq \mathbf{1}_m, \\
x_1 &\geq 0, & x_2 &\geq 0, \\
u_1 &\in \{0, 1\}^n, & u_2 &\in \{0, 1\}^m.
\end{aligned} \tag{3.3.3}$$

Ainsi, si n et m sont les nombres des pures séquences de jeu des joueurs I et II respectivement, chaque combinaison de vecteurs de variables binaires u_1 et u_2 définit un polytope. L'ensemble Q est alors l'ensemble de ces polytopes disjoints.

Théorème 3.3.1 *Soit un jeu étendu à deux joueurs et soit NE son ensemble d'équilibres de Nash, $NE = \{(x_1, x_2) : (x_1, x_2) \text{ est un équilibre}\}$ et $Ext(NE)$ son ensemble d'équilibres extrêmes. Soit Q l'ensemble des solutions aux conditions (3.3.3) et $Ext(Q)$ l'ensemble des points extrêmes des polytopes de Q . Nous avons alors :*

$$NE = Proj_x(Q) \text{ et } Ext(NE) \subseteq Proj_x(Ext(Q)).$$

Ainsi, la projection de Q sur l'espace des x engendre NE . Chaque élément de NE correspond à au moins un élément de Q et chaque élément de $Ext(NE)$ peut être obtenu par la projection d'un élément de $Ext(Q)$.

Preuve 3.3.1 *Romanovskii (56) et von Stengel (60) montrent que chaque élément $X = (x_1, x_2)$ de NE est tel que x_1 et x_2 sont simultanément les solutions optimales de la paire de programmes (3.3.1). Nous avons alors introduit les variables duales α et β et les variables binaires u_1 et u_2 dans le but de satisfaire les conditions de complémentarité (3.3.2).*

Chaque élément de NE est donc la projection sur l'espace des x d'au moins un élément de Q , d'où $NE \subseteq Proj_x(Q)$. Ceci implique que la projection de tout élément de Q sur l'espace des x est un équilibre de la représentation séquentielle : $Proj_x(Q) \subseteq NE$. Ainsi : $Proj_x(Q) = NE$. Notons par ailleurs que $Ext(NE) \subseteq NE$ implique que $Ext(NE) \subseteq Proj_x(Q)$.

En plus, tout élément de $Ext(NE)$ est un équilibre de Nash extrême qui ne peut s'exprimer par une combinaison convexe d'autres équilibres. Supposons alors qu'il existe un élément $X \in Ext(NE)$ qui soit la projection de $q = (X, \alpha, u) \in Q$, tel que $q \notin Ext(Q)$. Alors q peut s'exprimer comme combinaison convexe d'au moins deux éléments de $Ext(Q)$: $q = \sum_i \lambda^i q^i$, tel que $\forall i : q^i = (X^i, \alpha^i, u^i)$, $q^i \in Ext(Q)$, $\lambda^i > 0$ et $\sum \lambda^i = 1$.

Alors, $q = (\sum_i \lambda^i X^i, \sum_i \lambda^i \alpha^i, \sum_i \lambda^i u^i) = (X, \alpha, u)$. Ce qui veut dire que $X = \sum_i \lambda^i X^i$, avec $\forall i : X^i = (x_1^i, x_2^i) \in NE$. Sachant que $X \in Ext(NE)$, nous avons alors $X^i = X \forall i$. Nous pouvons donc conclure que tout élément $X \in Ext(NE)$ est aussi la projection d'au moins un élément de $Ext(Q)$. Ainsi $Ext(NE) \subseteq Proj_x(Ext(Q))$.

Pour l'exemple de la Figure 3.1, le programme linéaire mixte 0–1 satisfaisant les conditions (3.3.3) est le suivant :

$$\begin{aligned}
& \min_{\alpha, \beta, x, y, u, v} f(\alpha_1, \beta_1) \\
& \text{s.à.} \\
& x_1 = 1, \quad y_1 = 1, \\
& -x_1 + x_2 + x_3 = 0, \quad -y_1 + y_2 + y_3 = 0, \\
& -x_2 + x_4 + x_5 = 0, \\
& -\alpha_1 + \alpha_2 \leq 0, \quad -\beta_1 + \beta_2 + 4x_3 \leq 0, \\
& -\alpha_2 + \alpha_3 \leq 0, \quad -\beta_2 + x_4 \leq 0, \\
& -\alpha_2 + 3y_1 \leq 0, \quad -\beta_2 + 2x_5 \leq 0, \\
& -\alpha_3 + 6y_3 \leq 0, \\
& -\alpha_3 + 2y_2 + 5y_3 \leq 0, \\
& -\alpha_2 - 5u_1 \leq 0, \quad \beta_1 - \beta_2 - 4x_3 - 4v_1 \leq 0, \\
& \alpha_2 - \alpha_3 - 5u_2 \leq 0, \quad \beta_2 - x_4 - 4v_2 \leq 0, \\
& \alpha_2 - 3y_1 - 5u_3 \leq 0, \quad \beta_2 - 2x_5 - 4v_3 \leq 0, \\
& \alpha_3 - 6y_3 - 5u_4 \leq 0, \\
& \alpha_3 - 2y_2 - 5y_3 - 5u_5 \leq 0, \\
& x_1 + u_1 \leq 1, \quad y_1 + v_1 \leq 1, \\
& x_2 + u_2 \leq 1, \quad y_2 + v_2 \leq 1, \\
& x_3 + u_3 \leq 1, \quad y_3 + v_3 \leq 1, \\
& x_4 + u_4 \leq 1, \\
& x_5 + u_5 \leq 1, \\
& x \geq 0, \quad y \geq 0, \\
& u_1 \in \{0, 1\}^5, \quad u_2 \in \{0, 1\}^3.
\end{aligned}$$

Pour le choix de l'objectif, n'importe quelle fonction linéaire pourrait être retenue. Les algorithmes présentés dans le quatrième chapitre ont parmi leurs objectifs d'énumérer tous les équilibres extrêmes de la représentation séquentielle d'un jeu étendu à deux joueurs.

3.4 Discussion

Nous avons présenté dans ce chapitre une revue des différents concepts utilisés pour les jeux étendus. Nous avons ensuite étudié les différentes représentations d'un jeu étendu à deux joueurs. Nous avons proposé finalement un programme linéaire mixte 0 – 1 dont les solutions extrêmes permettent, par projection sur l'espace des variables des stratégies mixtes, d'énumérer les équilibres de Nash extrêmes de la représentation séquentielle d'un jeu étendu à deux joueurs. Nous présenterons au chapitre suivant l'algorithme *E χ MIP* qui utilise ce programme linéaire mixte 0 – 1 afin d'énumérer les équilibres de Nash extrêmes de la forme étendue d'un jeu à deux joueurs.

CHAPITRE 4

ALGORITHMES DE RÉOLUTION

Ce chapitre présente les deux algorithmes $E\chi MIP$ et EEE , permettant l'énumération complète des équilibres de Nash extrêmes d'un jeu bimatriciel et de la représentation séquentielle d'un jeu étendu à deux joueurs. L'algorithme $E\chi MIP$ permet aussi d'énumérer les équilibres de Nash extrêmes d'un jeu polymatriciel à trois joueurs. Les deux algorithmes sont implantés en C++, en utilisant le logiciel d'optimisation *Cplex* pour une résolution en arithmétique réelle et une librairie de très grands entiers afin d'obtenir une résolution en arithmétique exacte. Nous décrivons aussi le logiciel *XGame Solver* que nous avons élaboré au cours de ce projet de thèse de doctorat et dont nous comptons faire bénéficier la communauté scientifique.

4.1 Présentation de l'algorithme $E\chi MIP$

Pour obtenir des temps de calculs réduits, il est primordial de concevoir un algorithme doté d'une aptitude à détecter rapidement toute information susceptible d'accélérer l'exploration.

Tout d'abord, l'algorithme que nous allons concevoir doit exploiter l'information relative à la non-réalisabilité d'un des sous-problèmes qu'il aura généré. Pour satisfaire cette exigence, nous proposons un algorithme qui explore un arbre de branchement où à chaque niveau de profondeur s'offre un choix dichotomique de branchement sur une seule des variables binaires.

Par ailleurs, l'algorithme doit permettre d'exploiter toute l'information fournie par toute

solution (*équilibre de Nash extrême*) qu'il rencontre. Ceci a pour but de faciliter l'exploration du domaine des solutions possibles qu'un vecteur de variables binaires peut fournir. Il faudrait dès lors que l'algorithme puisse générer, à partir de chaque noeud de l'arbre principal, un sous-arbre de branchement par rapport aux variables continues et strictement positives dans la solution extrême obtenue par la résolution du sous-programme courant. Ce branchement intensif est indispensable pour l'énumération complète des équilibres extrêmes, puisqu'une seule combinaison de variables binaires peut donner plusieurs points extrêmes.

L'algorithme d'énumération des équilibres extrêmes va garder en mémoire tous les points extrêmes qu'il rencontre lors de son exploration de l'arbre principal et de ses recherches intensives dans les sous-arbres. Cependant, un point d'équilibre extrême pourrait être rencontré à plusieurs reprises lors de l'exécution de l'algorithme, car deux ou plusieurs combinaisons différentes de variables binaires peuvent donner lieu à un ou plusieurs mêmes points extrêmes. L'algorithme doit donc être capable de vérifier si la solution courante (*équilibre extrême*) qu'il rencontre a déjà été rencontrée lors de son exploration de l'arbre principal ou des sous-arbres. Il est donc possible qu'il n'enregistre qu'une partie des équilibres engendrés par une combinaison de variables binaires.

À chaque fois que l'algorithme ajoutera une branche dans l'arbre principal ou dans les sous-arbres, il prendra le noeud nouvellement créé comme noeud courant pour traiter son(ses) sous-programme(s). Ainsi, l'ordre de traitement des noeuds générés dans l'arbre principal et les arbres secondaires est toujours "*profondeur d'abord*".

Chaque noeud de l'arbre principal représente un sous-programme linéaire mixte 0 – 1 créé par l'ajout d'une ou plusieurs contraintes de fixation de variables binaires et d'élimination d'une ou plusieurs combinaisons de variables binaires. Au même temps, chaque noeud d'un arbre secondaire représente un sous-programme linéaire créé à la fois par la fixation d'une combinaison de variables binaires et d'une ou d'un ensemble de variables

continues égales à 0.

Enfin, à chaque noeud de l'arbre principal, ou des sous-arbres qu'il aura générés, l'algorithme devra optimiser respectivement dans l'un ou l'autre cas un programme linéaire mixte 0 – 1 ou relaxé, en faisant appel à une librairie d'optimisation.

4.2 Algorithme $E\chi MIP$

L'algorithme $E\chi MIP$ exploite la formulation linéaire mixte 0 – 1 d'un jeu bimatriciel ou polymatriciel ou de la représentation séquentielle d'un jeu étendu à deux joueurs, afin d'énumérer tous leurs équilibres de Nash extrêmes.

Étape 0 : Initialisation

Soient :

P ; Problème linéaire mixte 0 – 1 principal.

X ; Ensemble des variables continues associé au problème P .

U ; Ensemble des variables binaires associé au problème P .

$NE = \emptyset$; Ensemble des points extrêmes.

$N = 0$; Niveau de profondeur dans l'arbre principal.

R ; Noeud racine de l'arbre principal, ayant P comme problème.

x_i^q, u_i^q ; Variable continue et variable binaire associée à la $q^{\text{ème}}$ stratégie ($q = 1, 2, \dots, m_i$) du joueur i ($i = 1, 2, \dots, n$).

→ *Aller à l'Étape 1.*

Étape 1 : Résolution et mémorisation

Si $N > |X|$, → *Arrêt.* Sinon $N \leq |X|$, résoudre le problème au noeud courant. Si le problème n'admet aucune solution → *Aller à l'Étape 3.*

Sinon, soit \hat{e} le point extrême trouvé : Si $\hat{e} \notin NE$, ajouter \hat{e} aux éléments de NE et → *Aller à l'Étape 2.*

Étape 2 : Branchement secondaire

Si le noeud courant appartient à l'arbre principal, fixer le vecteur des variables binaires \hat{u} ($\hat{u} \in U$) à sa valeur dans \hat{e} et $\forall x_i^q \in X$, tel que $\hat{x}_i^q > 0$:

→ Ajouter la branche $x_i^q = 0$ et *Aller à l'Étape 1.*

Étape 3 : Branchement principal

Si le noeud courant appartient à l'arbre principal alors le branchement à partir de ce noeud ne peut donner des points extrêmes → *Arrêt.*

Sinon, revenir au noeud père dans l'arbre principal et ajouter une contrainte pour éliminer la combinaison de variables binaires \hat{u} valide dans \hat{e} .

Choisir une variable $u_i^q \in U$, sur laquelle aucun branchement n'a encore été effectué dans les noeuds précédents de l'arbre principal et telle que sa variable continue x_i^q est la plus proche de $\frac{1}{2}$ (choisir arbitrairement en cas d'égalité).

Soit $p = N + 1$:

Si $p \leq |X|$ mettre $N = p$ et :

– Ajouter la branche $u_i^q = 0$: si, dans \hat{u} , $\hat{u}_i^q = 1$: enlever la contrainte d'élimination de \hat{u} et → *Aller à l'Étape 1.*

– Ajouter la branche $u_i^q = 1$: si, dans \hat{u} , $\hat{u}_i^q = 0$: enlever la contrainte d'élimination de \hat{u} et → *Aller à l'Étape 1.*

Sinon → *Aller à l'Étape 4.*

Étape 4 : Fin

Afficher $|NE|$ et tous les éléments de NE .

Théorème 4.2.1 *L'algorithme $E_{\chi}MIP$ permet d'énumérer tous les équilibres extrêmes d'un jeu bimatriciel ou polymatriciel ou de la forme séquentielle d'un jeu étendu à deux joueurs.*

Preuve 4.2.1 *$E_{\chi}MIP$ explore à travers le branchement sur l'arbre principal, toutes les combinaisons de variables binaires qui permettent d'obtenir des points extrêmes et ce par :*

- L'ajout de contraintes d'élimination pour les combinaisons déjà explorées.
- Le branchement sur les variables binaires jusqu'à une profondeur maximale égale au nombre de stratégies associées au jeu bimatriciel ou polymatriciel.
- Grâce au branchement sur l'arbre secondaire, l'algorithme énumère tous les points extrêmes qui peuvent être obtenus à partir d'une combinaison de variables binaires \hat{u} car :
 - La combinaison de variables binaires \hat{u} est fixée.
 - Le branchement secondaire consiste à ajouter des branches $x_i^q = 0$.

Ce branchement permet d'énumérer à partir de \hat{u} tous les équilibres extrêmes où une ou plusieurs conditions de complémentarité sont doublement satisfaites vu que dans \hat{e} : $x_i^q > 0$ et $u_i^q = 0$. Ce qui permet d'obtenir après branchement :

$$\left\{ \begin{array}{l} x_i^q = 0 \quad \text{et} \quad u_i^q = 0, \\ (\alpha_i - \sum_{j=1, j \neq i}^n A_{ij}^q X_j) = 0, \end{array} \right. \implies x_i^q (\alpha_i - \sum_{j=1, j \neq i}^n A_{ij}^q X_j) = 0.$$

Nous avons montré dans les chapitres 1, 2 et 3 que tout équilibre de Nash extrême correspond forcément à un ou plusieurs points extrêmes du programme linéaire mixte 0 – 1 principal. Puisque l'algorithme $E\chi MIP$ permet d'explorer toutes les possibilités de satisfaire les conditions de complémentarité, alors pour tout point extrême \hat{e} du programme linéaire mixte 0 – 1 principal, il existe nécessairement un chemin dans l'arbre d'exploration créé par $E\chi MIP$ qui mène à ce point extrême.

Notons ici que l'algorithme $E\chi MIP$ est actuellement le seul algorithme permettant d'énumérer tous les équilibres de Nash extrêmes d'un jeu polymatriciel.

4.3 Exemples illustrés

4.3.1 Jeux bimatriciels

Exemple 4.3.1 *Pour le jeu bimatriciel de taille (4×3) du premier chapitre, cinq équilibres de Nash extrêmes sont énumérés. L'élimination d'une stratégie fortement dominée, la 2^{ème} stratégie du joueur I, avait permis d'obtenir un jeu de taille (3×3) :*

$$A' = \begin{pmatrix} 3 & 2.5 & 5 \\ 2 & 3.5 & 1.5 \\ 4.5 & 0.5 & 5.5 \end{pmatrix} \quad B' = \begin{pmatrix} 5 & 4.5 & 2.5 \\ 1 & 1.5 & 3.5 \\ 2.5 & 3.5 & 4 \end{pmatrix}$$

La Figure 4.1 illustre une grande partie des 46 noeuds générés par l'algorithme lors de l'énumération des équilibres de Nash extrêmes. Les lignes pleines représentent les branchements de l'algorithme et les lignes pointillées représentent les fixations des combinaisons de variables binaires avant le début des branchements secondaires.

Le programme linéaire mixte 0 – 1 obtenu pour ce jeu est le suivant :

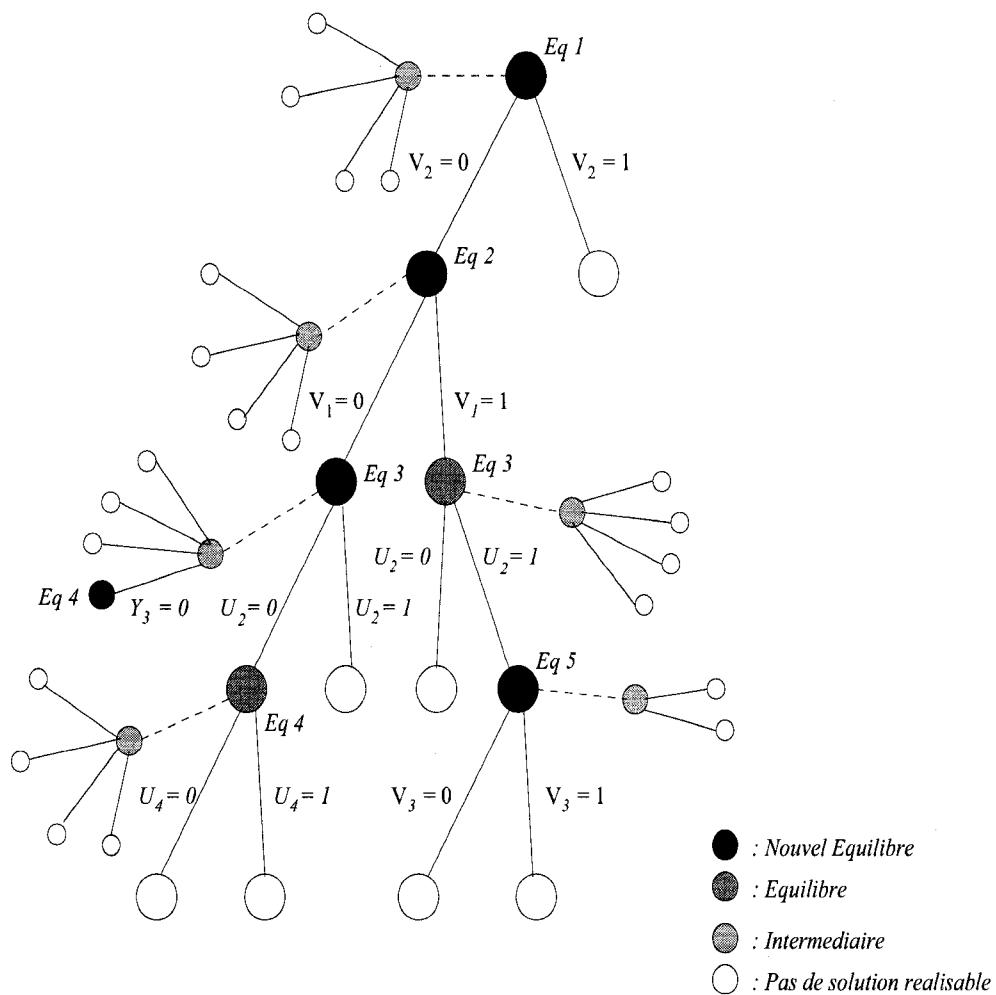


Figure 4.1 Arbre d'exploration avec $E_{\chi}MIP$ sur l'exemple (4.1) : (3×3)

$$\min_{\alpha, \beta, x, y, u, v} f(\alpha, \beta)$$

sujet à

$$\begin{aligned} x_1 + x_2 + x_3 &= 1, & y_1 + y_2 + y_3 &= 1, \\ x_1 + u_1 &\leq 1, & y_1 + v_1 &\leq 1, \\ x_2 + u_2 &\leq 1, & y_2 + v_2 &\leq 1, \\ x_3 + u_3 &\leq 1, & y_3 + v_3 &\leq 1, \\ \alpha - 3y_1 - 2.5y_2 - 5y_3 - 5.5u_1 &\leq 0, & \beta - 5x_1 - x_2 - 2.5x_4 - 4v_1 &\leq 0, \\ \alpha - 2y_1 - 3.5y_2 - 1.5y_3 - 5.5u_3 &\leq 0, & \beta - 4.5x_1 - 1.5x_2 - 3.5x_4 - 4v_2 &\leq 0, \\ \alpha - 4.5y_1 - 0.5y_2 - 5.5y_3 - 5.5u_4 &\leq 0, & \beta - 2.5x_1 - 3.5x_2 - 4x_4 - 4v_3 &\leq 0, \\ -\alpha + 3y_1 + 2.5y_2 + 5y_3 &\leq 0, & -\beta + 5x_1 + x_2 + 2.5x_4 &\leq 0, \\ -\alpha + 2y_1 + 3.5y_2 + 1.5y_3 &\leq 0, & -\beta + 4.5x_1 + 1.5x_2 + 3.5x_4 &\leq 0, \\ -\alpha + 4.5y_1 + 0.5y_2 + 5.5y_3 &\leq 0, & -\beta + 2.5x_1 + 3.5x_3 + 4x_4 &\leq 0, \\ x &\geq 0, & y &\geq 0, \\ u &\in \{0, 1\}^3, & v &\in \{0, 1\}^3. \end{aligned}$$

La résolution de ce programme linéaire mixte 0–1 permet d’obtenir le premier équilibre extrême au premier noeud de l’arbre principal.

Eq1 :

$$\begin{aligned} \alpha &= 4.5, & \beta &= 3.7, \\ X^t &= (0.2, 0, 0.8), & Y^t &= (0, 0.2, 0.8), \\ U^t &= (0, 1, 0), & V^t &= (1, 0, 0). \end{aligned}$$

En fixant la combinaison des variables binaires, l’algorithme crée un sous-arbre de branchement par rapport aux variables continues non nulles, dont la racine est représentée par un noeud intermédiaire dans la Figure 4.1. Aucune solution n’est rencontrée sur cet arbre secondaire et l’algorithme retourne à l’arbre principal pour ajouter une contrainte permettant d’éliminer la combinaison de variables binaires rencontrée à

l'équilibre 1 : $U^t = (0, 1, 0)$ et $V^t = (1, 0, 0)$:

$$(1 - u_1) + u_2 + (1 - u_3) + v_1 + (1 - v_2) + (1 - v_3) \leq 5$$

\Leftrightarrow

$$-u_1 + u_2 - u_3 + v_1 - v_2 - v_3 \leq 1.$$

Cette contrainte n'est cependant utile à ajouter que pour le noeud créé par le branchement $v_2 = 0$. L'autre noeud, créé par le branchement $v_2 = 1$, satisfait déjà cette contrainte. L'algorithme $E\chi MIP$ effectue donc deux branchements, l'un à gauche et l'autre à droite, en ajoutant respectivement les contraintes $v_2 = 0$ et $v_2 = 1$, et le deuxième équilibre extrême est rencontré au noeud obtenu par ajout de la contrainte $v_2 = 0$.

Eq2 :

$$\begin{aligned} \alpha &= 2.786, & \beta &= 4.167, \\ X^t &= (0.667, 0, 0.333), & Y^t &= (0.571, 0.429, 0), \\ U^t &= (0, 1, 0), & V^t &= (0, 0, 1). \end{aligned}$$

Aucune solution n'est rencontrée sur l'arbre secondaire créé à partir de cet équilibre par fixation de la combinaison des variables binaires.

L'algorithme retourne alors à l'arbre principal en ajoutant, à gauche la contrainte $v_1 = 0$ et à droite les contraintes $v_1 = 1$ et $-u_1 + u_2 - u_3 - v_1 - v_2 + v_3 \leq 1$, afin d'éliminer la combinaison de variables binaires de l'équilibre 2. Ainsi, le troisième équilibre extrême est rencontré au noeud obtenu par ajout de la contrainte $v_1 = 0$.

Eq3 :

$$\begin{aligned} \alpha &= 3.056, & \beta &= 3 \\ X^t &= (0.5, 0.5, 0), & Y^t &= (0, 0.778, 0.222) \\ U^t &= (0, 0, 1), & V^t &= (0, 0, 0). \end{aligned}$$

Le quatrième équilibre extrême est ensuite obtenu à un noeud de l'arbre secondaire créé à partir de cet équilibre, par fixation de la combinaison de variables binaires et l'ajout

de la contrainte $y_3 = 0$.

Eq4 :

$$\begin{aligned}\alpha &= 2.75, & \beta &= 3, \\ X^t &= (0.5, 0.5, 0), & Y^t &= (0.5, 0.5, 0), \\ U^t &= (0, 0, 1), & V^t &= (0, 0, 0).\end{aligned}$$

Ce même équilibre est obtenu une deuxième fois au noeud de l'arbre principal créé, après élimination de la combinaison de variables binaires des équilibres 3 et 4, par l'ajout de la contrainte $u_2 = 0$. Aucune solution réalisable n'est rencontrée à partir de ce point et l'algorithme revient au traitement du noeud créé à partir de l'équilibre 2 par l'ajout de la contrainte $v_1 = 1$. L'équilibre 3 est obtenu encore une fois à ce noeud et seul le branchement sur l'arbre principal par ajout de la contrainte $u_2 = 1$, permet d'obtenir le cinquième équilibre extrême.

Eq5 :

$$\begin{aligned}\alpha &= 5.5, & \beta &= 4, \\ X^t &= (0, 0, 1), & Y^t &= (0, 0, 1), \\ U^t &= (1, 1, 0), & V^t &= (1, 1, 0).\end{aligned}$$

Le tableau suivant récapitule les cinq équilibres de Nash extrêmes énumérés :

Tableau 4.1 Équilibre de Nash extrêmes de l'exemple (4.3.1) (3×3)

| Eq | α | β | X | Y | U | V |
|----|----------|---------|-------------------|-------------------|-----------|-----------|
| 1 | 4.5 | 3.7 | (0.2, 0, 0.8) | (0, 0.2, 0.8) | (0, 1, 0) | (1, 0, 0) |
| 2 | 2.786 | 4.167 | (0.667, 0, 0.333) | (0.571, 0.429, 0) | (0, 1, 0) | (0, 0, 1) |
| 3 | 3.056 | 3 | (0.5, 0.5, 0) | (0, 0.778, 0.222) | (0, 0, 1) | (0, 0, 0) |
| 4 | 2.75 | 3 | (0.5, 0.5, 0) | (0.5, 0.5, 0) | (0, 0, 1) | (0, 0, 0) |
| 5 | 5.5 | 4 | (0, 0, 1) | (0, 0, 1) | (1, 1, 0) | (1, 1, 0) |

Exemple 4.3.2 Le jeu bimatriciel suivant, de taille (6×2), est tiré de Winkels (68) et

illustré dans l'article d'Audet et al. (6) :

$$A^t = \begin{pmatrix} 1 & 1 & 3 & 3 & 2.5 & 2.5 \\ 3 & 3 & 1 & 1 & 2.5 & 2.5 \end{pmatrix} \quad \text{et} \quad B^t = \begin{pmatrix} 1 & 0 & -2 & 4 & -1 & 6 \\ 2 & -1 & 2 & -1 & 6 & -1 \end{pmatrix}$$

D'après Winkels (68), ce jeu comporte 11 équilibres de Nash extrêmes. Audet et al. (6) ont cependant montré que ce jeu possède 12 équilibres de Nash extrêmes. L'algorithme $E\chi MIP$ permet aussi d'énumérer les 12 équilibres extrêmes, en générant un arbre principal et des sous-arbres comportant au total 175 noeuds.

La Figure 4.2 résume le comportement de cet algorithme lors de l'énumération de ces équilibres extrêmes. La résolution du programme linéaire mixte 0 – 1 obtenu, permet d'obtenir le premier équilibre extrême au premier noeud de l'arbre principal.

Eq1 :

$$\begin{aligned} \alpha &= 2.5, & \beta &= 2.5, \\ X^t &= (0, 0, 0, 0, 0.5, 0.5), & Y^t &= (0.25, 0.75), \\ U^t &= (0, 0, 1, 1, 0, 0), & V^t &= (0, 0). \end{aligned}$$

En fixant la combinaison des variables binaires, l'algorithme crée un sous-arbre de branchement par rapport aux variables continues non nulles. En ajoutant la contrainte $x_5 = 0$ et en résolvant, le deuxième équilibre extrême est rencontré.

Eq2 :

$$\begin{aligned} \alpha &= 2.5, & \beta &= 0.5, \\ X^t &= (0.5, 0.5, 0, 0, 0, 0), & Y^t &= (0.25, 0.75), \\ U^t &= (0, 0, 1, 1, 0, 0), & V^t &= (0, 0). \end{aligned}$$

Ensuite, en ajoutant à partir de ce noeud la contrainte $x_2 = 0$ et en résolvant, le troisième équilibre extrême est rencontré.

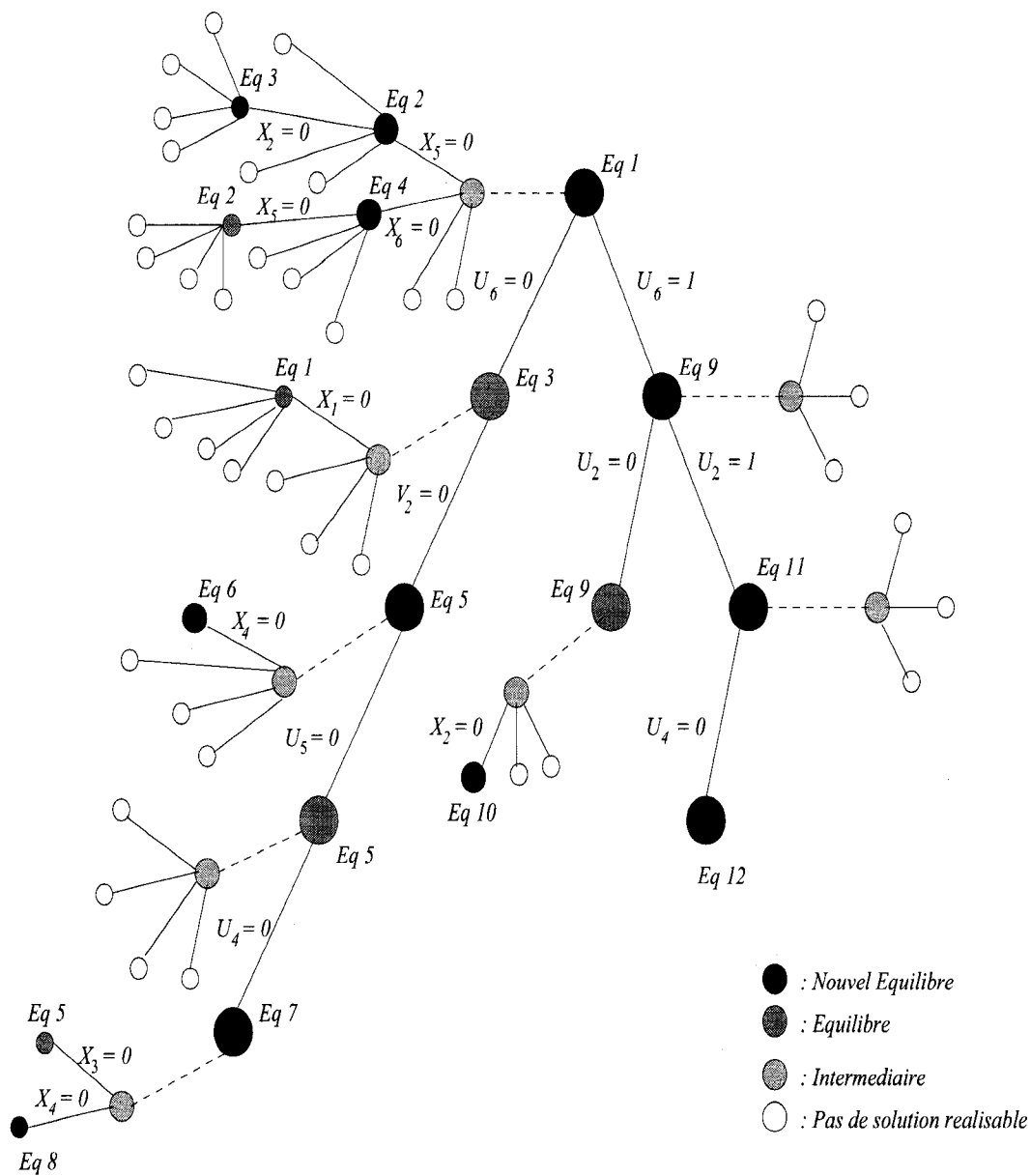


Figure 4.2 Arbre d'exploration avec $E_{\chi}MIP$ sur le jeu bimatriciel de Winkels (68)

Eq3 :

$$\begin{aligned}\alpha &= 2.5, & \beta &= 1.625, \\ X^t &= (0.875, 0, 0, 0, 0, 0.125), & Y^t &= (0.25, 0.75), \\ U^t &= (0, 0, 1, 1, 0, 0), & V^t &= (0, 0).\end{aligned}$$

Comme aucun des autres noeuds à partir des équilibres 2 et 3 ne donne de solution, l'algorithme retourne à la résolution du sous-problème créé par l'ajout de la contrainte $x_6 = 0$, après fixation de la combinaison de variables binaires obtenue avec l'équilibre 1. Ainsi le quatrième équilibre extrême est rencontré.

Eq4 :

$$\begin{aligned}\alpha &= 2.5, & \beta &= -0.125, \\ X^t &= (0, 0.875, 0, 0, 0.125, 0), & Y^t &= (0.25, 0.75), \\ U^t &= (0, 0, 1, 1, 0, 0), & V^t &= (0, 0).\end{aligned}$$

Ensuite, en ajoutant à partir de ce noeud la contrainte $x_5 = 0$ et en résolvant, le deuxième équilibre est rencontré à nouveau. Aucun noeud restant dans cet arbre secondaire ne donne de solution et l'algorithme revient à l'arbre principal. L'algorithme enlève alors les contraintes de fixation des variables binaires et branche sur la variable u_6 , à gauche en ajoutant $u_6 = 0$ et à droite en ajoutant $u_6 = 1$, tout en prenant soin d'ajouter une contrainte permettant d'éliminer la combinaison de variables binaires rencontrée à l'équilibre 1 : $U^t = (0, 0, 1, 1, 0, 0)$ et $V^t = (0, 0)$:

$$(1 - u_1) + (1 - u_2) + u_3 + u_4 + (1 - u_5) + (1 - u_6) + (1 - v_1) + (1 - v_2) \leq 7$$

$$\iff$$

$$-u_1 - u_2 + u_3 + u_4 - u_5 - u_6 - v_1 - v_2 \leq -1.$$

Cette contrainte n'est utile à ajouter que pour le noeud créé par le branchement $u_6 = 0$. L'autre noeud créé par le branchement $u_6 = 1$ satisfait par défaut cette contrainte. Il est à noter que l'élimination des stratégies 2, 4 et 6 (faiblement dominées) du joueur I n'aurait pas permis d'énumérer tous les équilibres extrêmes. Le Tableau 4.2 récapitule

les 12 équilibres de Nash extrêmes énumérés :

Tableau 4.2 Équilibres de Nash extrêmes du jeu bimatriciel de Winkels (68)

| Eq | α | β | X | Y | U | V |
|----|----------|---------|----------------------------|--------------|--------------------|--------|
| 1 | 2.5 | 2.5 | (0, 0, 0, 0, 0.5, 0.5) | (0.25, 0.75) | (0, 0, 1, 1, 0, 0) | (0, 0) |
| 2 | 2.5 | 0.5 | (0.5, 0.5, 0, 0, 0, 0) | (0.25, 0.75) | (0, 0, 1, 1, 0, 0) | (0, 0) |
| 3 | 2.5 | 1.625 | (0.875, 0, 0, 0, 0, 0.125) | (0.25, 0.75) | (0, 0, 1, 1, 0, 0) | (0, 0) |
| 4 | 2.5 | -0.125 | (0, 0.875, 0, 0, 0.125, 0) | (0.25, 0.75) | (0, 0, 1, 1, 0, 0) | (0, 0) |
| 5 | 2.5 | 1.917 | (0, 0, 0, 0.583, 0.417, 0) | (0.75, 0.25) | (1, 1, 1, 0, 0, 0) | (0, 0) |
| 6 | 2.5 | 2.5 | (0, 0, 0, 0, 0.5, 0.5) | (0.75, 0.25) | (1, 1, 1, 0, 0, 0) | (0, 0) |
| 7 | 2.5 | 0.667 | (0, 0, 0.556, 0.444, 0, 0) | (0.75, 0.25) | (1, 1, 0, 0, 0, 0) | (0, 0) |
| 8 | 2.5 | 0.909 | (0, 0, 0.636, 0, 0, 0.364) | (0.75, 0.25) | (1, 1, 0, 0, 0, 0) | (0, 0) |
| 9 | 3 | 0.5 | (0.5, 0.5, 0, 0, 0, 0) | (0, 1) | (0, 0, 1, 1, 1, 1) | (0, 0) |
| 10 | 3 | 2 | (1, 0, 0, 0, 0, 0) | (0, 1) | (0, 0, 1, 1, 1, 1) | (1, 0) |
| 11 | 3 | 0.667 | (0, 0, 0.556, 0.444, 0, 0) | (1, 0) | (1, 1, 0, 0, 1, 1) | (0, 0) |
| 12 | 3 | 4 | (0, 0, 0, 1, 0, 0) | (1, 0) | (1, 1, 1, 0, 1, 1) | (0, 1) |

4.3.2 Jeux polymatriciels

Exemple 4.3.3 Soit un jeu polymatriciel à 3 joueurs, de dimension $(3 \times 3 \times 3)$, dont les matrices se détaillent comme suit :

$$A = \left(\begin{array}{ccc|ccc} 1 & 1 & -1 & 2 & 3 & 2.5 \\ 2 & -1 & -1 & -1 & 0 & 2 \\ 3 & -1.5 & -1 & -1 & 2 & 1 \end{array} \right) \quad B = \left(\begin{array}{ccc|ccc} -1 & 2 & 1 & -2 & 3 & 1 \\ 3 & 0 & 3.5 & 1 & -1 & -1 \\ 3 & 3.5 & 3 & 2 & 1 & -2 \end{array} \right)$$

$$C = \left(\begin{array}{ccc|ccc} -3 & -1 & 1 & 1 & 2 & -3 \\ 4 & 1 & 4 & 2 & 1 & 2 \\ 1 & 2 & 2.2 & 3 & 2 & 4 \end{array} \right)$$

L'algorithme $E_{\chi}MIP$ permet d'énumérer les sept équilibres de Nash extrêmes de ce jeu. La Figure 4.3 permet d'illustrer le comportement de $E_{\chi}MIP$ lors de l'énuméra-

tion de ces équilibres extrêmes. Une partie des 107 noeuds créés par l'algorithme y est illustrée.

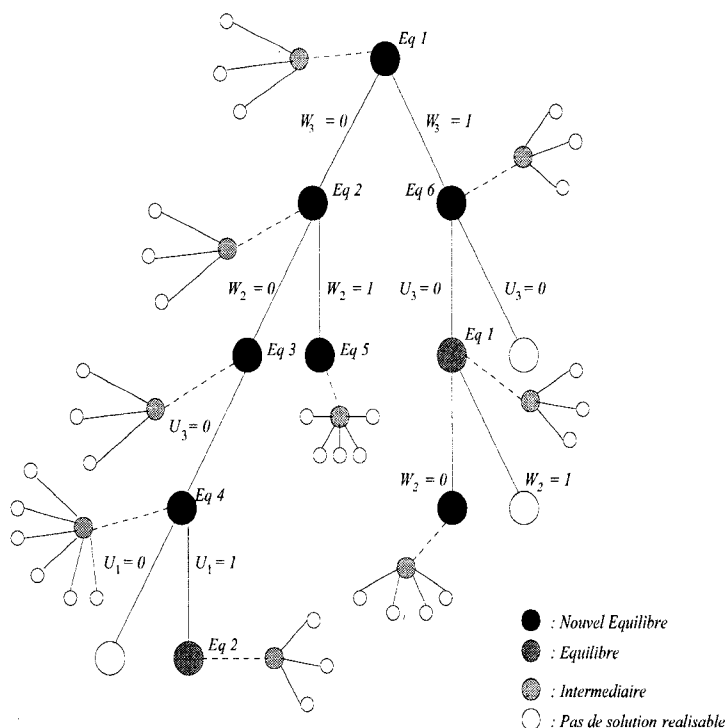


Figure 4.3 Arbre d'exploration avec $E\chi MIP$ sur l'exemple (4.3.3) : $(3 \times 3 \times 3)$

La résolution du programme linéaire mixte 0 – 1 permet d'obtenir le premier équilibre au premier noeud de l'arbre principal.

Eq1 :

$$\alpha = 5, \quad \beta = 4, \quad \gamma = 6,$$

$$X^t = (0, 0, 1), \quad Y^t = (1, 0, 0), \quad Z = (0, 1, 0),$$

$$U^t = (1, 1, 0), \quad V^t = (0, 1, 1), \quad W = (1, 0, 1).$$

En fixant la combinaison des variables binaires, l'algorithme crée un sous-arbre de branchement par rapport aux variables continues non nulles. La racine de cet arbre secondaire est représentée par un noeud intermédiaire dans la Figure 4.3. Aucune solution réalisable n'est rencontrée sur cet arbre secondaire et l'algorithme retourne à l'arbre

principal pour ajouter une contrainte permettant d'éliminer la combinaison de variables binaires rencontrée à l'équilibre 1 : $U^t = (1, 1, 0)$, $V^t = (0, 1, 1)$ et $W^t = (1, 0, 1)$:

$$u_1 + u_2 + (1 - u_3) + (1 - v_1) + v_2 + v_3 + w_1 + (1 - w_2) + w_3 \leq 8$$

\iff

$$u_1 + u_2 - u_3 - v_1 + v_2 + v_3 + w_1 - w_2 + w_3 \leq 5.$$

Cette contrainte n'est cependant utile à ajouter que pour le noeud créé par le branchement $w_3 = 1$. L'autre noeud créé par le branchement $w_3 = 0$, satisfait par défaut cette contrainte. L'algorithme effectue donc deux branchements, l'un à gauche et l'autre à droite, en ajoutant respectivement les contraintes $w_3 = 0$ et $w_3 = 1$ et le deuxième équilibre est rencontré au noeud obtenu par ajout de la contrainte $w_3 = 0$.

Eq2 :

$$\begin{aligned} \alpha &= 4, & \beta &= 3, & \gamma &= 5, \\ X^t &= (0, 1, 0), & Y^t &= (1, 0, 0), & Z &= (0, 0, 1), \\ U^t &= (1, 0, 0), & V^t &= (0, 1, 1), & W &= (1, 1, 0). \end{aligned}$$

À partir de cet équilibre, l'exploration de l'arbre secondaire, après fixation de la combinaison des variables binaires, ne donne aucun équilibre. L'algorithme retourne au traitement des noeuds de l'arbre principal en ajoutant la contrainte :

$$u_1 - u_2 - u_3 - v_1 + v_2 + v_3 + w_1 + w_2 - w_3 \leq 4.$$

Le branchement par rapport à la variable w_2 , à gauche en mettant $w_2 = 0$ et à droite en mettant $w_2 = 1$, donnera à gauche les équilibres 3 et 4 et à droite l'équilibre 5.

Ensuite, l'algorithme revient au traitement de la branche $w_3 = 1$, issue directement du premier noeud de l'arbre principal. Les équilibres 6 puis 7 sont alors rencontrés.

Le Tableau 4.3 donne les sept équilibres extrêmes rencontrés dans ce jeu.

Tableau 4.3 Équilibres de Nash extrêmes de l'exemple (4.3.3) ($3 \times 3 \times 3$)

| Eq | α | β | γ | X^t | Y^t | Z^t | U^t | V^t | W^t |
|----|----------|---------|----------|--------------------|-----------------|-----------------|-----------|-----------|-----------|
| 1 | 5 | 4 | 6 | (0, 0, 1) | (1, 0, 0) | (0, 1, 0) | (1, 1, 0) | (0, 1, 1) | (1, 0, 1) |
| 2 | 4 | 3 | 5 | (0, 1, 0) | (1, 0, 0) | (0, 0, 1) | (1, 0, 0) | (0, 1, 1) | (1, 1, 0) |
| 3 | 4 | 2.286 | 5.143 | (0, 0.29, 0.71) | (1, 0, 0) | (0, 0, 1) | (1, 0, 0) | (0, 1, 1) | (1, 0, 0) |
| 4 | 3 | 1.487 | 4.713 | (0.56, 0.43, 0.01) | (0.73, 0, 0.27) | (0, 0.09, 0.91) | (0, 0, 0) | (0, 1, 0) | (1, 0, 0) |
| 5 | 2.5 | 1.2 | 4.9 | (0.6, 0.4, 0) | (0.5, 0, 0.5) | (0, 0, 1) | (0, 0, 1) | (0, 1, 0) | (1, 1, 0) |
| 6 | 2 | 4 | 6 | (1, 0, 0) | (0, 0, 1) | (0, 1, 0) | (0, 1, 1) | (1, 1, 0) | (1, 0, 1) |
| 7 | 3 | 4 | 6 | (0, 0, 1) | (0.5, 0, 0.5) | (0, 1, 0) | (0, 1, 0) | (0, 1, 0) | (1, 0, 1) |

4.3.3 Les jeux étendus à deux joueurs

Exemple 4.3.4 En appliquant l'algorithme $E\chi MIP$ au jeu étendu à deux joueurs illustré par la Figure 3.1 de von Stengel et al. (64), la forme stratégique permet comme avec EEE (6), d'obtenir trois équilibres de Nash extrêmes. La représentation séquentielle permet d'obtenir les séquences de mouvements $S_1 = \{\emptyset, L, R, LS, LT\}$ et $S_2 = \{\emptyset, l, r\}$, pour les joueurs I et II respectivement. L'algorithme $E\chi MIP$ énumère quatre points extrêmes pour le programme linéaire mixte 0 – 1 associé à la représentation séquentielle.

Tableau 4.4 Équilibres extrêmes de la forme séquentielle du jeu de von Stengel et al. (64)

| q. | X | x_1 | | | x_2 | | | α_1 | | α_2 | | | | |
|----|---------|-------|---|---|-------|-----|---|------------|-----|------------|---|---|-----|-----|
| 1 | $= X^1$ | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 3 | 3 | 2 | 4 | 0 |
| 2 | | 1 | 1 | 0 | 2/3 | 1/3 | 1 | 1/3 | 2/3 | 4 | 4 | 4 | 2/3 | 2/3 |
| 3 | | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 3 | 3 | 3 | 4 | 0 |
| 4 | | 1 | 0 | 1 | 0 | 0 | 1 | 2/3 | 1/3 | 3 | 3 | 3 | 4 | 0 |

Nous pouvons constater que les équilibres 1 et 3 ont les mêmes valeurs pour les variables X , mais des valeurs différentes pour les variables duales α_1 .

Exemple 4.3.5 En appliquant l'algorithme $E\chi MIP$ à l'exemple de Kohlber et Mertens (31) illustré par la Figure 4.4, nous obtenons des résultats différents dépendamment de

la représentation utilisée.

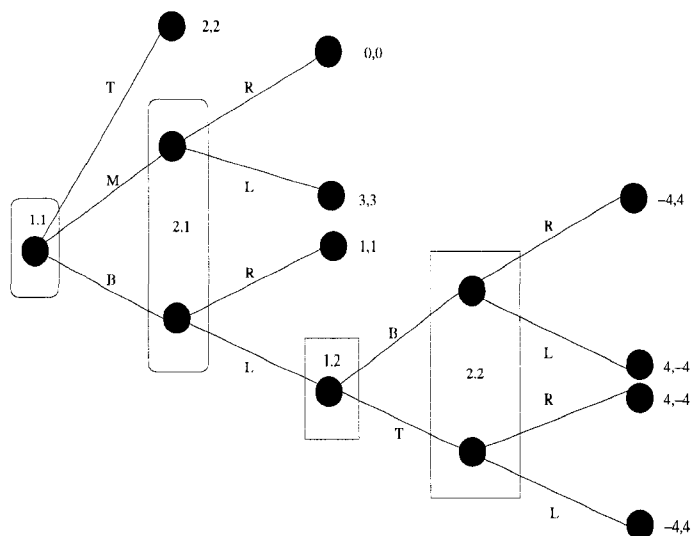


Figure 4.4 Jeu étendu de Kohlberg et Mertens (31)

La représentation stratégique contient les stratégies pures suivantes $\{T, M, BB, BT\}$ pour le joueur I et $\{R, LR, LL\}$ pour le joueur II.

$$A = \begin{pmatrix} 2 & 2 & 2 \\ 0 & 3 & 3 \\ 1 & -4 & 4 \\ 1 & 4 & -4 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 2 & 2 \\ 0 & 3 & 3 \\ 1 & 4 & -4 \\ 1 & -4 & 4 \end{pmatrix}.$$

Sept équilibres de Nash extrêmes sont obtenus pour cette représentation stratégique (Tableau 4.5).

En considérant la forme séquentielle de ce jeu étendu à deux joueurs, les séquences du joueur I sont $S_1 = \{\emptyset, T, M, B, BB, BT\}$ et les séquences du joueur II sont $S_2 = \{\emptyset, R, L, LR, LL\}$. Les matrices des gains des deux joueurs sont alors :

Tableau 4.5 Équilibres extrêmes de la forme stratégique du jeu de Kohlberg et Mertens (31)

| Eq. | x_1 | | | | x_2 | | | α_1 | α_2 |
|-----|-------|---|---|---|-------|-------|-------|------------|------------|
| 1 | 0 | 1 | 0 | 0 | 0 | 7/8 | 1/8 | 3 | 3 |
| 2 | 0 | 1 | 0 | 0 | 0 | 1/8 | 7/8 | 3 | 3 |
| 3 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 2 |
| 4 | 1 | 0 | 0 | 0 | 2/3 | 0 | 1/3 | 2 | 2 |
| 5 | 1 | 0 | 0 | 0 | 2/3 | 1/3 | 0 | 2 | 2 |
| 6 | 1 | 0 | 0 | 0 | 1/3 | 13/24 | 1/8 | 2 | 2 |
| 7 | 1 | 0 | 0 | 0 | 1/3 | 1/8 | 13/24 | 2 | 2 |

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -4 & 4 \\ 0 & 0 & 0 & 4 & -4 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & -4 \\ 0 & 0 & 0 & -4 & 4 \end{pmatrix},$$

et les matrices E_1 et E_2 suivantes permettent de suivre les liens entre les différentes séquences.

$$E_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \quad \text{et} \quad E_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 1 \end{pmatrix}.$$

Dix points extrêmes sont énumérés pour le programme linéaire mixte 0 – 1 associés à la représentation séquentielle de ce jeu (Tableau 4.6).

Dans le Tableau 4.6, les points extrêmes 4 et 5 correspondent au même équilibre de Nash extrême mais ont des variables duales α_1 . Le vecteur X^2 des stratégies mixtes du point extrême 2 peut être obtenu par une combinaison convexe $(\frac{1}{2}, \frac{1}{2})$ des vecteurs x_2 des

Tableau 4.6 Équilibres extrêmes de la forme séquentielle du jeu étendu de Kohlberg et Mertens (31)

| $q.$ | X | x_1 | x_2 | | | | α_1 | | α_2 | | | | |
|------|-------------------------------|-------------|-------|-----|-----|-------|------------|---|------------|-----|---|---|---|
| 1 | | 1 0 1 0 0 0 | 1 | 0 | 1 | 7/8 | 1/8 | 3 | 3 | 3 | 3 | 3 | 0 |
| 2 | $= \frac{1}{2}(X^1 + X^3)$ | 1 0 1 0 0 0 | 1 | 0 | 1 | 1/2 | 1/2 | 3 | 3 | 0 | 3 | 3 | 0 |
| 3 | | 1 0 1 0 0 0 | 1 | 0 | 1 | 1/8 | 7/8 | 3 | 3 | 3 | 3 | 3 | 0 |
| 4 | | 1 1 0 0 0 0 | 1 | 1 | 0 | 0 | 0 | 2 | 2 | 0 | 2 | 0 | 0 |
| 5 | $= X^4$ | 1 1 0 0 0 0 | 1 | 1 | 0 | 0 | 0 | 2 | 2 | 1 | 2 | 0 | 0 |
| 6 | | 1 1 0 0 0 0 | 1 | 2/3 | 1/3 | 1/3 | 0 | 2 | 2 | 4/3 | 2 | 0 | 0 |
| 7 | | 1 1 0 0 0 0 | 1 | 2/3 | 1/3 | 0 | 1/3 | 2 | 2 | 4/3 | 2 | 0 | 0 |
| 8 | | 1 1 0 0 0 0 | 1 | 1/3 | 2/3 | 13/24 | 1/8 | 2 | 2 | 5/3 | 2 | 0 | 0 |
| 9 | $= \frac{1}{2}(X^8 + X^{10})$ | 1 1 0 0 0 0 | 1 | 1/3 | 2/3 | 1/3 | 1/3 | 2 | 2 | 0 | 2 | 0 | 0 |
| 10 | | 1 1 0 0 0 0 | 1 | 1/3 | 2/3 | 1/8 | 13/24 | 2 | 2 | 5/3 | 2 | 0 | 0 |

équilibres extrêmes 1 et 3. Le point extrême 2 a cependant des valeurs différentes pour les variables duales α_1 . Toutefois $Proj_{(x)}(q^2) \notin Ext(NE)$.

L'information supplémentaire présente dans le Tableau 4.6 et non dans le Tableau 4.5 peut être utilisée pour un raffinement. En effet, le point extrême 2 correspond à un équilibre sous-jeu parfait. D'une part, le joueur II choisit la stratégie mixte $1/2 - 1/2$ entre L et R , dans le sous-jeu à somme nulle. Ce choix correspond effectivement à l'unique équilibre de ce sous-jeu. Cet équilibre sous-jeu parfait n'a pas été trouvé lors de l'énumération basée sur la représentation stratégique. La raison est que, dans cet équilibre, la variable duale pour le second ensemble informationnel (1.2) du joueur I est nulle, ce qui correspond au gain du sous-jeu. Les deux mouvements T et B du joueur I à cet ensemble informationnel, correspondant aux séquences BT et BB , ont en effet des contraintes serrées relativement à la variable duale. Ceci est dû à l'un des branchements binaires de l'algorithme.

D'autre part, les stratégies mixtes $7/8 - 1/8$ pour L et R pour les points extrêmes q^1 et q^3 donnent un gain égal à 3 au joueur I à l'ensemble informationnel (1.2). Ceci produit seulement une seule contrainte serrée pour uniquement une des séquences BT ou BB .

Le gain égal à 3 laisse le joueur 1 indifférent entre choisir B (qui n'est pas choisie) et le choix M de l'équilibre à l'ensemble informationnel (1.1). Nous observons aussi la même chose pour le point extrême q^9 par rapport aux points q^8 et q^{10} .

Cet exemple montre que l'utilisation de la représentation séquentielle peut permettre d'obtenir un équilibre extrême sous-jeu parfait qui ne serait pas obtenu si la représentation stratégique est utilisée. Ceci est dû au fait que la représentation séquentielle est plus fidèle à la forme d'origine du jeu étendu que la représentation stratégique.

4.4 L'algorithme *EEE* revisité

Cette section illustre l'implantation de l'algorithme *EEE* pour la forme séquentielle d'un jeu étendu à deux joueurs.

Nous avons modifié l'algorithme proposé par Audet *et al.* (6; 7) afin de l'adapter à la représentation séquentielle d'un jeu étendu à deux joueurs. L'algorithme *EEE* adopte une approche énumérative et sélective en étudiant seulement les sommets qui satisfont les conditions des écarts complémentaires :

$$\begin{cases} x_1^t A x_2 = x_1^t E_1^t \alpha_1, \\ x_1^t B x_2 = \alpha_2 E_2^t x_2. \end{cases} \Rightarrow \begin{cases} x_1^t (E_1^t \alpha_1 - A x_2) = 0, \\ (\alpha_2^t E_2 - x_1^t B) x_2 = 0. \end{cases}$$

Nous considérons deux programmes linéaires à objectifs paramétrés :

$$P(x_2) \equiv \max_{x_1, \alpha_2} x_1^t A x_2 - e_1^t \alpha_1,$$

$$Q(x_1) \equiv \max_{x_2, \alpha_1} x_1^t B x_2 - e_2^t \alpha_2.$$

L'algorithme *EEE* construit un arbre d'exploration où à chaque noeud est associée une

paire de *sous-programmes* linéaires à objectifs paramétrés, $\tilde{P}(x_2)$ et $\tilde{Q}(x_1)$. À chaque noeud est associée une variable x_1 ou x_2 représentant une stratégie réalisable, pour l'un ou l'autre des sous-problèmes courants.

Les programmes $\tilde{P}(x_2)$ et $\tilde{Q}(x_1)$ sont quasi-identiques à $P(x_2)$ et $Q(x_1)$, mais avec en plus un ensemble d'inégalités ou de contraintes de non-négativité qui sont mises à égalité.

Ainsi, nous distinguons quatre formes possibles que peuvent prendre les problèmes $\tilde{P}(\cdot)$ et $\tilde{Q}(\cdot)$:

- $P^i(\cdot) \equiv \tilde{P}(\cdot)$ avec une contrainte $x_{1i} = 0$ à la place de $x_{1i} \geq 0$.
- $P_j(\cdot) \equiv \tilde{P}(\cdot)$ avec une contrainte $x_1^t B_j = \alpha_2^t E_2$ à la place de $x_1^t B_j \leq \alpha_2^t E_2$.
- $Q^j(\cdot) \equiv \tilde{Q}(\cdot)$ avec une contrainte $x_{2j} = 0$ à la place de $x_{2j} \geq 0$.
- $Q_i(\cdot) \equiv \tilde{Q}(\cdot)$ avec une contrainte $A_i x_2 = E_1^t \alpha_1$ à la place de $A_i x_2 \leq E_1^t \alpha_1$.

À chaque noeud de cet arbre, une optimisation de $\tilde{P}(x_2)$ et une autre de $\tilde{Q}(x_1)$ sont lancées successivement (l'ordre dépend des cas) et trois sortes de résultats différents peuvent être obtenus :

- $\tilde{P}(\cdot)$ ou $\tilde{Q}(\cdot)$ est non-réalisable.
- $\tilde{P}(\cdot)$ et $\tilde{Q}(\cdot)$ sont réalisables mais un manque d'information ne permet pas de déduire un point d'équilibre.
- $\tilde{P}(\cdot)$ et $\tilde{Q}(\cdot)$ sont réalisables et un point d'équilibre peut être déduit à partir des informations réunies.

Mis à part le premier cas où le noeud courant sera éliminé, dans les deux autres il sera remplacé par ses descendants directs. Chaque *fils* possède une seule différence par rapport à son *père* qui consiste en une contrainte supplémentaire pour $\tilde{P}(\cdot)$ ou $\tilde{Q}(\cdot)$ selon la règle de branchement choisie.

Cette règle de branchement se détermine à partir de deux paramètres τ_i et π_j , tels que :

$$\tau_i = \begin{cases} x_{1i}(E_1^t \alpha_1 - A_i x_2), & \text{si } \tilde{P}(\cdot) \text{ n'a pas été changé en } P^i(\cdot) \\ & \text{et } \tilde{Q}(\cdot) \text{ n'a pas été changé en } Q_i(\cdot), \\ -1 & , \text{ sinon,} \end{cases}$$

et

$$\pi_j = \begin{cases} (\alpha_2^t E_2 - x_1^t B_{.j}) x_{2j}, & \text{si } \tilde{Q}(\cdot) \text{ n'a pas été changé en } Q^j(\cdot) \\ & \text{et } \tilde{P}(\cdot) \text{ n'a pas été changé en } P_j(\cdot), \\ -1 & , \text{ sinon.} \end{cases}$$

En choisissant, pour $i \in \{1, \dots, n\}$ et $j \in \{1, \dots, m\}$, les i et j qui maximisent les valeurs de τ_i et π_j , trois cas différents permettent de conclure sur la règle de branchement à adopter :

- Si $-1 \neq \tau_i \geq \pi_j$, le noeud courant aura deux descendants :

$$(x_2, P^i(\cdot), \tilde{Q}(\cdot)) \quad \text{et} \quad (x_1, Q_i(\cdot), \tilde{P}(\cdot))$$

- Si $\tau_i < \pi_j$, le noeud courant aura deux descendants :

$$(x_1, Q^j(\cdot), \tilde{P}(\cdot)) \quad \text{et} \quad (x_2, P_j(\cdot), \tilde{Q}(\cdot))$$

- Si $\tau_i = \pi_j = -1$, (x_1, x_2) est un équilibre extrême et le noeud courant aura quatre descendants :

$$\{(x_2, P^i(\cdot), \tilde{Q}(\cdot)) : x_{1i} > 0\}, \quad \{(x_1, Q_i(\cdot), \tilde{P}(\cdot)) : A_i x_2 < E_1^t \alpha_1\},$$

$$\{(x_1, Q^j(\cdot), \tilde{P}(\cdot)) : x_{2j} > 0\} \quad \text{et} \quad \{(x_2, P_j(\cdot), \tilde{Q}(\cdot)) : x_1^t B_{.j} < \alpha_2^t E_2\}$$

Quand la profondeur du noeud courant est supérieure à $(n + m)$, la solution (x_1, x_2) obtenue est un équilibre de Nash extrême, car chacune des conditions des écarts complémentaires est satisfaite.

L'algorithme *EEE* adapté à la formulation séquentielle permet d'énumérer tous les équilibres de Nash extrêmes de la formulation séquentielle d'un jeu étendu à deux joueurs.

4.5 Implantation en arithmétique exacte et le logiciel *XGame Solver*

Au début de notre projet, notre but était d'élaborer des algorithmes permettant l'énumération des équilibres de Nash extrêmes pour un certain nombre de jeux et de proposer des méthodes automatiques de raffinement de ces équilibres.

Comme première étape, nous avons implanté des algorithmes d'énumération en interaction avec le logiciel d'optimisation *Cplex*. L'importance de la précision numérique est devenue plus grande lorsque nous avons voulu rendre nos implantations indépendantes en codant à notre propre manière l'algorithme du *Simplexe* (19). En effet, l'implantation d'une arithmétique exacte s'est révélée être problématique sans l'utilisation de bibliothèques. Ainsi, pour notre logiciel *XGame-Solver*, nous avons décidé de faire appel à une bibliothèque d'arithmétique exacte facile à implanter et très efficace.

Cette section détaille les différentes classes de l'implantation en arithmétique exacte des deux algorithmes *EEE* et *E_χMIP* ainsi que les différentes possibilités offertes à l'utilisateur de l'application *Xgame Solver*.

4.5.1 Implantation en arithmétique exacte

Dans nos implantations récentes des algorithmes *EEE* et *E χ MIP*, les informations numériques sont sauvegardées en utilisant des rationnels. Un rationnel est défini par une paire de nombres entiers : un numérateur et un dénominateur. Notre librairie de classes en arithmétique exacte contient : les classes *BigInteger* pour les grands nombres entiers, la classe *Rational* pour les nombres rationnels, la classe *Simplex* pour la résolution en utilisant l'algorithme du Simplexe (19) et la classe *Node* pour la gestion des branchements de l'algorithme *E χ MIP*. Pour l'implantation de l'algorithme *EEE*, nous utilisons des méthodes de branchement permettant de calquer les différentes étapes de cet algorithme.

Durant l'implantation de l'algorithme *E χ MIP*, nous avons observé que le numérateur ou le dénominateur d'un nombre rationnel peut dépasser en valeur le plus grand entier INTMAX de la librairie de programmation en C/C++. Il existe à la disposition de la communauté scientifique un nombre assez grand de bibliothèques d'arithmétique exacte dans différents langages informatiques. Outre le fait que certaines de ces bibliothèques sont compliquées à implanter et à adapter à nos propres besoins, d'autres sont très efficaces et très simples à utiliser. Pour ces raisons, nous avons choisi l'ensemble des classes *BigInteger* implanté et couramment mis à jour par McCutchen (44).

Les sous-sections suivantes décrivent les différentes classes utilisées dans nos implantations des algorithmes *EEE* et *E χ Mip*.

4.5.2 La librairie *BigInteger*

Les classes de la librairie *BigIntegers* définissent un nouveau type d'entiers qui seront utilisés lors de l'énumération des équilibres de Nash extrêmes. Ces classes permettent de surdéfinir les opérations élémentaires pour des grands entiers. Elles permettent aussi

d'augmenter dynamiquement la taille d'un grand entier en cas de besoin. Les différents tests que nous avons effectués sur cette librairie se sont avérés très probants. Nous remercions à cette occasion leur auteur Matt McCutchen (44) pour ses efforts.

4.5.3 La classe *Rational*

La classe *Rational* est basée sur les classes *BigInteger*. Un rationnel consiste en deux grands entiers : un numérateur et un dénominateur. Nous avons surdéfini les opérations élémentaires pour ces rationnels et nous utilisons une fonction permettant de trouver le plus grand dénominateur commun entre les deux éléments de chaque rationnel. Tous les rationnels sont donc toujours sauvegardés sous leur forme la plus simple.

4.5.4 La classe *Simplex*

La classe *Simplex* définit quant à elle un dictionnaire (16) ainsi que l'ensemble des opérations de l'algorithme du *Simplexe* (19). Chaque sous-programme linéaire obtenu lors des branchements successifs de l'algorithme *EXMIP* ou *EEE* est représenté sous la forme d'un dictionnaire et sera résolu en utilisant les méthodes de cette classe. Un dictionnaire représente les coefficients des variables dans le sous-programme linéaire à résoudre. Toutes les contraintes sont préalablement converties en des contraintes d'égalité et la résolution commence par résoudre un programme auxiliaire dans le cas où un des membres de droite des contraintes est négatif. Si le programme auxiliaire est réalisable alors la résolution avec la solution initiale trouvée a lieu.

4.5.5 La classe *Node*

Enfin, la classe *Node* représente la structure de l'algorithme $E\chi MIP$. Chaque noeud contient le dictionnaire du sous-programme linéaire courant et un pointeur vers le noeud père. Cette classe contient aussi les différentes méthodes de branchement de l'algorithme $E\chi MIP$.

En ce qui concerne EEE , nous n'utilisons pas une telle classe mais nous faisons appel à des méthodes de branchement calquant la structure de l'algorithme EEE .

En pratique, l'algorithme $E\chi MIP$ peut être utilisé afin d'énumérer tous les équilibres de Nash extrêmes des jeux bimatriciels, polymatriciels à trois joueurs et des formes séquentielles des jeux étendus à deux joueurs. L'algorithme EEE peut quant à lui être utilisé afin d'énumérer les équilibres de Nash extrêmes d'un jeu bimatriciel et de la forme séquentielle d'un jeu étendu à deux joueurs. Pour chaque forme de jeux, les deux algorithmes sont disponibles en utilisant les bibliothèques de *Cplex* ou les bibliothèques en arithmétique exacte. Tous nos codes sources sont publiés sur le lien internet <http://www.gerad.ca/~slimb> et peuvent être compilés et exécutés sous MSWindows, Linux, Unix et Mac Os.

4.5.6 Résultats numériques

Cette partie présente une analyse des temps comparatifs de calcul entre les algorithmes EEE et $E\chi MIP$, sur une machine SPARC station ULTRA 2 (Solaris 2.4-27), sur des jeux bimatriciels générés aléatoirement, de taille et de densité différentes. Nous y présentons aussi un tableau des temps de calcul de l'algorithme $E\chi MIP$ sur des jeux polymatriciels à trois joueurs générés aléatoirement, de taille et de densité différentes. Les abréviations suivantes sont utilisées :

- d : densité des matrices des gains.

- Nds : nombre de noeuds générés.
- Temps : temps de calcul en secondes.
- E : nombre d'équilibres extrêmes énumérés.
- μ : moyenne sur dix problèmes de même taille.
- σ : écart type sur dix problèmes de même taille.

Tableau 4.7 Jeux bimatriciels avec $E\chi MIP$ vs EEE

| | | $d = 0.12$ | | | | | | $d = 0.25$ | | | | | |
|---------|----------|-------------------|-----------------------|----------|-------------------|-----------------------|----------|-------------|-----------------------|----------|-------------------|-----------------------|----------|
| $n = m$ | | $E\chi MIP$ | | | EEE | | | $E\chi MIP$ | | | EEE | | |
| | n | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> |
| 5 | μ | 1432.6 | 1.93 | 18.2 | 8037.2 | 2.22 | 18.2 | 655.6 | 1.07 | 13.2 | 2981 | 0.93 | 13.2 |
| | σ | 647.1 | 0.69 | 4.9 | 10617.4 | 1.24 | 4.9 | 434.7 | 0.64 | 8.4 | 1894 | 0.62 | 8.4 |
| 7 | μ | 11156 | 19.42 | 56.4 | 185544 | 59 | 56.4 | 1984 | 4.69 | 41 | 55883 | 18.4 | 41 |
| | σ | 12353 | 6.9 | 29.5 | 13483 | 26 | 29.5 | 2029 | 4.63 | 36.1 | 10213 | 21 | 36.1 |
| 10 | μ | 176921 | 543 | 271.7 | 2.2×10^7 | 8311 | 271.7 | 12380 | 55.8 | 247 | 1.9×10^7 | 7109 | 247 |
| | σ | 134527 | 292.3 | 166.8 | 14073 | 6201 | 166.8 | 7184 | 39.7 | 160.1 | 13700 | 6614 | 160.1 |
| 14 | μ | 1.5×10^6 | 20893 | 2881 | | $> 4 \times 10^9$ | 2881 | 228498 | 9736 | 2387 | 1.3×10^8 | 151843 | 2387 |
| | σ | 5443 | 16518 | 2190 | | 2190 | | 18369 | 7012 | 3801 | 1.5×10^8 | 236496 | 3801 |
| | | $d = 0.50$ | | | | | | $d = 1.00$ | | | | | |
| $n = m$ | | $E\chi MIP$ | | | EEE | | | $E\chi MIP$ | | | EEE | | |
| | n | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> |
| 5 | μ | 114.6 | 0.24 | 6.4 | 730.8 | 0.24 | 6.4 | 22.6 | 0.09 | 3.4 | 86.3 | 0.04 | 3.4 |
| | σ | 162.7 | 0.27 | 7.0 | 1049.9 | 0.34 | 7.0 | 11 | 0.05 | 1.2 | 38.8 | 0.02 | 1.2 |
| 7 | μ | 27.8 | 0.22 | 3.8 | 168.5 | 0.08 | 3.8 | 32 | 0.33 | 4.4 | 197.7 | 0.09 | 4.4 |
| | σ | 24.8 | 0.18 | 3.4 | 118.3 | 0.04 | 3.4 | 16.1 | 0.17 | 2 | 97.6 | 0.05 | 2 |
| 10 | μ | 33 | 0.98 | 5.6 | 409.4 | 0.23 | 5.6 | 50.6 | 3.18 | 10.4 | 900.9 | 0.54 | 10.4 |
| | σ | 18.1 | 0.52 | 2.9 | 196.6 | 0.11 | 2.9 | 21.1 | 1.33 | 4.7 | 436.2 | 0.27 | 4.7 |
| 14 | μ | 107.4 | 16.77 | 23.4 | 2715 | 1.98 | 23.4 | 119.6 | 41.12 | 25.2 | 4166.8 | 3.5 | 25.2 |
| | σ | 40.2 | 10.25 | 9.2 | 1121.2 | 0.87 | 9.2 | 46.6 | 18.85 | 9.7 | 1577.4 | 1.52 | 9.7 |

Le Tableau 4.7 donne une comparaison des temps de calculs, entre les algorithmes EEE et $E\chi MIP$ sur des jeux bimatriciels générés aléatoirement.

Selon le temps nécessaire à l'énumération des équilibres de Nash extrêmes, nous observons que l'algorithme EEE est plus rapide que $E\chi MIP$ sur les jeux de densité moyenne ou grande. Le contraire est aussi vrai puisque $E\chi MIP$ est plus rapide que EEE sur les jeux de faible densité. Ceci est notamment dû à la formulation linéaire mixte 0 – 1 sur laquelle se base $E\chi MIP$ en comparaison avec la formulation biniveau sur laquelle se base EEE . En effet, les branchements de l'algorithme sont plus efficaces lorsque les sous-programmes à résoudre sont de densité moyenne ou grande. Dans le

même temps, la résolution de programmes linéaires mixtes 0 – 1 est plus lente lorsque les sous-programmes à résoudre sont de densité moyenne ou grande. Nous observons l'effet inverse lorsque les sous-programmes à résoudre sont de faible densité.

Tableau 4.8 Jeux polymatriciels à trois joueurs avec $E\chi MIP$; $m_1 = m_2 = m_3$

| $d = 0.12$ | | | | | $d = 0.25$ | | | | |
|-------------------|----------|------------|-----------------------|----------|-------------------|----------|------------|-----------------------|----------|
| $m_1 = m_2 = m_3$ | | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> | $m_1 = m_2 = m_3$ | | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> |
| 3 | μ | 281.7 | 1.365 | 10.7 | 3 | μ | 40.9 | 0.188 | 4.6 |
| | σ | 230.1 | 1.272 | 3.6 | | σ | 36.9 | 0.142 | 2.4 |
| 5 | μ | 6532.3 | 15.981 | 29.8 | 5 | μ | 408.5 | 1.719 | 12.1 |
| | σ | 7981.8 | 9.138 | 13.7 | | σ | 679.3 | 2.681 | 12 |
| 7 | μ | 46476.9 | 197.443 | 129.8 | 7 | μ | 419.4 | 4.472 | 21.6 |
| | σ | 40377.4 | 179.128 | 79.9 | | σ | 1006.1 | 5.569 | 35.4 |
| 9 | μ | 223409 | 1736.11 | 418.9 | 9 | μ | 155.4 | 16.752 | 10.4 |
| | σ | 9438.3 | 2273.81 | 307.9 | | σ | 310.9 | 8.286 | 5.5 |
| 11 | μ | 528015 | 5223.02 | 586.5 | 11 | μ | 141.6 | 93.402 | 31.2 |
| | σ | 8810 | 5694.68 | 738.9 | | σ | 116 | 92.053 | 24.9 |
| 13 | μ | 570925 | 18189 | 2507.9 | 13 | μ | 319.8 | 921.763 | 75.2 |
| | σ | 643217 | 22344.1 | 3345.6 | | σ | 213.9 | 677.844 | 49.4 |
| $d = 0.50$ | | | | | $d = 1.00$ | | | | |
| $m_1 = m_2 = m_3$ | | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> | $m_1 = m_2 = m_3$ | | <i>Nds</i> | <i>Temps</i> (sec) | <i>E</i> |
| 3 | μ | 5.8 | 0.043 | 2.4 | 3 | μ | 6.2 | 0.058 | 2.6 |
| | σ | 4 | 0.029 | 2.0 | | σ | 3 | 0.044 | 1.5 |
| 5 | μ | 18.8 | 0.225 | 2.9 | 5 | μ | 29 | 0.539 | 5.2 |
| | σ | 13.3 | 0.180 | 1.9 | | σ | 24.6 | 0.509 | 4.9 |
| 7 | μ | 44.6 | 4.510 | 9.4 | 7 | μ | 37.6 | 4.625 | 7.8 |
| | σ | 27.1 | 4.932 | 5.7 | | σ | 23.7 | 2.278 | 5.2 |
| 9 | μ | 115.2 | 46.908 | 26.4 | 9 | μ | 106.6 | 68.348 | 23.1 |
| | σ | 35.5 | 32.283 | 8.6 | | σ | 41.7 | 26.633 | 11.0 |
| 11 | μ | 232.4 | 330.787 | 54.2 | 11 | μ | 336.2 | 643.011 | 78.6 |
| | σ | 148.4 | 209.451 | 32.2 | | σ | 264.9 | 524.062 | 65.6 |
| 13 | μ | 468.2 | 1465.54 | 113.1 | 13 | μ | 572.6 | 3721.27 | 137.6 |
| | σ | 349 | 1109.62 | 87.8 | | σ | 388.1 | 2259.23 | 95.5 |

Le Tableau 4.8 donne une idée générale sur les temps de calculs de l'algorithme $E\chi MIP$, sur une machine SPARC station ULTRA 2 (Solaris 2.4-27), sur des jeux polymatriciels à trois joueurs générés aléatoirement, de taille et de densité différentes.

4.5.7 Description du logiciel *XGame Solver*

Dans le but de faire bénéficier la communauté scientifique de notre travail, nous avons décidé de développer une application à orientation utilisateur qui serait facile à utiliser et à comprendre. Les sous-sections suivantes décrivent les différentes interfaces et possibilités incorporées dans le logiciel *XGame Solver*.

4.5.8 Interface principale

En premier lieu, nous observons que l'utilisateur peut choisir de charger un jeu déjà existant dans la librairie d'exemples disponibles ou bien choisir de créer un nouveau jeu. L'icône de chargement d'un jeu existant permet de choisir un fichier d'extension ".xga" (XGame) qui contient le type de jeu ainsi que les informations concernant la taille du jeu et les différentes valeurs des gains des joueurs dans ce jeu. L'utilisateur peut éditer le contenu de ce fichier à sa guise ou en appliquant certaines transformations mathématiques pré-programmées. L'icône de création d'un nouveau jeu donne le choix à l'utilisateur de choisir le nom qu'il désire donner à son fichier d'extension ".xga", de choisir le type de jeu qu'il désire créer, de rentrer la taille du jeu et aussi les différents gains des joueurs. La Figure 4.5 est une image de l'interface principale du logiciel *XGame Solver*.

Nous observons aussi que l'utilisateur peut ouvrir plusieurs jeux du même type et/ou de types différents s'il le désire. De même, l'utilisateur peut lancer plusieurs énumérations des équilibres en même temps. Les jeux bimatriciels et séquentiels peuvent être résolus en faisant appel à l'algorithme *EEE* ou *E χ MIP*. L'utilisateur peut donc changer de méthode juste en sélectionnant celle désirée, avant de lancer l'énumération. Une icône représentant *un drapeau rouge* permet d'arrêter la résolution courante si le besoin se pose.

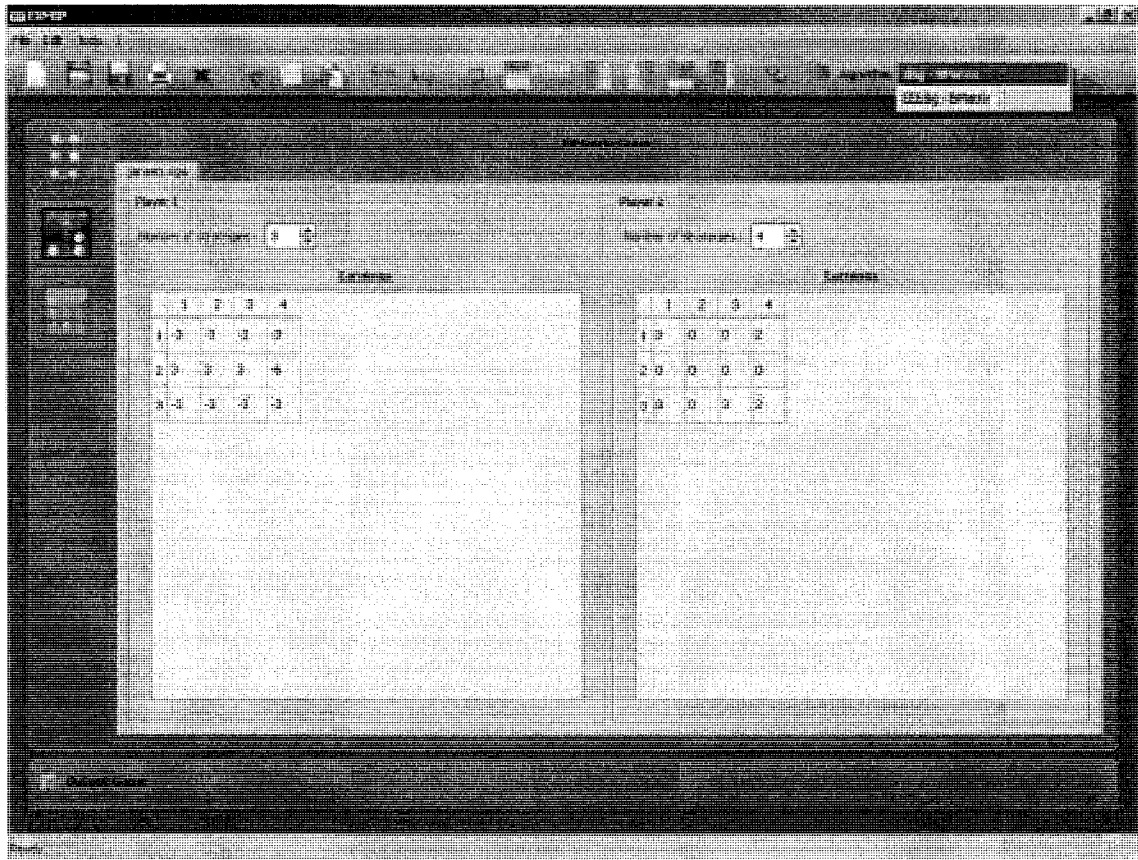


Figure 4.5 Interface principale de *XGame Solver*

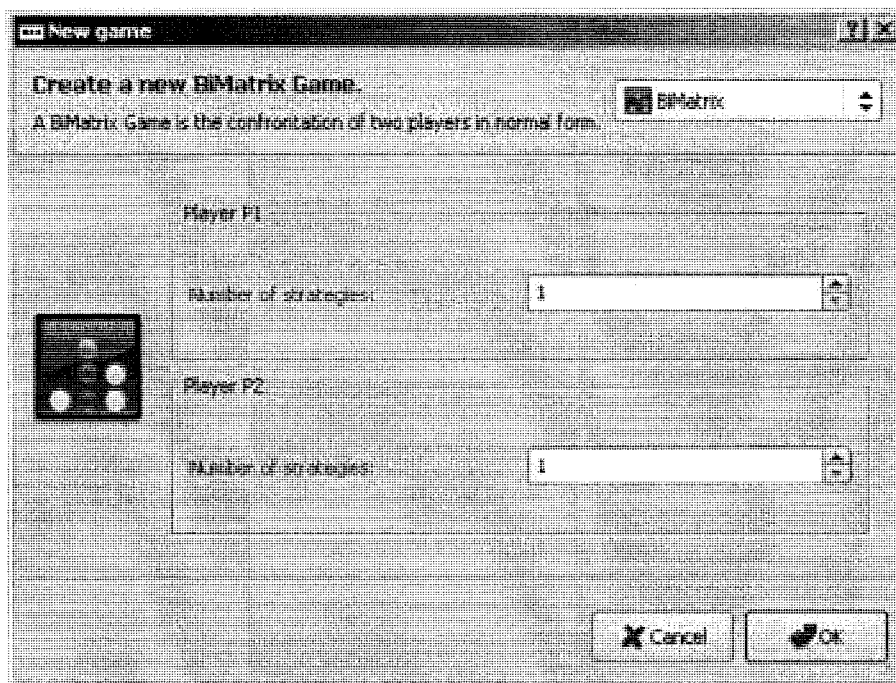


Figure 4.6 Interface Nouveau jeu bimatriciel

4.5.9 Créer un nouveau jeu

L'icône *Create new game* permet à l'utilisateur de créer un nouveau jeu en choisissant le nom qu'il désire donner à son fichier d'extension ".xga". Cette icône appelle une petite interface qui permet de choisir le type de jeu à créer et de lui donner la taille voulue. Les trois figures (Figure 4.6), (Figure 4.7) et (Figure 4.8) permettent d'avoir une idée sur l'interface correspondant à chaque type de jeu. Ces interfaces contiennent aussi une brève description de chaque type de jeu afin d'aider un utilisateur débutant à trouver la meilleure réponse à ses besoins.

Nous comptons aussi élargir la gamme des jeux pouvant être résolus par le logiciel *XGame Solver* en implantant leurs algorithmes de résolution appropriés. Nous croyons que ces ajouts seraient d'un grand intérêt pour la communauté scientifique.

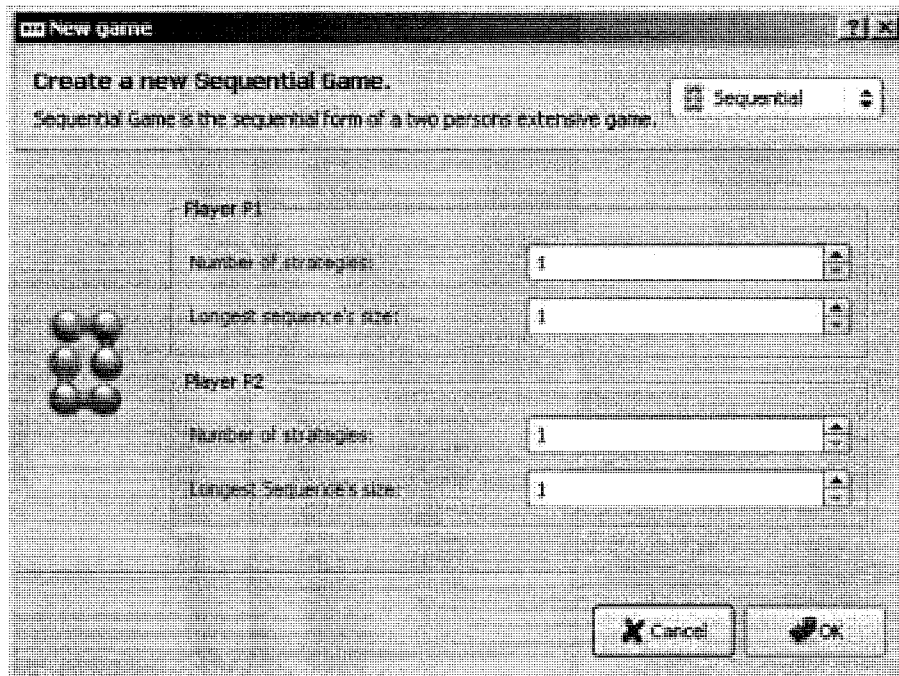


Figure 4.7 Interface Nouveau jeu séquentiel

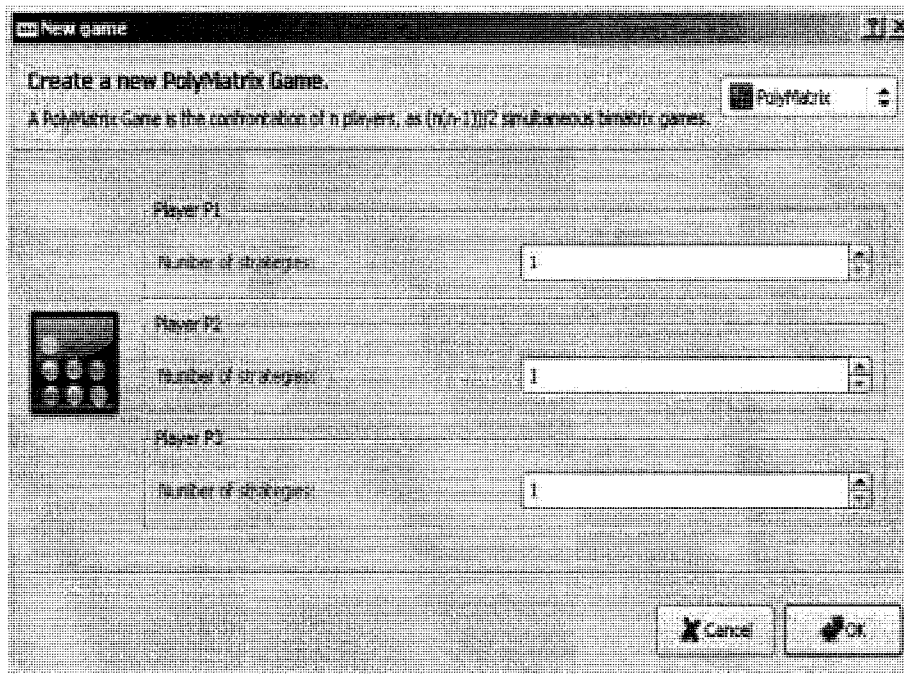


Figure 4.8 Interface Nouveau jeu polymatriciel

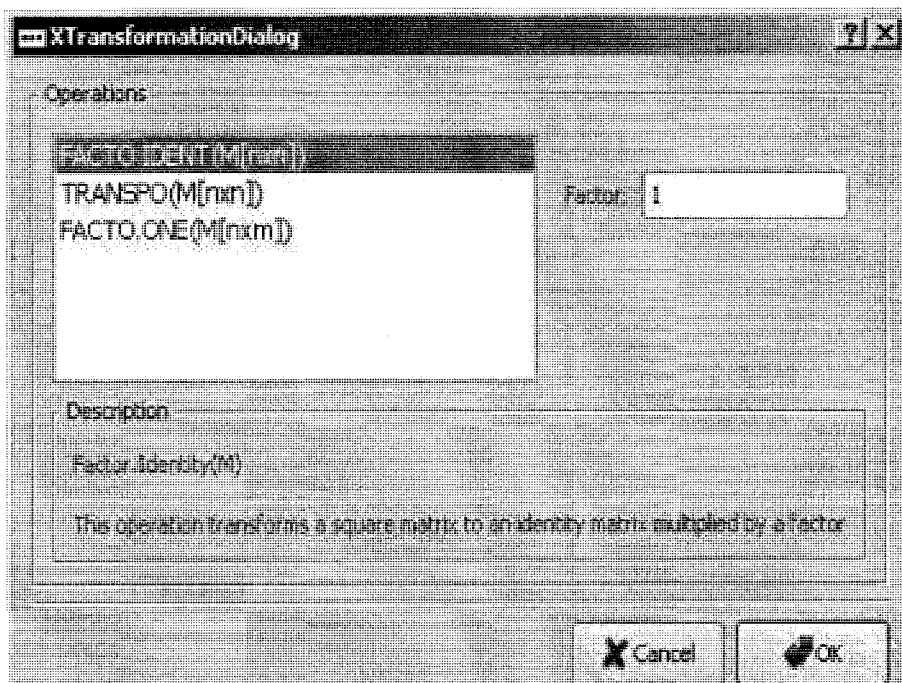


Figure 4.9 Interface de Transformation

4.5.10 Les icônes d'édition

En plus des icônes d'édition classiques : *Copy*, *Cut*, *Paste*, nous avons ajouté d'autres icônes au menu d'édition. Ainsi les icônes : *Add row above*, *Add row below*, *Add column left*, *Add column right*, *Remove rows*, *Remove columns* permettent d'ajouter ou d'éliminer des lignes aux matrices des gains. L'icône *Transform* appelle quant à elle une petite interface qui permet d'utiliser des fonctions de transformation mathématique sur une matrice. Les fonctions *Transpose*, *Identity*, *Factor*, permettent respectivement de transposer une matrice carrée, d'initialiser la matrice à un multiple de la matrice identité ou de rentrer la même valeur dans toutes les cellules. La Figure 4.9 montre l'interface de transformation.

4.5.11 La fenêtre de sortie

Les résultats des algorithmes d'énumération apparaissent dans la fenêtre de sortie située en bas de l'interface principale. L'utilisateur peut cacher ou montrer cette fenêtre ou bien l'élargir et la parcourir. L'utilisateur peut aussi choisir de sauvegarder les sorties provenant de l'énumération dans un fichier en utilisant l'icône de sauvegarde située juste au dessus de cette fenêtre. La Figure 4.10 montre la fenêtre de sortie.

4.6 Discussion

Nous avons commencé ce chapitre en présentant l'algorithme $E\chi MIP$ pour l'énumération des équilibres extrêmes en théorie des jeux. Nous avons aussi proposé une formulation biniveau permettant d'étendre l'algorithme EEE aux formes séquentielles des jeux étendus à deux joueurs. Nous avons pu implanter les algorithmes EEE et $E\chi MIP$, en langages C et $C++$, en faisant appel aux bibliothèques de $Cplex$ et à la bibliothèque d'arith-

```
EX-MIP
File Edit Tools
***** Dominance Check *****
n=3
m=3
***** Dominance Check *****
Equilibrium 1 (Node 13)
-----
X = 0 ; 0 ; 1 ;
Y = 0 ; 0 ; 1 ;
Alpha = 3 ; 3 ;
Beta = 0 ; 0 ;
Equilibrium 2 (Node 19)
-----
X = 0 ; 0 ; 1 ;
Y = 0 ; 1 ; 0 ;
Alpha = 3 ; 3 ;
Beta = 0 ; 0 ;
Equilibrium 3 (Node 35)
-----
X = 0 ; 0 ; 1 ;
Y = 1 ; 0 ; 0 ;
Alpha = 3 ; 3 ;
Beta = 0 ; 0 ;
Equilibrium 4 (Node 79)
-----
X = 0 ; 1 ; 0 ;
```

Figure 4.10 Fenêtre de sortie

métique exacte de McCutchen (44). Nous avons décrit dans ce chapitre les détails techniques de ces implantations et nous y avons exposé le logiciel *XGame Solver* qui regroupe ces implantations sous une application à interfaces facile à manipuler et mise gratuitement à la disposition de la communauté scientifique. La comparaison complète des performances de notre logiciel avec le logiciel *Gambit* (43) rassemblant les contributions de plusieurs chercheurs en théorie des jeux est pour le moment un peu prématurée. *Gambit* permet de trouver des équilibres pour des jeux étendus ou stratégiques à plusieurs joueurs. Il permet aussi d'énumérer les équilibres extrêmes pour les jeux bimatriels. *XGame Solver* possède cependant quelques avantages puisque *Gambit* ne permet pas de résoudre les jeux polymatriciels et les formes séquentielles des jeux étendus à deux joueurs.

CHAPITRE 5

RAFFINEMENT DES ÉQUILIBRES

Les différentes représentations d'un jeu étendu ont donné l'espoir de développer des outils d'analyse à travers les jeux sous forme normale. Cependant, il est facile de construire des jeux étendus différents mais qui auraient la même représentation sous forme stratégique et donc le même ensemble d'équilibres de Nash extrêmes. Ainsi, malgré ses divers avantages, le concept d'équilibre en théorie des jeux semble avoir plusieurs points faibles.

Le problème de sélection d'un équilibre parmi d'autres est l'une des plus importantes faiblesses de la notion d'équilibre. En effet, une théorie qui permet uniquement de prévoir que la solution d'un jeu non-coopératif est un équilibre sans spécifier exactement lequel, reste extrêmement faible.

Harsanyi et Selten (23) illustrent ce problème à l'aide d'un exemple simple. Dans cet exemple, deux joueurs se partagent un montant de 100 dollars en proposant chacun un nombre x_1 et x_2 , respectivement. Si $x_1 + x_2 \leq 100$ alors chaque joueur obtient un montant égal au nombre qu'il a proposé, x_1 ou x_2 . Sinon, les deux joueurs obtiennent un gain nul. Ainsi, si x_1 et x_2 sont des réels positifs ou nuls alors chaque paire (x_1, x_2) telle que $x_1 + x_2 = 100$ est un équilibre. Même si x_1 et x_2 ne prenaient que des valeurs entières positives il y aurait 101 équilibres différents. Il faudrait ensuite définir un critère de sélection afin de choisir un équilibre parmi toutes ces solutions possibles.

Dans ce chapitre, nous étudions plusieurs raffinements majeurs des équilibres en théorie des jeux et nous y apportons quelques contributions (2). Ainsi, dans les deux dernières sections de ce chapitre, nous considérons les raffinements parfait et propre du concept

d'équilibre de Nash. En quelque sorte, ces deux raffinements partent de l'idée qu'un équilibre raisonnable doit être stable advenant des perturbations des stratégies d'équilibre.

5.1 Équilibre essentiel

Dans cette section, nous allons considérer un raffinement basé sur l'idée qu'un équilibre doit être stable advenant des perturbations des gains du jeu. En effet, tout équilibre en stratégies mixtes peut s'avérer instable et ne pourrait pas dans ce cas être retenu comme la solution d'un jeu. Afin d'illustrer ce problème, Harsanyi et Selten (23) considèrent les jeux n'ayant que des équilibres en stratégies mixtes. Dans le jeu suivant, nous présentons les matrices des gains des deux joueurs sous la forme d'un tableau contenant les entrées des matrices A et B conjointement. Le seul équilibre qui existe est un équilibre en stratégies mixtes $E = (X_1, X_2)$, où $X_1 = (\frac{1}{3}, \frac{2}{3})$ et $X_2 = (\frac{4}{5}, \frac{1}{5})$.

Tableau 5.1 Jeu bimatriciel avec un seul équilibre en stratégies mixtes

| | | |
|---|--------|--------|
| | 1 | 2 |
| 1 | 45, 30 | 0, 90 |
| 2 | 30, 75 | 60, 45 |

Pour faciliter l'analyse de ce jeu, une ligne X_1 et une colonne X_2 sont ajoutées aux matrices des gains dans le Tableau 5.2. Ainsi, si le joueur 1 s'attend à ce que le joueur 2 joue sa stratégie mixte X_2 , alors il serait indifférent quant à jouer sa stratégie X_1 , car il aurait le même gain quelque soit son choix. Réciproquement, si le joueur 2 s'attend à ce que le joueur 1 joue sa stratégie mixte X_1 , alors il serait indifférent quant à jouer sa stratégie X_2 .

L'équilibre unique de ce jeu $E = (X_1, X_2)$ est instable, car même si aucun joueur n'a

Tableau 5.2 Ajout des stratégies d'un équilibre instable

| | | | |
|-------|--------|--------|--------|
| | 1 | 2 | X_2 |
| 1 | 45, 30 | 0, 90 | 36, 42 |
| 2 | 30, 75 | 60, 45 | 36, 69 |
| X_1 | 35, 60 | 40, 60 | 36, 60 |

un avantage à ne pas jouer sa stratégie X_1 ou X_2 , aucun des joueurs n'a un intérêt à le faire. Une petite perturbation des gains de chaque joueur donnerait donc un équilibre très différent de celui qui a été trouvé. Pour contrer ce genre de problèmes, Wu Wen-tsun et Jiang Jia-he (69) ont défini le concept d'équilibre *essentiel*.

Définition 5.1.1 *Un équilibre (x_1, x_2) est essentiel pour un jeu bimatriciel $G(A, B)$ si il existe pour chaque voisinage N_x de (x_1, x_2) un voisinage N_G de (A, B) tel que $G(A', B')$ a au moins un équilibre dans N_x pour tout $(A', B') \in N_G$.*

Il a été démontré par van Damme (59) que tout équilibre essentiel est parfait. Ce qui veut dire que la stabilité advenant des perturbations des gains implique la stabilité advenant des perturbations des stratégies. En plus, Kojima *et al.* (32) ont renforcé le concept de stabilité d'un équilibre en définissant le concept d'équilibre *fortement stable*. Un tel équilibre varie d'une façon unique et continue advenant des perturbations des gains du jeu.

5.2 Équilibre régulier

Le concept d'équilibre *régulier* a été défini en premier lieu par Harsanyi (22) tel que le jacobien d'une certaine application associée au jeu et évalué au point d'équilibre est non-singulier. Cette définition a été spécialisée par van Damme (59) pour le cas d'un jeu à deux joueurs. Il a montré qu'un équilibre est *régulier* si et seulement si il est *quasi-fort*

et *isolé*. Il a aussi montré qu'un équilibre régulier est aussi propre et fortement stable. Dans cette section, nous allons donner les définitions des raffinements *quasi-fort* et *isolé*.

5.2.1 Équilibre quasi-fort

Pour un équilibre (x_1, x_2) d'un jeu bimatriciel $G(A, B)$, soit $M(A, x_2)$ l'ensemble des pures meilleures répliques du joueur I contre le choix stratégique x_2 du joueur II :

$$M(A, x_2) = \{i \in \mathbb{N}^n : e_i A x_2 = \max_{k \in \mathbb{N}_n} e_k A x_2\}, \quad (5.2.1)$$

et similairement, soit :

$$M(x_1, B) = \{j \in \mathbb{N}^m : x_1 B e_j = \max_{k \in \mathbb{N}_m} x_1 B e_k\}, \quad (5.2.2)$$

l'ensemble des pures meilleures répliques du joueur II contre le choix stratégique x_1 du joueur I.

Définition 5.2.1 Soit $C(x_1)$ le vecteur porteur de x_1 . $C(x_1)$ est l'ensemble $\{i \in \mathbb{N}^n : x_{1i} > 0\}$ et le vecteur porteur de x_2 , $C(x_2)$, est l'ensemble $\{j \in \mathbb{N}^m : x_{2j} > 0\}$. Selon Harsanyi (22), un équilibre (x_1, x_2) est *quasi-fort* si :

$$C(x_1) = M(A, x_2) \text{ et } C(x_2) = M(x_1, B).$$

Exemple 5.2.1 Soit A et B les matrices des gains d'un jeu bimatriciel de dimension (4×2) énoncé par Myerson (49) :

$$A = \begin{pmatrix} 4 & 4 \\ 4 & 4 \\ 6 & 3 \\ 0 & 2 \end{pmatrix} \quad B = \begin{pmatrix} 4 & 4 \\ 4 & 4 \\ 6 & 0 \\ 0 & 2 \end{pmatrix}.$$

Nous avons utilisé les algorithmes $E\chi MIP$ (4; ?) et EEE (6) dans le but d'énumérer tous les équilibres extrêmes de ce jeu. Les deux algorithmes énumèrent cinq équilibres de Nash extrêmes (Tableau 5.3).

Tableau 5.3 Équilibres de Nash extrêmes pour Myerson (49)

| Eq. | x_1 | | | | x_2 | | α_1 | α_2 |
|-----|-------|---|---|---|-------|-----|------------|------------|
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 6 | 6 |
| 2 | 0 | 1 | 0 | 0 | 0 | 1 | 4 | 4 |
| 3 | 0 | 1 | 0 | 0 | 1/3 | 2/3 | 4 | 4 |
| 4 | 1 | 0 | 0 | 0 | 0 | 1 | 4 | 4 |
| 5 | 1 | 0 | 0 | 0 | 1/3 | 2/3 | 4 | 4 |

Ce jeu possède deux sous-ensembles de Nash maximaux $T_1 = \{1\}$ et $T_2 = \{2, 3, 4, 5\}$. Pour le premier équilibre de Nash extrême de ce jeu : $x_1 = (0, 1, 0, 0)$ et $x_2 = (0, 1)$. Donc $C(x_1) = \{3\}$ et $C(x_2) = \{1\}$. Nous avons $M(A, x_2) = \{3\}$ et $M(x_1, B) = \{1\}$. Cet équilibre de Nash extrême est donc quasi-fort. Les quatre autres équilibres de Nash extrêmes de ce jeu ne sont pas quasi-forts.

5.2.2 Équilibre isolé

Jansen (26) a porté une attention particulière aux équilibres qui sont en même temps quasi-forts et isolés. Ces équilibres sont des équilibres *essentiels*.

Définition 5.2.2 Un équilibre de Nash (x_1, x_2) d'un jeu bimatriciel $G(A, B)$ est isolé si

il existe un voisinage N_x de (x_1, x_2) tel qu'il soit le seul équilibre de $G(A, B)$ dans ce voisinage N_x .

En d'autres mots, tout équilibre de Nash isolé est un équilibre de Nash extrême définissant à lui seul un sous-ensemble de Nash maximal et isolé. Ainsi, l'énumération des sous-ensemble maximaux de Nash peut être aussi utilisée afin de déterminer automatiquement les équilibres de Nash isolés. Par ailleurs, Jansen (26) a aussi proposé les définitions suivantes :

Définition 5.2.3 Soit (x_1, x_2) un équilibre de Nash quasi-fort d'un jeu bimatriciel $G(A, B)$ avec $A, B > 0$. L'équilibre (x_1, x_2) est isolé si et seulement si $|C(x_1)| = |C(x_2)|$ et les matrices $[a_{ij}]_{i \in C(x_1), j \in C(x_2)}$ et $[b_{ij}]_{i \in C(x_1), j \in C(x_2)}$ sont inversibles.

Même si cette définition s'applique seulement aux jeux bimatriels $G(A, B)$ tels que $A, B > 0$, il est notoirement connu qu'il est facile de transformer les matrices des gains A et B de tout jeu bimatriciel pour que A et B aient des éléments strictement positifs et ce sans changer les sous-ensembles Nash maximaux. Par exemple, ceci peut être réalisé en ajoutant $1 + |a_{min}|$, avec $a_{min} = \min a_{ij}$, à chaque élément de A et en ajoutant $1 + |b_{min}|$, avec $b_{min} = \min b_{ij}$, à chaque élément de B .

Jansen (26) mentionne qu'un équilibre isolé est essentiel si et seulement si il est quasi-fort. Par ailleurs, van Damme (59) a montré qu'un équilibre isolé et quasi-fort est parfait et propre. Ceci a été aussi montré par Okada (52) pour les jeux bimatriels.

Jansen (27) a aussi montré qu'un équilibre d'un jeu bimatriciel est dit régulier si et seulement si il est isolé et quasi-fort. Il a aussi souligné l'équivalence entre les équilibres fortement stables et les équilibres réguliers.

Exemple 5.2.2 Le premier équilibre de Nash extrême du jeu bimatriciel (exemple 5.2.1)

de dimension (4×2) de Myerson (49), est quasi-fort et isolé. Cet équilibre est donc essentiel, fortement stable, régulier, parfait et propre.

5.3 Équilibre parfait

La distinction entre équilibre *parfait* et équilibre imparfait dans le cas des jeux sous forme normale est souvent difficile. Afin d'illustrer le problème d'imperfection, Harsanyi et Selten (23) donnent l'exemple du jeu étendu illustré par la Figure 5.1.

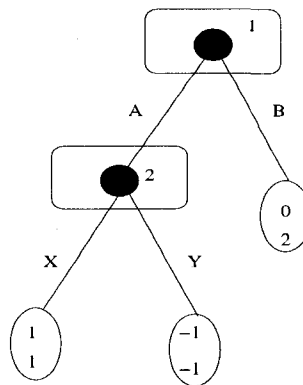


Figure 5.1 Jeu étendu d'Harsanyi et Selten (23)

Dans ce jeu, le joueur 1 fait le premier mouvement et a le choix entre choisir la stratégie A ou la stratégie B . Si le joueur 1 choisit sa stratégie B , le jeu se termine et les gains des joueurs seraient $\alpha_1 = 0$ et $\alpha_2 = 2$. Si le joueur 1 choisit sa stratégie A , le joueur 2 aura la possibilité de faire un mouvement en choisissant la stratégie X ou la stratégie Y . Si le joueur 2 choisit la stratégie X , les gains des joueurs seraient $\alpha_1 = 1$ et $\alpha_2 = 1$. Si il choisit la stratégie Y , les gains des joueurs seraient $\alpha_1 = \alpha_2 = -1$. Ainsi, le joueur 1 a deux stratégies pures, A et B , et le joueur 2 a deux stratégies pures X et Y . La forme stratégique réduite de ce jeu peut être représentée par le Tableau 5.4. Les matrices des

Tableau 5.4 Forme stratégique réduite (23)

| | X | Y |
|---|------|--------|
| A | 1, 1 | -1, -1 |
| B | 0, 2 | 0, 2 |

gains pour les deux joueurs sont donc respectivement :

$$\begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} 1 & -1 \\ 2 & 2 \end{pmatrix}.$$

En appliquant l'algorithme *EEE* (6) à cette forme stratégique réduite, les trois équilibres de Nash extrêmes suivants sont énumérés

$$E_1 : \begin{bmatrix} \alpha_1 = 0 & \alpha_2 = 2 \\ x_1 = [0, 1] & x_2 = [0, 1] \end{bmatrix} \quad E_2 : \begin{bmatrix} \alpha_1 = 1 & \alpha_2 = 1 \\ x_1 = [1, 0] & x_2 = [1, 0] \end{bmatrix}$$

$$E_3 : \begin{bmatrix} \alpha_1 = 0 & \alpha_2 = 2 \\ x_1 = [0, 1] & x_2 = [\frac{1}{2}, \frac{1}{2}] \end{bmatrix}$$

Le jeu a donc deux équilibres extrêmes en stratégies pures : $E_1 = (B, Y)$ et $E_2 = (A, X)$.

Cependant, E_2 est un équilibre parfait alors que E_1 est un équilibre imparfait puisqu'il fait appel à des stratégies irrationnelles. Ainsi, en choisissant Y , le joueur 2 fait un choix irrationnel puisque les deux joueurs auraient le même gain $\alpha_1 = \alpha_2 = -1$, alors qu'en choisissant X , il aurait obtenu avec le joueur 1 le même gain $\alpha_1 = \alpha_2 = 1$. En outre, le choix de la stratégie B par le joueur 1 est aussi irrationnel, puisqu'il sait qu'en choisissant A le joueur 2 choisirait certainement X . Ceci lui procurerait un gain $\alpha_1 = 1$ alors qu'en choisissant B , il aurait un gain $\alpha_1 = 0$.

En réalité, si les joueurs choisissaient d'agir comme dans E_1 , alors le joueur 2 n'aurait jamais à faire ce choix irrationnel Y . La probabilité de réalisation du mouvement Y est donc nulle. Ce mouvement ne peut donc altérer le gain espéré par le joueur 2 puisqu'il a une probabilité de réalisation égale à 0.

En effet, une stratégie d'équilibre maximise le gain espéré par le joueur concerné étant donné les choix des autres joueurs. Ainsi, un équilibre ne peut contenir un mouvement irrationnel à un ensemble informationnel qui serait atteint avec une probabilité positive, si tous les joueurs appliquaient leurs stratégies d'équilibre.

Cependant, un équilibre pourrait contenir un mouvement irrationnel pour un joueur à un ensemble informationnel qui serait atteint avec une probabilité nulle. Un équilibre imparfait est précisément un équilibre qui contient un mouvement irrationnel à un ensemble informationnel qui serait atteint avec une probabilité nulle.

Selten (57) a défini un équilibre parfait d'un jeu étendu comme étant tout équilibre parfait de sa représentation multiagent. Myerson (49) donne la définition mathématique suivante d'un équilibre parfait pour un jeu sous forme stratégique.

Définition 5.3.1 Soit $\Gamma = (N, (C_i)_{i \in N}, (u_i)_{i \in N})$ un jeu fini sous forme stratégique. Avec N : l'ensemble des joueurs, C_i : l'ensemble des mouvements possibles (stratégies pures) du joueur i et u_i : l'utilité ou le gain du joueur i selon ses choix.

Soit $\Delta(C_i)$ l'ensemble des distributions de probabilité du joueur i sur l'ensemble de ses mouvements possibles. Soit $\Delta^0(C_i)$ le sous-ensemble de $\Delta(C_i)$ tel que chaque stratégie pure est choisie avec une probabilité strictement positive.

Une séquence de mouvements $\sigma \in \prod_{i \in N} \Delta(C_i)$ est un équilibre parfait de Γ si et seulement si il existe une série $(\hat{\sigma}^k)_{k=1}^{\infty}$, telle que :

$$\begin{aligned} \hat{\sigma}^k &\in \prod_{i \in N} \Delta^0(C_i), & \forall k \in \{1, 2, 3, \dots\}, \\ \lim_{k \rightarrow \infty} \hat{\sigma}^k &= \sigma_i(c_i), & \forall i \in N, \text{ et } \forall c_i \in C_i, \\ \sigma_i &\in \operatorname{argmax}_{\tau \in \Delta(C_i)} u_i(\hat{\sigma}_{-i}^k, \tau_i), & \forall i \in N. \end{aligned}$$

où les notations utilisées sont : $\sigma = (\sigma_i)_{i \in N}$, $\hat{\sigma}^k = (\hat{\sigma}_i^k)_{i \in N}$ et $\hat{\sigma}_{-i}^k = (\hat{\sigma}_j^k)_{j \in N-i}$.

Ces conditions garantissent que chaque stratégie pure de chaque joueur est choisie avec une probabilité strictement positive, que $\hat{\sigma}^k$ est une distribution de probabilité très proche de σ et que le choix stratégique de chaque joueur i est la meilleure réplique aux choix stratégiques des autres joueurs.

Il est facile de démontrer grâce à la dernière condition, que chaque équilibre parfait est un équilibre de Nash pour le jeu Γ . Encore mieux, Selten (57) montre qu'à chaque équilibre parfait de la représentation multiagent d'un jeu étendu correspond un équilibre séquentiel du jeu étendu.

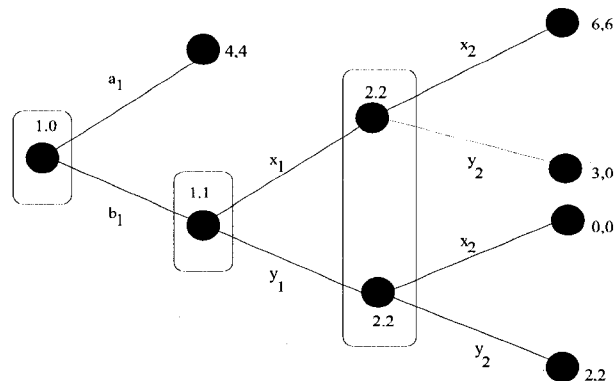


Figure 5.2 Jeu étendu à deux joueurs de Myerson (49)

Afin de mieux illustrer cette définition, considérons le jeu tiré de Myerson (49) et représenté par la Figure 5.2. La séquence de mouvements $\sigma = ([a_1 x_1], [y_2])$ est un équilibre

parfait de la représentation sous forme normale de ce jeu. Cependant, cette séquence ne définit pas un équilibre séquentiel pour le jeu étendu.

Dans le but de montrer que σ est un équilibre parfait de la représentation sous forme normale de ce jeu, soit la séquence de mouvements perturbée :

$$\hat{\sigma}^k = ((1 - \epsilon) [a_1x_1] + 0.1\epsilon [a_1y_1] + 0.1\epsilon [b_1x_1] + 0.8\epsilon [b_1y_1], \epsilon [x_2] + (1 - \epsilon) [y_2]),$$

où $\epsilon = 1/(k + 2)$. Si $\epsilon \leq 1/3$, alors pour le joueur 1, $[a_1x_1]$ est une meilleure réponse à $\epsilon [x_2] + (1 - \epsilon) [y_2]$, car $4 \geq 6\epsilon + 3(1 - \epsilon)$. D'un autre côté, quand la probabilité de b_1y_1 est positive et plus de trois fois supérieure à la probabilité de b_1x_1 , alors y_2 est une meilleure réponse pour le joueur 2. La suite $\{\hat{\sigma}^k\}_{k=1}^{\infty}$ satisfait donc la définition (5.3.1) et $\sigma = ([a_1x_1], [y_2])$ est un équilibre parfait de la représentation stratégique.

Un équilibre de Nash de la représentation multiagent peut ne pas être parfait et ne pas représenter un équilibre séquentiel du jeu étendu. Par ailleurs, un équilibre de Nash de la représentation stratégique (non multiagent) peut être parfait mais ne pas représenter un équilibre séquentiel du jeu étendu (Myerson (49)).

Ainsi, lorsque un jeu a plusieurs solutions d'équilibres possibles, les décideurs doivent raffiner leurs choix en utilisant d'autres concepts rationnels en plus du concept d'équilibre de Nash en théorie des jeux. La définition suivante d'un équilibre *parfait* résume les définitions de Selten (57), van Damme (59) et Borm *et al.* (14).

Définition 5.3.2 Soit (\hat{x}_1, \hat{x}_2) un équilibre de Nash pour un jeu bimatriciel. Si il existe un vecteur de probabilité x_1 tel que $x_1A \geq \hat{x}_1A$ et $x_1A \neq \hat{x}_1A$, ou si il existe un vecteur de probabilité x_2 tel que $Bx_2 \geq B\hat{x}_2$ et $Bx_2 \neq B\hat{x}_2$, alors (\hat{x}_1, \hat{x}_2) est imparfait. Autrement, (\hat{x}_1, \hat{x}_2) est un équilibre parfait.

Selten (57) montre que chaque jeu fini sous forme stratégique possède au moins un

équilibre parfait et donc chaque jeu étendu possède au moins un équilibre séquentiel.

Proposition 5.3.1 *L'équilibre (\hat{x}_1, \hat{x}_2) est parfait si et seulement si les objectifs optimaux des programmes linéaires suivants sont égaux à zéro :*

$$\begin{array}{ll}
 \max_{(x_1, \epsilon_1)} & \mathbf{I}^t \epsilon_1 \\
 \text{s.à.} & \mathbf{I}x_1 = 1, \\
 & x_1 A \geq \hat{x}_1 A + \epsilon_1, \\
 & x_1, \epsilon_1 \geq 0, \\
 \max_{(x_2, \epsilon_2)} & \mathbf{I}^t \epsilon_2 \\
 \text{s.à.} & \mathbf{I}x_2 = 1, \\
 & Bx_2 \geq B\hat{x}_2 + \epsilon_2, \\
 & x_2, \epsilon_2 \geq 0.
 \end{array} \tag{5.3.3}$$

Preuve 5.3.1 *Soit (x_1^*, ϵ_1^*) et (x_2^*, ϵ_2^*) des solutions optimales pour ces deux programmes (5.3.3). Si un des deux objectifs des solutions optimales a une valeur strictement positive, alors au moins une des variables ϵ est strictement positive. Ceci signifie qu'il existe au moins une variable $\epsilon_1^i > 0, i \in \{1, 2, \dots, n\}$, ou $\epsilon_2^j > 0, j \in \{1, 2, \dots, m\}$, telle que $x_1^* A_i \geq \hat{x}_1 A_i + \epsilon_1^i$, ou $B_j x_2^* \geq B_j \hat{x}_2 + \epsilon_2^j$. Ainsi, ou bien nous avons $x_1^* A_i > \hat{x}_1 A_i$, ou $B_j x_2^* > B_j \hat{x}_2$. Ce qui veut dire que $x_1 A \neq \hat{x}_1 A$, ou $Bx_2 \neq B\hat{x}_2$, alors que $x_1^* A \geq \hat{x}_1 A + \epsilon$ et $Bx_2^* \geq B\hat{x}_2 + \epsilon$ sont satisfaits. Ainsi, l'équilibre (\hat{x}_1, \hat{x}_2) n'est pas parfait.*

Si les deux objectifs optimaux sont égaux à zéro, alors tous les éléments des vecteurs ϵ_1^ et ϵ_2^* sont égaux à zéro. Ces vecteurs ϵ_1^* et ϵ_2^* correspondent aux valeurs maximales des écarts entre $x_1^* A$ et $\hat{x}_1 A$, et Bx_2^* et $B\hat{x}_2$. Ainsi, $x_1^* A = \hat{x}_1 A$ et $Bx_2^* = B\hat{x}_2$. Si toutes les variables ϵ sont égales à zéro, alors l'équilibre (\hat{x}_1, \hat{x}_2) est parfait.*

Les deux programmes linéaires (5.3.3) sont toujours réalisables ($\epsilon_1 = 0$ et $x_1 = \hat{x}_1$ ou $\epsilon_2 = 0$ et $x_2 = \hat{x}_2$). Les mêmes bibliothèques d'arithmétique exacte (3) peuvent être utilisées dans le but d'obtenir des solutions exactes pour ces deux programmes linéaires.

Exemple 5.3.1 *Dans l'exemple de jeu bimatriciel de dimension (4×2) (exemple 5.2.1) énoncé par Myerson (49), pour le premier équilibre extrême les deux programmes li-*

néaires, permettant de déterminer si il s'agit d'un équilibre parfait, s'expriment comme suit :

$$\begin{array}{ll}
 \max_{x_1, \epsilon_1} & \epsilon_{11} + \epsilon_{12} \\
 \text{s.à.} & \\
 x_{11} + x_{12} + x_{13} + x_{14} & = 1, \\
 4x_{11} + 4x_{12} + 6x_{13} + 0x_{14} & \geq 6 + \epsilon_{11}, \\
 4x_{11} + 4x_{12} + 3x_{13} + 2x_{14} & \geq 3 + \epsilon_{12}, \\
 x_1, \epsilon_1 & \geq 0,
 \end{array}
 \qquad
 \begin{array}{ll}
 \max_{x_2, \epsilon_2} & \epsilon_{21} + \epsilon_{22} + \epsilon_{23} + \epsilon_{24} \\
 \text{s.à.} & \\
 x_{21} + x_{22} & = 1, \\
 4x_{21} + 4x_{22} & \geq 4 + \epsilon_{21}, \\
 4x_{21} + 4x_{22} & \geq 4 + \epsilon_{22}, \\
 6x_{21} + 0x_{22} & \geq 6 + \epsilon_{23}, \\
 0x_{21} + 2x_{22} & \geq 0 + \epsilon_{24}, \\
 x_2, \epsilon_2 & \geq 0.
 \end{array}$$

Les cinq équilibres extrêmes obtenus sont parfaits. Leurs paires de programmes linéaires ont des objectifs de valeurs optimales égales à zéro. Les stratégies mixtes \hat{x}_1 ou \hat{x}_2 ne sont pas dominées pour tous ces équilibres de Nash extrêmes.

Un *sous-ensemble de Selten maximal* est un ensemble d'équilibres parfaits qui appartiennent au même sous-ensemble de Nash maximal, comme décrit par Borm *et al.* (14). Chaque point extrême d'un sous-ensemble de Selten maximal correspond à un équilibre extrême parfait.

Exemple 5.3.2 *Dans le jeu bimatriciel énoncé par Borm et al. (14) et présenté dans l'exemple (1.4.1), pour le deuxième équilibre de Nash extrême les deux programmes linéaires permettant de déterminer si il s'agit d'un équilibre parfait, s'expriment comme suit :*

$$\begin{array}{ll}
\max_{x_1, \epsilon_1} & \epsilon_{11} + \epsilon_{12} + \epsilon_{13} + \epsilon_{14} \\
s.\dot{a}. & \\
x_{11} + x_{12} + x_{13} & = 1, \\
-3x_{11} + 3x_{12} - 3x_{13} & \geq 3 + \epsilon_{11}, \\
-3x_{11} + 3x_{12} - 3x_{13} & \geq 3 + \epsilon_{12}, \\
-3x_{11} + 3x_{12} - 3x_{13} & \geq 3 + \epsilon_{13}, \\
-3x_{11} - 6x_{12} - 3x_{13} & \geq -6 + \epsilon_{14}, \\
x_1, \epsilon_1 \geq 0, &
\end{array}
\qquad
\begin{array}{ll}
\max_{x_2, \epsilon_2} & \epsilon_{21} + \epsilon_{23} \\
s.\dot{a}. & \\
x_{21} + x_{22} + x_{23} + x_{24} & = 1, \\
3x_{21} + 2x_{24} & \geq 0 + \epsilon_{21}, \\
3x_{23} + 2x_{24} & \geq 0 + \epsilon_{23}, \\
x_2, \epsilon_2 \geq 0. &
\end{array}$$

Pour cet équilibre de Nash extrême, le vecteur de stratégies mixtes $\hat{x}_1 = (0, 1, 0)$ n'est pas dominé, alors que $\hat{x}_2 = (0, 1, 0, 0)$ est dominé par $x_2 = (0, 0, 0, 1)$, avec $\epsilon_{21} = 2$ et $\epsilon_{23} = 2$. Ainsi, le deuxième équilibre de Nash extrême de ce jeu n'est pas parfait.

Parmi les dix équilibres extrêmes de Nash énumérés, huit équilibres sont parfaits. Ces équilibres de Nash extrêmes parfaits sont les équilibres 1, 3, 4, 6, 7, 8, 9 et 10. Ces équilibres de Nash extrêmes sont les points extrêmes des sous-ensembles de Selten maximaux $S_1 = \{1, 3, 4, 6\}$, $S_2 = \{4, 8\}$, $S_3 = \{6, 10\}$, $S_4 = \{7, 9\}$, $S_5 = \{7, 10\}$ et $S_6 = \{8, 9\}$.

Cette définition alternative que nous proposons d'un équilibre parfait permet donc de définir une nouvelle méthode d'énumérer tous les équilibres de Nash extrêmes et de déterminer les équilibres extrêmes parfaits.

5.4 Équilibre propre

Le concept d'équilibre parfait s'applique à la représentation multiagent d'un jeu étendu et permet donc de distinguer entre deux jeux étendus qui auraient la même représentation

stratégique. Ainsi, un équilibre parfait pour la représentation multiagent de l'un de ces deux jeux pourrait ne pas être parfait pour la représentation multiagent de l'autre.

Pour éviter de distinguer entre deux jeux étendus différents, tout en explorant leurs ensembles d'équilibres séquentiels, un raffinement qui peut toujours définir un équilibre séquentiel devait être défini. En effet, les *équilibres propres* forment un sous-ensemble des équilibres parfaits et permettent d'identifier les équilibres séquentiels d'un jeu étendu, même à partir de sa représentation stratégique.

L'idée principale derrière le raffinement propre d'un équilibre est qu'un joueur qui se trompe, essaierait encore plus fort d'éviter les erreurs les plus coûteuses que les erreurs les moins coûteuses. Il y aurait donc une sorte de rationalité dans le mécanisme de faire des erreurs. Certaines définitions doivent être énoncées au préalable.

Définition 5.4.1 Soit $\Gamma = (N, (C_i)_{i \in N}, (u_i)_{i \in N})$ un jeu stratégique fini. Soit ϵ un réel positif quelconque. Une séquence de mouvements σ est un équilibre ϵ -parfait si et seulement si : $\sigma \in \prod_{i \in N} \Delta^0(C_i)$, et pour tout joueur i et toute stratégie pure $c_i \in C_i$, si $\sigma_i \notin \underset{e_i \in C_i}{\operatorname{argmax}} u_i(\sigma_{-i}, [e_i])$, alors $\sigma_i(c_i) < \epsilon$.

Un équilibre ϵ -parfait est tel que toutes les stratégies pures de tous les joueurs sont choisies avec des probabilités strictement positives. En plus, si pour un joueur i , une stratégie pure c_i n'est pas la meilleure réplique par rapport aux choix des autres joueurs, alors la probabilité que ce joueur i se trompe de choix en lui accordant une probabilité positive est inférieure à ϵ .

Cette définition d'un équilibre ϵ -parfait permet d'introduire le concept d'équilibre ϵ -propre.

Définition 5.4.2 Soit $\Gamma = (N, (C_i)_{i \in N}, (u_i)_{i \in N})$ un jeu sous forme stratégique fini. Soit ϵ un réel positif quelconque. Une séquence de mouvements σ est un équilibre ϵ -propre si

et seulement si : $\sigma \in \prod_{i \in N} \Delta^0(C_i)$, et pour tout joueur i et pour toute paire de stratégies pures c_i et e_i , de C_i , si $u_i(\sigma_{-i}, [c_i]) < u_i(\sigma_{-i}, [e_i])$, alors $\sigma_i(c_i) \leq \epsilon \sigma_i(e_i)$.

Un équilibre ϵ -propre est donc aussi un équilibre ϵ -parfait. En plus, si pour un joueur i , une stratégie pure e_i est une meilleure réplique que la stratégie pure c_i , par rapport aux choix des autres joueurs, alors la probabilité que ce joueur i se trompe de choix en accordant une probabilité positive à c_i est ϵ fois inférieure à sa probabilité de choisir e_i .

Définition 5.4.3 Un équilibre (x_1^k, x_2^k) d'un jeu bimatriciel est ϵ_k -propre, pour un réel $\epsilon_k > 0$, si les conditions suivantes sont satisfaites :

$$\text{si } A_i x_2^k < A_h x_2^k, \text{ alors } x_{1i}^k \leq \epsilon_k x_{1h}^k, \quad \forall i, h \in \{1, 2, \dots, n\}, \quad (5.4.4)$$

$$\text{si } x_1^k B_j < x_1^k B_l, \text{ alors } x_{2j}^k \leq \epsilon_k x_{2l}^k, \quad \forall j, l \in \{1, 2, \dots, m\}, \quad (5.4.5)$$

$$x_{1i}^k > 0, \quad \forall i \in \{1, 2, \dots, n\}, \quad x_{2j}^k > 0, \quad \forall j \in \{1, 2, \dots, m\}. \quad (5.4.6)$$

Dans ce qui suit, nous allons essayer de nous procurer un outil pratique permettant d'identifier les équilibres ϵ -propres et les équilibres non-propres. Nous introduisons l'ensemble Ω_ϵ^σ , pour tout $0 \leq \epsilon < 1$ et $0 \leq \sigma < 1$, tel que :

$$\begin{aligned}
\Omega_\epsilon^\sigma = \{ (x_1, x_2) : \exists u, v \text{ tel que} & \quad \mathbf{1}x_1 = 1, \mathbf{1}x_2 = 1, \\
& \sigma \leq x_{1i}, \quad \forall i \in \{1, 2, \dots, n\}, \\
& \sigma \leq x_{2j}, \quad \forall j \in \{1, 2, \dots, m\}, \\
& A_h x_2 \leq A_i x_2 + L u_{ih}, \quad \forall i, h \in \{1, 2, \dots, n\}, i \neq h, \\
& x_{1i} + u_{ih} \leq \epsilon x_{1h} + 1, \quad \forall i, h \in \{1, 2, \dots, n\}, i \neq h, \\
& u_{ih} + u_{hi} \leq 1, \quad \forall i, h \in \{1, 2, \dots, n\}, i < h, \\
& u_{ih} \in \{0, 1\}, \quad \forall i, h \in \{1, 2, \dots, n\}, i \neq h, \\
& x_1 B_i \leq x_1 B_j + L v_{jl}, \quad \forall j, l \in \{1, 2, \dots, m\}, j \neq l, \\
& x_{2j} + v_{jl} \leq \epsilon x_{2l} + 1, \quad \forall j, l \in \{1, 2, \dots, m\}, j \neq l, \\
& v_{jl} + v_{lj} \leq 1, \quad \forall j, l \in \{1, 2, \dots, m\}, j < l, \\
& v_{jl} \in \{0, 1\}, \quad \forall j, l \in \{1, 2, \dots, m\}, j \neq l \quad \}.
\end{aligned}$$

Tout élément de Ω_ϵ^σ est un équilibre ϵ -propre, lorsque $\epsilon > 0$.

Proposition 5.4.1 *Si $(x_1, x_2) \in \Omega_\epsilon^\sigma$ pour des réels $0 < \epsilon < 1$ et $0 < \sigma < 1$ alors (x_1, x_2) est un équilibre ϵ -propre.*

Preuve 5.4.1 *Supposons que (x_1, x_2) est un élément de Ω_ϵ^σ , pour des réels $0 < \epsilon < 1$ et $0 < \sigma < 1$. On pose i et h des indices dans $\{1, 2, \dots, n\}$ tels que $i \neq h$. L'inégalité $u_{ih} + u_{hi} \leq 1$ assure que la combinaison $u_{ih} = 1$ et $u_{hi} = 1$ est impossible. En plus,*

- si $u_{ih} = 0$ et $u_{hi} = 0$ alors $A_h x_2 = A_i x_2$.
- si $u_{ih} = 1$, alors $\epsilon x_{1i} \leq x_{1i} \leq \epsilon x_{1h} \leq x_{1h}$ implique $x_{1h} \geq \epsilon x_{1i}$ et $u_{hi} = 0$, donc $A_i x_2 \leq A_h x_2$,
- si $u_{hi} = 1$, alors $\epsilon x_{1h} \leq x_{1h} \leq \epsilon x_{1i} \leq x_{1i}$ implique $x_{1i} \geq \epsilon x_{1h}$ et $u_{ih} = 0$, donc

$$A_h x_2 \leq A_i x_2.$$

Les conditions (5.2.2) sont donc satisfaites. De la même manière, les conditions (5.2.3) sont satisfaites en utilisant les variables binaires v_{jl} , pour tout $j, l \in \{1, 2, \dots, m\}$ avec $j \neq l$.

Enfin, avec $0 < \sigma \leq x_{2j}$, pour tout $j \in \{1, 2, \dots, m\}$, les conditions (5.2.4) sont satisfaites.

Réciproquement, la proposition suivante assure que tout équilibre ϵ -propre appartient à Ω_ϵ^σ pour toutes les valeurs suffisamment petites de σ .

Proposition 5.4.2 *Si (x_1, x_2) est un équilibre ϵ -propre pour des réels $\epsilon > 0$, alors il existe un réel $\bar{\sigma} > 0$ tel que $(x_1, x_2) \in \Omega_\epsilon^\sigma$ pour chaque $0 \leq \sigma \leq \bar{\sigma}$.*

Preuve 5.4.2 *Si (x_1, x_2) est un équilibre ϵ -propre pour des réels $\epsilon > 0$, les conditions (5.2.2) peuvent être reformulées en utilisant les variables binaires u_{ih} , pour tout $i, h \in$*

$\{1, 2, \dots, n\}, i \neq h :$

$$\text{Si } \begin{cases} A_i x_2 < A_h x_2, \\ x_{1i} \leq \epsilon x_{1h}, \end{cases} \text{ alors } \begin{cases} A_i x_2 \leq A_h x_2 + L u_{hi}, \\ x_{1i} + u_{ih} \leq \epsilon x_{1h} + 1, \\ u_{hi} = 0, \\ u_{ih} = 1. \end{cases}$$

$$\text{Si } \begin{cases} A_h x_2 < A_i x_2, \\ x_{1h} \leq \epsilon x_{1i}, \end{cases} \text{ alors } \begin{cases} A_h x_2 \leq A_i x_2 + L u_{ih}, \\ x_{1h} + u_{hi} \leq \epsilon x_{1i} + 1, \\ u_{ih} = 0, \\ u_{hi} = 1. \end{cases}$$

$$\text{Si } \begin{cases} A_i x_2 = A_h x_2, \\ x_{1i} \leq 1, \\ x_{1h} \leq 1, \end{cases} \text{ alors } \begin{cases} A_i x_2 \leq A_h x_2 + L u_{hi}, & x_{1i} + u_{ih} \leq \epsilon x_{1h} + 1, \\ A_h x_2 \leq A_i x_2 + L u_{ih}, & x_{1h} + u_{hi} \leq \epsilon x_{1i} + 1, \\ u_{hi} = 0, & u_{ih} = 0. \end{cases}$$

De la même manière, les conditions (5.2.3) peuvent être reformulées en utilisant les variables binaires v_{jl} , pour tout $j, l \in \{1, 2, \dots, m\}, j \neq l$.

Et finalement, les conditions (5.2.4) assurent qu'il existe un $\bar{\sigma} > 0$, tel que $\bar{\sigma} \leq x_{1i}$, pour tout $i \in \{1, 2, \dots, n\}$ et $\bar{\sigma} \leq x_{2j}$, pour tout $j \in \{1, 2, \dots, m\}$.

Donc pour chaque σ tel que $0 \leq \sigma \leq \bar{\sigma}$ et $\sigma > 0$:

$$\begin{aligned} \sigma &\leq x_{1i}, \text{ pour tout } i \in \{1, 2, \dots, n\}, \\ \sigma &\leq x_{2j}, \text{ pour tout } j \in \{1, 2, \dots, m\}. \end{aligned}$$

Ainsi, $(x_1, x_2) \in \Omega_\epsilon^\sigma$ pour chaque σ , tel que $0 \leq \sigma \leq \bar{\sigma}$ et $\sigma > 0$.

Myerson (48) et Jansen (28) définissent un équilibre *propre* comme étant la limite d'une

séquence d'équilibres ϵ_k -propres, avec ϵ_k convergeant vers zéro.

Définition 5.4.4 *Un équilibre (\hat{x}_1, \hat{x}_2) est dit propre si il existe une séquence d'équilibres ϵ_k -propres (x_1^k, x_2^k) tels que*

$$\lim_{k \rightarrow \infty} \epsilon_k = 0 \text{ et } \lim_{k \rightarrow \infty} (x_1^k, x_2^k) = (\hat{x}_1, \hat{x}_2). \quad (5.4.7)$$

Un équilibre propre est donc un équilibre qui peut être approximé par des séquences de mouvements dans lesquels chaque stratégie pure est choisie avec une probabilité strictement positive. Cependant, toute stratégie pure qui représenterait une erreur de choix serait choisie avec une probabilité beaucoup plus petite, par un multiple ϵ , que la probabilité de toute autre stratégie pure qui serait meilleure.

Tout équilibre propre est donc parfait et l'ensemble des équilibres propres d'un jeu stratégique représente un sous-ensemble non-vide de l'ensemble de ses équilibres parfaits.

En outre, tout équilibre propre de la représentation stratégique définit un équilibre séquentiel du jeu étendu (Myerson (49)).

À titre d'exemple, pour la représentation stratégique du jeu étendu représenté par la Figure 5.2, la séquence des mouvements

$$\hat{\sigma} = ((1 - \epsilon) [a_1 x_1] + 0.1\epsilon [a_1 y_1] + 0.1\epsilon [b_1 x_1] + 0.8\epsilon [b_1 y_1], \epsilon [x_2] + (1 - \epsilon) [y_2]),$$

est un équilibre ϵ -parfait pour tout $0 < \epsilon < 1/3$. Lorsque $\epsilon \rightarrow 0$, $\hat{\sigma}^k \rightarrow \sigma$ qui est un équilibre parfait pour cette représentation normale.

Cependant, $\hat{\sigma}$ n'est pas un équilibre ϵ -propre. La stratégie $b_1 y_1$ serait un choix erroné pire que $b_1 x_1$ pour le joueur 1 contre le joueur 2, car $0\epsilon + 2(1 - \epsilon) < 6\epsilon + 3(1 - \epsilon)$. Il faudrait donc que $\hat{\sigma}(b_1 y_1)/\hat{\sigma}(b_1 x_1)$ soit plus petit que ϵ pour avoir un équilibre ϵ -propre.

Le seul équilibre propre de la représentation normale de ce jeu est (b_1x_1, x_2) . Ceci se justifie par la séquence d'équilibres ϵ -propres de la forme :

$$((1 - \epsilon - 0.5\epsilon^2) [b_1x_1] + 0.5\epsilon^2 [b_1y_1] + 0.5\epsilon [a_1x_1] + 0.5\epsilon [a_1y_1], (1 - \epsilon) [x_2] + \epsilon [y_2]).$$

Cet équilibre propre correspond au seul équilibre séquentiel $([b_1x_1], [x_2])$ du jeu étendu de la Figure 5.2.

La principale difficulté est de trouver une séquence $\{\epsilon_k\}_{k \in \mathbb{N}}$ de réels positifs convergeant vers zéro et telle que la séquence d'équilibres ϵ_k -propres $\{(x_1^k, x_2^k)\}_{k \in \mathbb{N}}$ converge vers (\hat{x}_1, \hat{x}_2) , pour chaque $k \in \mathbb{N}$. Myerson (48) montre que chaque jeu bimatriciel possède au moins un équilibre propre.

La proposition (5.4.1) donne les conditions nécessaires et suffisantes pour qu'un équilibre soit parfait. Les conditions incluent la résolution d'une paire de programmes linéaires. L'objectif est d'arriver à formuler un théorème analogue pour les équilibres propres. Afin de réaliser cet objectif, notre idée est de résoudre des programmes quadratiques mixtes 0 – 1 paramétrés dont les solutions, lorsque le paramètre σ converge vers 0, définissent une séquence d'équilibres ϵ -propres.

Proposition 5.4.3 *L'équilibre parfait (\hat{x}_1, \hat{x}_2) est un équilibre propre si et seulement si le programme quadratique mixte 0 – 1 suivant est réalisable pour des réels $\bar{\sigma} > 0$, et si*

$$\lim_{\sigma \rightarrow 0^+} f(\sigma) = 0.$$

$$f(\sigma) = \min_{(x_1, x_2) \in \Omega_{\sigma, \epsilon}^*} \epsilon \tag{5.4.8}$$

s.à.

$$\hat{x}_{1i} - \epsilon \leq x_{1i} \leq \hat{x}_{1i} + \epsilon, \quad \forall i \in \{1, 2, \dots, n\},$$

$$\hat{x}_{2j} - \epsilon \leq x_{2j} \leq \hat{x}_{2j} + \epsilon, \quad \forall j \in \{1, 2, \dots, m\},$$

$$0 \leq \epsilon \leq 1.$$

Preuve 5.4.3 Soit $(x_1(\sigma), x_2(\sigma), \epsilon(\sigma))$ une solution optimale au programme (5.4.8) pour un équilibre parfait (\hat{x}_1, \hat{x}_2) . La proposition (5.4.1) montre que $(x_1(\sigma), x_2(\sigma))$, est un équilibre $\epsilon(\sigma)$ -propre. Les conditions (5.4.7) sont reformulées en utilisant la minimization de ϵ tel que

$$\begin{aligned}\hat{x}_{1i} - \epsilon &\leq x_{1i} \leq \hat{x}_{1i} + \epsilon, \quad \forall i \in \{1, 2, \dots, n\}, \\ \hat{x}_{2j} - \epsilon &\leq x_{2j} \leq \hat{x}_{2j} + \epsilon, \quad \forall j \in \{1, 2, \dots, m\},\end{aligned}$$

dans le but de faire converger les équilibres ϵ -propres vers (\hat{x}_1, \hat{x}_2) .

Ainsi, si le programme quadratique mixte 0 – 1 (5.4.8) est réalisable pour tout $\sigma < 1$, lorsque $\sigma > 0$ converge vers 0, nous pouvons conclure à partir de la proposition (5.4.2) qu'il existe toujours un équilibre ϵ -propre $(x_1, x_2) \in \Omega_\epsilon^\sigma$.

En plus, si l'équilibre parfait (\hat{x}_1, \hat{x}_2) est propre alors la valeur optimale de l'objectif $f(\sigma) = \epsilon(\sigma)$ doit nécessairement converger vers 0, lorsque $\sigma > 0$ converge vers 0, dans le but de faire converger la solution $(x_1(\sigma), x_2(\sigma))$ vers (\hat{x}_1, \hat{x}_2) au même temps. Nous pouvons aussi observer que $f(0) = 0$.

Dans le cas contraire, si $f(\sigma)$ ne converge pas vers 0, lorsque $\sigma > 0$ converge vers 0, alors une telle séquence d'équilibres $(x_1(\sigma), x_2(\sigma))$ $\epsilon(\sigma)$ -propres n'existe pas, lorsque ϵ converge vers 0. Dans ce cas, nous pouvons prouver que l'équilibre n'est pas propre.

En conclusion, si $f(\sigma)$ converge vers 0, lorsque $\sigma > 0$ converge vers 0, il est possible de trouver une séquence d'équilibres $(x_1(\sigma), x_2(\sigma))$ $\epsilon(\sigma)$ -propres convergeant vers (\hat{x}_1, \hat{x}_2) , lorsque $\epsilon(\sigma)$ converge vers 0. Trouver de telles séquences serait plus difficile et la preuve qu'un tel équilibre est propre reste ouverte à cause des bruits numériques qui peuvent apparaître.

En pratique, la proposition précédente (5.4.3) est difficile à appliquer directement dans

le but de montrer qu'un équilibre (\hat{x}_1, \hat{x}_2) est propre ou non. Nous l'utilisons cependant en cherchant la valeur de $f(\sigma)$ pour des valeurs réelles de σ . Le programme quadratique mixte 0 – 1 est résolu en utilisant l'algorithme QP implanté par Perron (53). Cet algorithme est une nouvelle version de l'algorithme QP de Audet (5). L'algorithme QP procure une solution ξ -optimale pour les programmes quadratiques réalisables, avec ξ le paramètre de précision utilisé lors de la résolution. Dans le but de résoudre le programme quadratique mixte 0 – 1 (5.4.8) avec QP, nous réécrivons les conditions sur les variables binaires u et v sous la forme de contraintes quadratiques $u_{ih} - u_{ih}^2 = 0$ et $v_{jl} - v_{jl}^2 = 0$. Grâce aux valeurs discrètes prises par les variables binaires, nous pouvons être certains que QP procure une solution optimale au programme quadratique mixte 0 – 1 (5.4.8) si il est réalisable et borné. Cependant, les bruits numériques qui apparaissent lorsque σ devient assez petit nous empêchent de conclure directement si un équilibre est propre.

Corollaire 5.4.1 *Soit $(x_1(\sigma), x_2(\sigma), \epsilon(\sigma))$ une solution optimale au programme quadratique mixte 0–1 (5.4.8) pour un $\sigma > 0$. Alors $(x_1(\sigma), x_2(\sigma))$ est un équilibre $\epsilon(\sigma)$ -propre, et si $\sigma'' > \sigma' > 0$, alors $\epsilon(\sigma'') \geq \epsilon(\sigma') \geq 0$.*

Preuve 5.4.4 *Si $\sigma'' > \sigma' > 0$, le programme quadratique mixte 0–1 (5.4.8) pour $\sigma' > 0$ est une relaxation du programme quadratique mixte 0 – 1 (5.4.8) pour $\sigma'' > 0$. En effet la seule différence entre ces deux programmes est dans les contraintes de $\Omega_\epsilon^{\sigma'}$ et $\Omega_\epsilon^{\sigma''}$:*

$$\begin{aligned} \sigma' \leq x_{1i}, \quad \forall i \in \{1, 2, \dots, n\}, \quad \text{et} \quad \sigma'' \leq x_{1i}, \quad \forall i \in \{1, 2, \dots, n\}, \\ \sigma' \leq x_{2j}, \quad \forall j \in \{1, 2, \dots, m\}, \quad \sigma'' \leq x_{2j}, \quad \forall j \in \{1, 2, \dots, m\}. \end{aligned}$$

$$\begin{aligned} \Rightarrow \quad \sigma' < \sigma'' \leq x_{1i}, \quad \forall i \in \{1, 2, \dots, n\}, \\ \sigma' < \sigma'' \leq x_{2j}, \quad \forall j \in \{1, 2, \dots, m\}. \end{aligned}$$

Donc $\Omega_\epsilon^{\sigma''} \subseteq \Omega_\epsilon^{\sigma'}$ et $\epsilon(\sigma'') \geq \epsilon(\sigma') \geq 0$.

Deux résultats sont possibles lorsque on évalue $f(\sigma)$ pour de petites valeurs de σ . Une des possibilités est que $f(\sigma)$ converge vers zéro. Malheureusement, ce résultat n'est pas suffisant pour conclure dans le cas général que l'équilibre est propre. Cependant, avec un tel résultat pour le cas particulier d'un équilibre régulier, nous pouvons conclure que l'équilibre est propre.

L'autre possibilité est que $f(\sigma)$ soit bornée inférieurement par un réel strictement positif $\bar{\epsilon}$. Ceci implique qu'il n'existe aucun équilibre ϵ -propre proche de (\hat{x}_1, \hat{x}_2) pour des valeurs de ϵ inférieures à $\bar{\epsilon}$, et ainsi (\hat{x}_1, \hat{x}_2) ne serait pas propre.

Dans (5.4.8), supposons que $f(\sigma)$ converge vers $\bar{\epsilon} > 0$, lorsque $\sigma > 0$ converge vers 0. Nous définissons alors un programme quadratique mixte 0–1 avec les mêmes conditions que Ω , avec $\epsilon \leq \bar{\epsilon}/2$ et maximisant σ . Si la valeur optimale de l'objectif de ce programme est égale à zéro, nous pouvons conclure qu'il est impossible de trouver une séquence d'équilibres $(x_1(\sigma), x_2(\sigma))$ $\epsilon(\sigma)$ -propres convergeant vers cet équilibre. Cet équilibre n'est donc pas propre.

Théorème 5.4.1 *Si la valeur optimale de l'objectif du programme quadratique mixte 0 – 1 suivant*

$$\begin{aligned} \max_{(x_1, x_2) \in \Omega_\epsilon^\sigma, \epsilon, \sigma} \quad & \sigma & (5.4.9) \\ \text{s.à.} \quad & \hat{x}_{1i} - \epsilon \leq x_{1i} \leq \hat{x}_{1i} + \epsilon, \quad \forall i \in \{1, 2, \dots, n\}, \\ & \hat{x}_{2j} - \epsilon \leq x_{2j} \leq \hat{x}_{2j} + \epsilon, \quad \forall j \in \{1, 2, \dots, m\}, \\ & 0 \leq \epsilon \leq \bar{\epsilon}/2. \end{aligned}$$

est égale à zéro pour des réels $\bar{\epsilon} > 0$, alors l'équilibre (\hat{x}_1, \hat{x}_2) n'est pas propre.

Preuve 5.4.5 *Si la valeur optimale de l'objectif est égale à zéro, il est impossible de trouver une séquence d'équilibres $(x_1(\sigma), x_2(\sigma))$ $\epsilon(\sigma)$ -propres convergeant vers (\hat{x}_1, \hat{x}_2) .*

L'équilibre (\hat{x}_1, \hat{x}_2) n'est pas propre.

Avec ce dernier résultat, la détection automatique des équilibres de Nash extrêmes non-propres peut être réalisée sur tout ensemble des équilibres de Nash extrêmes d'un jeu bimatriciel.

Exemple 5.4.1 *Pour le jeu bimatriciel de dimension (4×2) de Myerson (49), le seul équilibre propre est le premier équilibre de Nash extrême que nous avons énuméré. Pour les quatre autres équilibres de Nash extrêmes, la valeur optimale de ϵ dans (5.4.9) converge vers $\hat{\epsilon} = 0.618$, lorsque σ converge vers 0. Ce résultat suggère que les équilibres ne sont pas propres. Pour chacun des équilibres de Nash extrêmes 2, 3, 4 et 5, nous définissons un programme quadratique mixte 0 – 1 avec les mêmes conditions que (5.4.9) et avec $\epsilon \leq \bar{\epsilon}/2$ en maximisant σ . Un tel programme a une valeur optimale de son objectif égale à zéro. Les équilibres de Nash extrêmes 2, 3, 4 et 5 de ce jeu ne sont donc pas propres. Pour le troisième équilibre de Nash extrême, la Figure 5.3 montre comment $f(\sigma)$ dans (5.4.9) décroît lorsque σ décroît de $\min(\frac{1}{4}, \frac{1}{2}) = \frac{1}{4}$ vers 0.*

L'ensemble des équilibres de Nash extrêmes et propres définit l'ensemble des points extrêmes des sous-ensemble de Myerson maximaux (28).

Exemple 5.4.2 *Pour le cinquième équilibre de Nash extrême énuméré pour le jeu bimatriciel (Exemple 1.4.1) de dimension (3×4) énoncé par Borm et. al., la valeur optimale de ϵ dans le programme (5.4.8) converge vers 0.284, lorsque σ converge vers 0. Ainsi, ϵ ne converge pas vers 0 lorsque σ converge vers 0. Ceci suggère que cet équilibre extrême n'est pas propre. Les équilibres de Nash extrêmes parfaits 1, 3, 4, 6, 7, 8 et 10 de ce jeu bimatriciel sont ϵ -propres pour des valeurs de ϵ très proches de zéro. Le Tableau 5.5 montre les résultats des expériences numériques sur les programmes quadratiques mixtes 0 – 1 (5.4.8) définis pour chacun des équilibres de Nash extrêmes de ce jeu.*

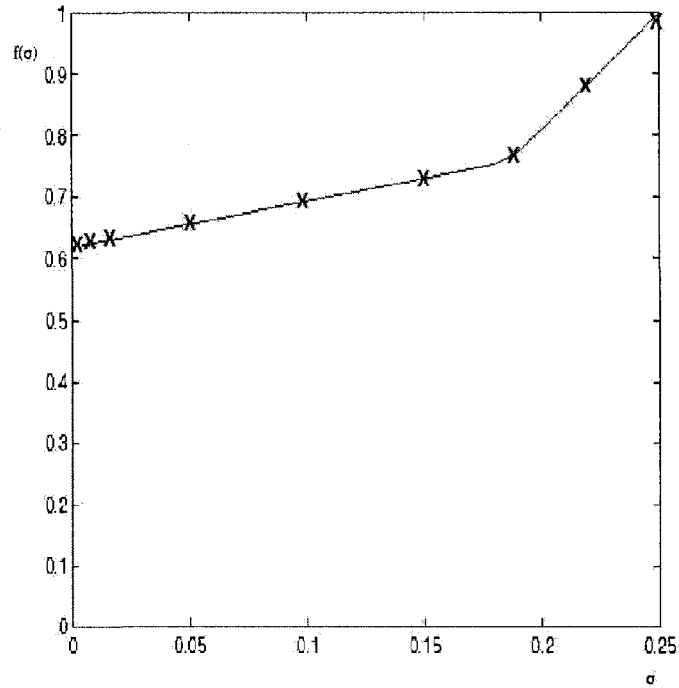


Figure 5.3 Courbe de $f(\sigma) = \min \epsilon$, pour l'équilibre 3

Exemple 5.4.3 Le jeu bimatriciel (5×5) suivant possède sept équilibres de Nash extrêmes identifiés par le Tableau 5.6.

$$A = B = \begin{pmatrix} 2 & 4 & 5 & 6 & 7 \\ 2 & 1 & 0 & 8 & 1 \\ 2 & 5 & 6 & 0 & 1 \\ 0 & 2 & 5 & 4 & 7 \\ 2 & 3 & 6 & 5 & 7 \end{pmatrix}$$

Ce jeu possède quatre ensembles de Nash maximaux $T_1 = \{1, 2, 6\}$, $T_2 = \{3, 4\}$, $T_3 = \{5\}$ et $T_4 = \{7\}$. Deux de ses équilibres de Nash extrêmes sont réguliers.

Tableau 5.5 Raffinement des équilibres de Nash extrêmes pour l'exemple de Borm et. al. (14)

| Eq. | Parfait | Propre | $\bar{\epsilon}$ | $\bar{\sigma}$ | Quasi-fort | Isolé | Régulier |
|-----|------------|------------|------------------|--------------------|------------|------------|------------|
| 1 | <i>oui</i> | <i>oui</i> | 0.027 | 10^{-4} | <i>non</i> | <i>non</i> | <i>non</i> |
| 2 | <i>non</i> | <i>non</i> | 0.8019 | 10^{-4} | <i>non</i> | <i>non</i> | <i>non</i> |
| 3 | <i>oui</i> | <i>oui</i> | 0.014 | 10^{-4} | <i>non</i> | <i>non</i> | <i>non</i> |
| 4 | <i>oui</i> | <i>oui</i> | 0.01 | 10^{-4} | <i>non</i> | <i>non</i> | <i>non</i> |
| 5 | <i>non</i> | <i>non</i> | 0.2705 | 10^{-4} | <i>non</i> | <i>non</i> | <i>non</i> |
| 6 | <i>oui</i> | <i>oui</i> | 0.052 | 10^{-3} | <i>non</i> | <i>non</i> | <i>non</i> |
| 7 | <i>oui</i> | <i>oui</i> | 0.046 | 10^{-4} | <i>non</i> | <i>non</i> | <i>non</i> |
| 8 | <i>oui</i> | <i>oui</i> | 0.056 | 10^{-3} | <i>non</i> | <i>non</i> | <i>non</i> |
| 9 | <i>oui</i> | <i>oui</i> | 0.081 | 5×10^{-4} | <i>non</i> | <i>non</i> | <i>non</i> |
| 10 | <i>oui</i> | <i>oui</i> | 0.0247 | 2×10^{-4} | <i>non</i> | <i>non</i> | <i>non</i> |

Tableau 5.6 Équilibres de Nash extrêmes pour le jeu bimatriciel (5×5)

| Eq. | x_1 | | | | | x_2 | | | | | α_1 | α_2 |
|-----|-------|-----|-----|---|-----|-------|---|---|-----|-----|------------|------------|
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 7 | 7 |
| 2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 7 | 7 |
| 3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 6 | 6 |
| 4 | 0 | 0 | 1/6 | 0 | 5/6 | 0 | 0 | 1 | 0 | 0 | 6 | 6 |
| 5 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 8 | 8 |
| 6 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 7 | 7 |
| 7 | 7/8 | 1/8 | 0 | 0 | 0 | 0 | 0 | 0 | 3/4 | 1/4 | 25/4 | 25/4 |

Équilibre 5 : $C(x_1) = \{2\}$ et $C(x_2) = \{4\}$, $M(A, x_2) = \{2\}$ et $M(x_1, B) = \{4\} \Rightarrow$ quasi-fort. Le déterminant de $\begin{pmatrix} 8 \end{pmatrix}$ est égal à $8 \neq 0 \Rightarrow$ équilibre isolé. Cet équilibre est régulier, essentiel, parfait et propre.

Équilibre 7 : $C(x_1) = \{1, 2\}$ et $C(x_2) = \{4, 5\}$, $M(A, x_2) = \{1, 2\}$ et $M(x_1, B) = \{4, 5\} \Rightarrow$ quasi-fort. Le déterminant de $\begin{pmatrix} 6 & 7 \\ 8 & 1 \end{pmatrix}$ est égal à $-50 \neq 0 \Rightarrow$ équilibre isolé. Cet équilibre est régulier, essentiel, parfait et propre.

Tableau 5.7 Exemple (5×5)

| <i>Eq.</i> | <i>Parfait</i> | <i>Propre</i> | $\bar{\epsilon}$ | $\bar{\sigma}$ | <i>Quasi-fort</i> | <i>Isolé</i> | <i>Régulier</i> |
|------------|----------------|---------------|------------------|--------------------|-------------------|--------------|-----------------|
| 1 | <i>oui</i> | <i>non</i> | 0.2894 | 5×10^{-3} | <i>non</i> | <i>non</i> | <i>non</i> |
| 2 | <i>non</i> | <i>non</i> | 0.7325 | 10^{-3} | <i>non</i> | <i>non</i> | <i>non</i> |
| 3 | <i>oui</i> | <i>oui</i> | 0.05627 | 10^{-5} | <i>non</i> | <i>non</i> | <i>non</i> |
| 4 | <i>oui</i> | <i>non</i> | 0.2000 | 10^{-6} | <i>non</i> | <i>non</i> | <i>non</i> |
| 5 | <i>oui</i> | <i>oui</i> | 0.0564 | 10^{-5} | <i>oui</i> | <i>oui</i> | <i>oui</i> |
| 6 | <i>oui</i> | <i>oui</i> | 0.054 | 10^{-5} | <i>non</i> | <i>non</i> | <i>no</i> |
| 7 | <i>oui</i> | <i>oui</i> | 0.0776 | 6×10^{-5} | <i>oui</i> | <i>oui</i> | <i>oui</i> |

5.5 Discussion

Dans ce chapitre, nous avons exposé les raffinements majeurs en théorie des jeux. Nous avons proposé une paire de programmes linéaires dont la résolution permet d'affirmer si un équilibre de Nash est parfait ou non. Nous avons aussi obtenu une série de résultats concernant les équilibres propres. Un premier résultat permet de trouver la valeur minimale de ϵ pour laquelle un équilibre est ϵ -propre. Une série de propositions a permis de conclure par un théorème qui énonce un programme quadratique mixte 0 – 1 dont la résolution permet de conclure si un équilibre n'est pas propre (2).

CHAPITRE 6

CONTRIBUTIONS ET PERSPECTIVES

Au début de ce projet de thèse de doctorat, nous nous sommes fixé plusieurs objectifs pour notre recherche en théorie des jeux. Nos objectifs de formulation mathématique et d'implantation ont été réalisés. Nous exposons dans ce chapitre ces différentes réalisations et nous détaillons aussi les perspectives de travaux futurs basés sur cette thèse.

6.1 Contributions

Notre premier objectif était d'utiliser la linéarisation des conditions de complémentarité afin de concevoir un algorithme d'énumération des équilibres extrêmes pour les jeux bimatriciels et polymatriciels.

Dans le premier chapitre, nous avons effectué une synthèse des principales formulations mathématiques pour la recherche d'équilibres de Nash pour les jeux bimatriciels. Nous avons exposé l'algorithme de Lemke et Howson (37) pour la recherche d'équilibre de Nash. Ensuite, nous avons présenté l'algorithme *EEE* (6) pour l'énumération des équilibres de Nash extrêmes. Nous avons aussi montré comment le problème d'énumération des sous-ensembles de Nash maximaux peut être résolu en utilisant un algorithme d'énumération des cliques maximales en théorie des graphes. Finalement, nous avons linéarisé les conditions de complémentarité en utilisant la méthode de Júdice et Mitra (29) et en introduisant des variables binaires. Cette linéarisation a permis de proposer un programme linéaire mixte 0 – 1 dont les solutions extrêmes sont des équilibres de Nash extrêmes du jeu bimatriciel.

Le deuxième chapitre a été consacré en premier lieu à une synthèse des formulations mathématiques pour la recherche d'équilibres de Nash pour les jeux polymatriciels. Nous avons ensuite exposé l'algorithme des pivots complémentaires de recherche d'équilibre de Nash pour les jeux polymatriciels (70). En troisième lieu, nous avons approfondi la méthode d'élimination des stratégies dominées pour les jeux bimatriciels et polymatriciels à trois joueurs. Finalement, nous avons linéarisé les conditions de complémentarité en introduisant des variables binaires. Cette nouvelle modélisation a permis de proposer un programme linéaire mixte 0 – 1 dont les solutions extrêmes sont des équilibres de Nash extrêmes du jeu polymatriciel.

Nous avons donc réalisé notre premier objectif en définissant en définissant pour chaque forme de jeu un programme linéaire mixte 0 – 1 dont les solutions extrêmes sont des équilibres de Nash extrêmes. Grâce à l'algorithme *E χ MIP* nous pourrions utiliser ces formulations afin d'énumérer les équilibres de Nash extrêmes d'un jeu bimatriciel ou d'un jeu polymatriciel à trois joueurs.

Notre deuxième objectif était d'étudier l'énumération des équilibres de Nash extrêmes pour la forme séquentielle d'un jeu étendu à deux joueurs. Nous avons présenté dans le troisième chapitre une revue de la littérature sur les différents concepts utilisés pour les jeux étendus. Ensuite, nous avons étudié les différentes représentations d'un jeu étendu à deux joueurs. Finalement, nous avons proposé un programme linéaire mixte 0 – 1 dont les solutions extrêmes permettent, par projection sur l'espace des variables des stratégies mixtes, d'énumérer les équilibres de Nash extrêmes de la représentation séquentielle d'un jeu étendu à deux joueurs.

Notre troisième objectif était de concevoir en premier lieu un algorithme permettant d'énumérer tous les équilibres de Nash extrêmes pour les jeux bimatriciels, polymatriciels et les formes séquentielles des jeux étendus à deux joueurs et d'implanter ce nouvel algorithme ainsi que l'algorithme *EEE* en utilisant une librairie d'arithmétique exacte.

Nous avons commencé le quatrième chapitre en présentant l'algorithme $E\chi MIP$ pour l'énumération des équilibres extrêmes en théorie des jeux. En première étape, nous avons implanté cet algorithme en utilisant les bibliothèques du logiciel d'optimisation *Cplex*. Notons que l'algorithme $E\chi MIP$ est actuellement le seul algorithme permettant d'énumérer tous les équilibres de Nash extrêmes d'un jeu polymatriciel. Les deux algorithmes EEE et $E\chi MIP$ sont aussi actuellement les deux seuls algorithmes permettant d'énumérer tous les équilibres de Nash extrêmes de la forme séquentielle d'un jeu étendu à deux joueurs.

Nous avons aussi proposé dans le quatrième chapitre une formulation biniveau permettant d'étendre l'algorithme EEE aux formes séquentielles des jeux étendus à deux joueurs. Nous avons pu implémenter les algorithmes EEE et $E\chi MIP$, en *C* et *C++*, en faisant appel à la bibliothèque d'arithmétique exacte de McCutchen (44). Nous avons décrit dans ce chapitre les détails techniques de ces implantations. En outre, nous y avons exposé le logiciel *XGame Solver* qui regroupe ces implantations sous une application à interfaces en *QT* facile à manipuler et mise gratuitement à la disposition de la communauté scientifique.

Notre quatrième objectif était de définir des méthodes automatiques pour le raffinement des équilibres en théorie des jeux. Dans le cinquième chapitre, nous avons exposé les différents raffinements dans la littérature de théorie des jeux. Nous avons proposé une paire de programmes linéaires dont la résolution permet d'affirmer si un équilibre de Nash est parfait ou non. Nous avons aussi obtenu une série de propositions concernant les équilibres propres. Ces propositions ont permis de conclure par un théorème qui définit un programme quadratique mixte 0 – 1 dont la résolution permet de conclure si un équilibre n'est pas propre.

6.2 Perspectives

En somme, cette thèse de doctorat a permis de réaliser les objectifs fixés et nous comptons poursuivre notre recherche en théorie des jeux afin d'exploiter ces quelques contributions. Tout d'abord, nous allons étendre nos études à d'autres formes de jeux. Nous allons implanter d'autres algorithmes d'énumération ou de recherche d'équilibres. Nous pensons aussi à quelques applications pratiques aux algorithmes *EEE* et *E χ MIP* dans plusieurs domaines.

Une première perspective de recherche serait d'étudier d'autres formes de jeux. Une des possibilités serait d'étudier le problème d'un jeu où le groupe majoritaire d'un chorum de porteurs de parts se partage à lui seul les revenus auxiliaires d'une organisation. Nous pensons utiliser la programmation linéaire en variables mixtes afin d'étudier ce jeu à majorité. Nous voyons aussi l'intérêt que la programmation quadratique pourrait avoir dans la caractérisation des ensembles d'équilibres d'un jeu stratégique non-polymatriciel à n joueurs. Entre autre, cette forme de jeu est la représentation stratégique d'un jeu étendu à plusieurs joueurs.

Une deuxième perspective de recherche serait d'ajouter au logiciel *XGame Solver* d'autres algorithmes de recherche d'équilibres de Nash pour les jeux bimatriciels tels que les algorithmes de Lemke et Howson (37) et Avis et Fukuda (10). D'autres catégories de raffinement pourrait être étudiés et incorporés dans *XGame Solver*. Nous comptons garder ce logiciel continuellement mis à jour afin d'en faire bénéficier le plus grand nombre de scientifiques.

Une troisième perspective de recherche serait d'utiliser notre formulation mathématique des jeux polymatriciels afin d'étudier une des formes de jeux les plus actives dans notre vie contemporaine. Parmi ces confrontations stratégiques les plus intenses, les confrontations entre groupes de pression et les organisations à intérêts privés et publics, sont des

plus passionnantes. En effet, les impacts des manifestations pacifistes ou non-pacifistes sur les législations, dans plusieurs pays industrialisés et non-industrialisés, deviennent de plus en plus importants. Par ailleurs, les firmes privées même les plus monopolistiques deviennent beaucoup plus sensibles aux attentes des consommateurs en matière d'écologie.

En effet, les groupes de pression et les organisations à intérêts privés et publics exploitent des moyens médiatiques, économiques (12) et stratégiques (13; 11), qui rencontrent de plus en plus de succès. Nous essayerons de formuler ce genre de confrontation comme des jeux polymatriciels ou non-polymatriciels à plusieurs joueurs.

RÉFÉRENCES

AGGARWAL, V., (1973). On the Generation of All Equilibrium Points for Bimatrix Games through the Lemke-Howson Algorithm, *Mathematical Programming*, 4, 233-234.

AUDET, C., BELHAIZA, S., HANSEN, P., (2007). Perfect and Proper Refinements of all extreme Nash equilibria for bimatrix games, *Les Cahiers du Gerad*, GERAD-HEC Montréal, G-2007-85, novembre, 2007.

AUDET, C., BELHAIZA, S., HANSEN, P., (2006). A new sequence form approach for the enumeration and refinement of all extreme Nash equilibria for extensive form games, *Les Cahiers du Gerad*, GERAD-HEC Montréal, G-2006-10, février, 2006.

AUDET, C., BELHAIZA, S., HANSEN, P., (2006). Enumeration of all Extreme Equilibria in Game Theory : Bimatrix and Polymatrix Games, *Journal of Optimization Theory and Applications*, 129-3, 349-372.

AUDET, C., HANSEN, P., JAUMARD, B., SAVARD, G., (2000). A branch and cut algorithm for nonconvex quadratically constrained quadratic programming, *Mathematical Programming*, 87-1, 131-152.

AUDET, C., HANSEN, P., JAUMARD, B., SAVARD, G., (2001). Enumeration of all extreme equilibrium strategies of bimatrix games, *SIAM Journal on Scientific Computing* 23-1, 323-338.

AUDET, C., HANSEN, P., JAUMARD, B., SAVARD, G., (1998). Complete Enumeration of Equilibria for Two-Person Games in Strategic and Sequence Forms, *Proceedings Eight International Symposium on Dynamic Games and Applications*, Maastricht, Netherlands.

AUDET, C., HANSEN, P., JAUMARD, B., SAVARD, G., (1997) Links Between Linear Bilevel and Mixed 0-1 Programming Problems, *Journal of Optimization Theory and Applications* 93-2, 273-300.

AUDET, C., (1997). Optimisation Globale Structurée : Propriétés, Équivalences et Résolution", *Thèse de doctorat, École Polytechnique de Montréal*, 91-109.

AVIS, C., FUKUDA, K., (1992). A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra, *Discrete and Computational Geometry* 8, 295-313.

BARON, D.P., (2003). Private Politics, *Journal of Economics and Management Strategy* 12-1, 31-66.

BARON, D.P., (2003). A Theory of Boycotts, *Working paper*, Stanford University, Stanford, CA.

BARON, D.P., YURDAY, E., (2004). Strategic Activism : The Rainforest Action Network, *Graduate School of Business*, Case P-44, Stanford University, Stanford, CA.

BORM, P.E.M., JANSEN, M.J.M, POTTERS, J.A.M., TIJS, S.H., (1993). On the structure of the set of perfect equilibria in bimatrix games, *O-R Spektrum* 15, 17-20.

BRON, C., KERBOSCH, J., (1971). Algorithm 457 : Finding All Cliques of an Undirected Graph, *Communications of the ACM* 16-9, 575-577.

ChVÁTAL, V., (1998). "Linear Programming", Freeman, New York.

COTTLE, R.W., DANTZIG, G.B., (1963). "Linear Programming Extensions", Princeton University Press, Princeton, N.J.

COTTLE, R.W., DANTZIG, G.B., (1968). Complementary Pivot Theory of Mathematical Programming, *Mathematics of the Decision Sciences, Part 1*, 11.

DANTZIG, G.B., (1951). Maximization of a Linear Function of Variables Subject to Linear Inequalities, *Chap. XXI of "Activity Analysis of Production and Allocation"*, edited by T.C. Koopmans, John Wiley, New York, pp. 359–373.

DICKHAUT, J., KAPLAN, T., (1991). A Program for Finding Nash Equilibria, *The Mathematica Journal* 1, 87-93.

EAVES, C.B., (1973). Polymatrix Games with Joint Constraints, *Society for Industrial and Applied Mathematics Journal of applied Mathematics* 24-3, 418-423.

HARSANYI, J.C., (1973). Games with randomly distributed payoffs : a new rationale for mixed-strategy equilibrium points, *International Journal of Game Theory* 2, 235-250.

HARSANYI, J.C., SELTEN, R., (1998). A General Theory of Equilibrium Selection in Games, The MIT Press, Cambridge, Massachusetts, London, England.

HOWSON, J.T., (1972). Equilibria of Polymatrix Games, *Management Science* 18, 312-318.

HOWSON, J.T., ROSENTHAL, R.W., (1974). Bayesian Equilibria of Finite two-Person Games with Incomplete Information, *Management Science* 21, 313-315.

JANSEN, M.J.M., (1981). Regularity and stability of Equilibrium Points of bimatrix games, *Mathematics and Operations Research* 6, 530-550.

JANSEN, M.J.M., (1987). Regular Equilibrium Points of Bimatrix Games, *OR Spektrum* 9, 87-92.

JANSEN, M.J.M., (1993). On the set of Proper Equilibria of a Bimatrix Game, *International Journal of Game Theory* 22, 97-106.

JUDICE, J.J., MITRA, G., (1988). Reformulations of mathematical programming problems as linear complementarity problems and investigation of their solution methods, *Journal of Optimization Theory and Applications* 57, 123-149.

KEIDING, H., (1997). On the Maximal Number of Nash Equilibria in a bimatrix game, *Games and Economic Behavior* 21(1-2), 148-160.

KOHLBERG, E., Mertens, J.F., (1986). On the strategic stability of equilibria, *Econometrica* 54, 1003–1038.

KOJIMA, M., OKADA, A., SHINDOH, S., (1985). Strongly stable equilibrium points of n -person noncooperative games, *Mathematics and Operations Research* 10, 650-663.

KOLLER, D., MEGIDDO, N., (1992). The complexity of two-person zero-sum games in extensive form, *Games and Economic Behavior* 4, 528–552.

KOLLER, D., MEGIDDO, N., (1996). Finding mixed strategies with small supports in extensive form games, *International Journal of Game Theory* 25, 73–92.

KUHN, H.W., (1961). An Algorithm for Equilibrium Points in Bimatrix Games, *Proceedings of the National Academy of Sciences* 47, 1657-1662.

KUHN, H.W., (1953). "Extensive games and the problem of information", *Contributions to the theory of Games II*, edited by H.W.Kuhn and A.Tucker, Annals of Mathematics Studies 28, Princeton Univ. Press, Princeton, 1953, 193–216.

LEMKE, C.E, HOWSON, T.T., (1964). Equilibrium Points of Bimatrix Games, *Journal of the Society for Industrial and Applied Mathematics* 12, 413-423.

LEMKE, C.E, HOWSON, T.T., (1965). Bimatrix Games Equilibrium Points and Mathematical Programmings, *Management Science* 11, 681–689

LUCE, R.D., RAIFFA, R., (1957). "Games and Decisions", Wiley, New York.

MANGASARIAN, O.L., (1964). Equilibrium Points of Bimatrix Games, *Journal of the Society for Industrial and Applied Mathematics* 12, 778-780.

MANGASARIAN, O.L., STONE, H., (1964). Two Person Nonzero-sum Games and Quadratic Programming, *Journal of Mathematical Analysis and Applications* 9, 348-355.

MCKELVEY, R.D., MCLENNAN, A.M., (1996). Computation of Equilibria in Finite Games, *Handbook of Computational Economics*, Vol I, (éditeurs Amman H.M., Kendrick D.A et Rust.J.). Elsevier, Amsterdam, 87-142.

MCKELVEY, R.D., MCLENNAN, A.M., TUROCY, T.L., (2007). Gambit : Software Tools for Game Theory, Version 0.2007.01.30, <http://econweb.tamu.edu/gambit/>.

MCCUTCHEN, M., (2007). An easy-to-use C++ library for calculations on big integers, <http://www.mattmccutchen.net/>.

MILLHAM, C.B., (1974). On Nash Subsets of Bimatrix Games, *Naval Research Logistics Quarterly* 74, 307-317.

MILLS, H., (1960). Equilibrium Points in Finite Games, *Journal of the Society for Industrial and Applied Mathematics* 8, 397-402.

MUKHAMADIEV, B.M., (1978). The Solution of Bilinear Programming Problems and Finding the Equilibrium Situations in Bimatrix Games, version anglaise : U.S.S.R.

Computational Mathematics and Mathematical Physics 18-2, 60-66.

MYERSON, R.B., (1978). Refinements of the Nash equilibrium concept, *International Journal of Game Theory* 7, 73-80.

MYERSON, R.B., (1997). "Game Theory : Analysis of Conflict", Harvard University Press, Cambridge-Massachusetts, London-England.

NASH, J.F., (1950). Equilibrium points in n-person games, *Proceedings of the National Academy of Sciences* 36, 48-49.

NASH, J.F., (1950). Non-Cooperative Games, *Annals of Mathematics* 54, 286-295.

OKADA, A., (1984). Strictly perfect equilibrium points of bimatrix games, *International Journal of Game Theory*, 13, 145-154.

PERRON, S., (2005). "Applications jointes de l'optimisation combinatoire et globale", Thèse de doctorat, École Polytechnique de Montréal.

QUINTAS, L.G., (1989). A Note on Polymatrix Games, *International Journal of Game Theory* 18-19, 261-272.

TUCKER, A.W., (1967). Pivotal Algebra, *Lecture Notes par Parson T.D.*, Department of Mathematics, Princeton University.

ROMANOVSKII, I.V., (1962). Reduction of a game with complete memory to a matrix game, *Soviet Mathematics* 3, 678–681.

SELTEN, R., (1975). Reexamination of the perfectness concept for equilibrium points in extensive games, *International Journal of Game Theory* 4, 22-55.

SELTEN, R., (1988). Evolutionary stability in extensive two-person games - correction and further development, *Mathematical Social Sciences* 16, 223–266.

VAN DAMME, E.E.C., (1983). Refinements of the Nash equilibrium concept, Springer, Berlin.

VON STENGEL, B., (1996) : Efficient computation of behavior strategies, *Games and Economic Behavior* 14, 220–246.

VON STENGEL, B., (1998). Computing Equilibria for Two-Person Games, *Handbook of Game Theory* 3.

VON STENGEL, B., (1998). New maximal numbers of equilibria in bimatrix games, *Discrete and Computational Geometry* 21, 557–568.

VON STENGEL, B., (1998). Improved equilibrium enumeration for bimatrix games. Extended Abstract of talk at OR'98, *International Conference on Operations Research*, ETH Zurich, Aug 31-Sept 3, (Unpublished).

VON STENGEL, B., VAN DEN ELZEN A., TALMAN D., (2002). Computing normal form perfect equilibria for extensive two-person games, *Econometrica* 70, 693–715.

VOROB'EV, N.N., (1958). Equilibrium Points in Bimatrix Games, *Thoriya Veroyatnostej i ee Primvneniya* 3, 318-331 [version anglaise : *Theory of Probability and its Applications* 3, 297-309].

WILSON, R., (1972). Computing equilibria of two-person games from the extensive form, *Management Science* 18, 448–460.

WILSON, R.,(1971). Computing Equilibria of N-Person Games, *Society for Industrial and Applied Mathematics Journal of applied Mathematics* 21-1, 80-87.

WINKELS, R., (1972). An Algorithm to Determine all Equilibrium Points of a Bimatrix Games, *Game Theory and Related Topics* (éditeurs Moeschlin O. et Pallaschke D.). North Holland Publishing Company, Amsterdam, New York, Oxford, 137-148.

WU WEN-TSUN, JIANG JIA-HE, (1972). Essential equilibrium points of n -person non cooperative games, *Sci.Sinica* 11, 1307-1322.

YANOVSKAYA, E.B., (1968). Equilibrium points in Polymatrix Games, (en russe) *Latvian Mathematical Collection* 8, 381-384.