Modélisation des interactions entre agents rationnels : les jeux booléens

Elise Bonzon

THÈSE

présentée devant

l'Université Paul Sabatier de Toulouse III

U.F.R. MATHÉMATIQUES, INFORMATIQUE ET GESTION

pour obtenir le titre de

DOCTEUR DE L'UNIVERSITÉ PAUL SABATIER

Mention INFORMATIQUE

par

ELISE BONZON

École doctorale :	Informatique et Télécommunication
Laboratoire d'accueil :	Institut de Recherche en Informatique de Toulouse
Équipe d'accueil :	Raisonnements Plausibles, Décision, Méthodes de Preuves

Modélisation des interactions entre agents rationnels : les jeux booléens

soutenue le 13 Novembre 2007 devant la commission d'examen :

Nicholas ASHER	Directeur de recherche, Université Paul Sabatier (membre)
Annie ASTIÉ-VIDAL	Professeur, Université Paul Sabatier (invité)
Olivier GASQUET	Professeur, Université Paul Sabatier (membre)
Andreas HERZIG	Directeur de recherche, Université Paul Sabatier (invité)
Marie-Christine LAGASQUIE-SCHIEX	Maître de conférences, Université Paul Sabatier (directrice de thèse)
Jérôme LANG	Chargé de recherche, Université Paul Sabatier (directeur de thèse)
Pierre MARQUIS	Professeur, Université d'Artois (rapporteur)
Nicolas MAUDET	Maître de conférences, Université Paris Dauphine (membre)
Leon van der TORRE	Professeur, Université du Luxembourg (rapporteur)
Bruno ZANUTTINI	Maître de conférences, Université de Caen (membre)

Elise Bonzon

MODÉLISATION DES INTERACTIONS ENTRE AGENTS RATIONNELS : LES JEUX BOOLÉENS

Directeurs de thèse :
Jérôme Lang, CR CNRS, HdR
Marie-Christine Lagasquie-Schiex, Maître de conférences
Université Paul Sabatier

Résumé

Les jeux booléens permettent de représenter les jeux stratégiques d'une manière succincte en tirant profit du pouvoir d'expression et de la concision de la logique propositionnelle. Informellement, un jeu booléen est un jeu à deux joueurs, chacun d'entre eux contrôlant un ensemble de variables propositionnelles, et à somme nulle. La fonction d'utilité du joueur 1 (et donc celle du joueur 2 qui est son opposé) est représentée par une formule de la logique propositionnelle, appelée *forme booléenne* du jeu. Ainsi, un joueur dans un jeu booléen a des préférences dichotomiques : son but est satisfait ou ne l'est pas.

Ces trois restrictions (deux joueurs, somme nulle, préférences binaires) limitent fortement l'expressivité de ce cadre. Les deux premières restrictions peuvent être facilement résolues en définissant les préférences des agents comme étant un n-uplet de formules propositionnelles (une pour chaque agent). Des outils simples issus de la logique propositionnelle nous permettent ainsi de caractériser certaines propriétés du jeu. Deux autres notions ont alors été étudiées : la dépendance entre joueurs (si le but, et donc la satisfaction, d'un joueur i dépend de variables contrôlées par le joueur j, alors i aura besoin de j pour satisfaire son but) et les coalitions de joueurs (une coalition dans un jeu booléen est efficace si elle peut garantir à tous ses membres que leurs buts sont satisfaits). Dans les deux cas, l'objectif est de faciliter le calcul des concepts de solution tels que les équilibres de Nash en stratégies pures.

Lever la troisième restriction consiste à exprimer des préférences (non binaires) dans un cadre propositionnel. Cela est possible en utilisant un langage de représentation compacte de préférences. Nous avons integré donc deux de ces langages aux jeux booléens : tout d'abord, les *buts à priorité* puis les *CP-nets*.

Institut de Recherche en Informatique de Toulouse - UMR 5505
Université Paul Sabatier, 118 route de Narbonne. 31062 TOULOUSE cedex 9

Elise Bonzon

BOOLEAN GAMES

Supervisors :
Jérôme Lang, CR CNRS, HdR
Marie-Christine Lagasquie-Schiex, Maître de conférences
Université Paul Sabatier

Abstract

Boolean games are a logical setting for representing static games in a succinct way, taking advantage of the expressive power and conciseness of propositional logic. Boolean games allow to express compactly two-players zero-sum static games with binary preferences : an agent's strategy consists of a truth assignment of the propositional variables she controls, and a player's preferences are expressed by a plain propositional formula.

These three restrictions (two-players, zero-sum, binary preferences) strongly limit the expressivity of the framework. The first two can be easily encompassed by defining the agents' preferences as an arbitrary n-uple of propositional formulas. Two others notions have been studied : dependencies between players (if the goal, and hence the satisfaction, of a player i depends on some variables controlled by a player j, then i may need some action of j to see her goal satisfied) and efficient coalitions (a coalition in a Boolean game is efficient if it has the power to guarantee that all goals of the members of the coalition are satisfied). We give simple characterizations of Nash equilibria and dominated strategies, and investigate the computational complexity of the related problems.

Then, we relax the last restriction by coupling Boolean games with propositional languages for compact preference representation ; we consider generalized Boolean games where players' preferences are expressed within the two following languages : propositionalized CP-nets, and then prioritized goals.

Institut de Recherche en Informatique de Toulouse - UMR 5505
Université Paul Sabatier, 118 route de Narbonne. 31062 TOULOUSE cedex 9

Remerciements

Je voudrais en premier lieu remercier Jérôme LANG et Marie-Christine LAGASQUIE-SCHIEX pour m'avoir aidée, accompagnée, encadrée et encouragée tout au long de ces trois années. Leur présence, leurs conseils et leur complémentarité m'ont permis non seulement de mener ce travail à bien, mais également de découvrir et d'aimer le monde de la recherche, comme celui de l'enseignement.

J'aimerais ensuite remercier Pierre MARQUIS et Leon van der TORRE pour avoir accepté d'être rapporteurs de ma thèse, et pour l'avoir lue avec autant d'attention. Merci pour leurs conseils, remarques et commentaires, qui m'ont permis d'améliorer ce manuscrit tant sur le fond que sur la forme.

Merci également à Nicholas ASHER, Annie ASTIÉ-VIDAL, Olivier GASQUET, Andreas HERZIG, Nicolas MAUDET et Bruno ZANUTTINI pour avoir accepté de participer à mon jury de thèse, et pour leurs commentaires qui ouvrent de nombreuses pistes de recherche. Je tiens particulièrement à remercier Pierre, Leon, Nicolas et Bruno qui ont accepté de se déplacer à Toulouse malgrè les grèves.

Merci à Florence BOUÉ et Martine LABRUYÈRE qui m'ont permis de traverser sans encombres toutes les embûches administratives semées sur le chemin de la soutenance. Merci Florence d'avoir été aussi efficace pour gérer tous les problèmes de dernières minutes.

Merci à Ulle ENDRISS pour m'avoir accueillie 6 semaines au sein de l'ILLC à Amsterdam, pour m'avoir ainsi permis de découvrir un autre environnement de travail, et pour m'avoir consacré du temps et de l'énergie.

Merci à Marie-Christine LAGASQUIE-SCHIEX, Florence BANNAY, Martin STRECKER, et tous les moniteurs de la promo 2004 pour m'avoir accompagné dans les divers enseignements que j'ai accompli. Ils sont pour beaucoup dans le plaisir que j'ai aujourd'hui à enseigner.

Merci à Jorge CHAM qui m'a permis de découvrir de façon ludique les côtés obscurs du milieu académique.

Merci aux membres du troisième étage de l'Irit qui, de pauses café en pauses repas, m'ont appris que le milieu de la recherche peut être extrêmement convivial.

Merci à mes copines de bureau, Marie et Sihem, pour avoir grandement contribué à rendre l'ambiance de travail très agréable.

Merci à Nicolas, Axel, Kevin, Virginie, Alexia, Sylvain, Florian, Mylen, Filou, Caroline, Cédric, Marie, Julie, Amélie, Elodie, Manue, Xavier, Dominique, Marjolaine et tant d'autres pour m'avoir permis de ne pas oublier le monde extérieur pendant ces trois ans.

Merci à ma famille, mes parents, mes frères et sœurs, pour m'avoir soutenue tout au long de ces années, et d'avoir toujours été là pour moi.

Et merci à Jérémie, d'être là tout simplement.

Table des matières

Introduction

Contexte

L’intelligence artificielle permet à des machines d’effectuer des tâches réputées intelligentes, en concevant et en réalisant des fonctions cognitives artificielles, et en utilisant la puissance calculatoire des ordinateurs. Parmi ces tâches se trouvent, entre autres, l’acquisition et la représentation des connaissances, la formalisation et la mécanisation de différents types de raisonnement, l’aide à la décision collective, la planification, les systèmes multi-agents, etc.

Objet de nombreuses recherches en intelligence artificielle, les systèmes multi-agents s’inspirent des sciences humaines pour modéliser des groupes d’agents. Un système multi-agent est un ensemble d’agents situés dans un certain environnement et interagissant selon une certaine organisation. Un agent rationnel est une entité caractérisée par le fait qu’elle est, au moins partiellement, autonome. Ce peut être un processus, un robot, un être humain, etc. La création de systèmes multi-agents conduit à cinq problématiques principales :

* Problématique de l’action : comment plusieurs agents peuvent-ils agir de manière simultanée dans un environnement commun, et comment cet environnement interagit en retour avec les agents ? Les questions sous-jacentes à celle-ci sont entre autres celles de la représentation de l’environnement par les agents, de la collaboration entre agents et de la planification multi-agents.
* Problématique de l’agent et de sa relation au monde : de quel modèle cognitif dispose l’agent pour représenter le monde ? Comment un agent peut-il mettre en œuvre les actions qui répondent au mieux à ses objectifs ? Cette capacité à la décision est liée à “l’état mental” de l’agent, qui reflète ses perceptions, ses représentations, ses croyances, ses désirs, ses préférences, etc.
* Problématique de l’interaction : quelles formes d’interaction existent entre les agents ? Comment est-elle représentée ? Quels moyens de communication existent entre les agents ?
* Problématique de l’adaptation : quelles sont les possibilités et les moyens d’adaptation individuelle et d’apprentissage pour chaque agent ? Existe-t’il un modèle d’adaptation collective ou d’évolution ?
* Problématique de l’implémentation : comment implémenter des systèmes multi-agents ? Et notamment comment choisir le langage de communication entre agents, le langage de description des lois de l’environnement, celui de représentation des connaissances des agents et de leurs préférences ?

En reprenant ces cinq problématiques, il est possible de décrire quelques éléments de l’architecture d'un système multi-agent :

* Les agents doivent être dotés de systèmes de décision. Les théories de la décision sont un domaine à part entière d'étude à ce sujet (voir par exemple [Hansson, 1991] pour un aperçu du domaine).
* Les agents doivent être dotés d'un modèle cognitif. Là-aussi, plusieurs modèles existent, l'un des plus classiques étant le modèle BDI (*Beliefs-Desires-Intentions*) [Rao et Georgeff, 1991]. Ce modèle considère d'une part l'ensemble de croyances (*Beliefs*) de l'agent sur son environnement, qui sont le résultat de ses connaissances et de ses perceptions ; l'ensemble de ses objectifs (*Desires*), qui sont les états possibles envers lesquels l'agent peut vouloir s'engager ; et enfin l'ensemble des intentions (*Intentions*) qui regroupe l'ensemble des projets qu'il a l'intention de mener à bien.
* Les agents doivent être dotés d'un système de communication. Plusieurs langages spécialisés ont vu le jour à cette fin : le *Knowledge Query and Manipulation Language* (KQML, [Finin *et al.*, 1994; Labrou et Finin, 1997]), et plus récemment, le standard *FIPA-ACL* créé par la *Foundation for Intelligent Physical Agents FIPA*. Ce dernier standard repose en particulier sur la théorie des actes de langage [Searle, 1969].
* La problématique de l'adaptation est un sujet épineux, objet de nombreuses recherches à l'heure actuelle. On pourrait toutefois citer l'exemple de certains virus, aussi bien biologiques qu'informatiques, capables de s'adapter à leur environnement en mutant.
* Enfin, l'implémentation effective du système multi-agent, si elle ne fait pas à proprement parler partie de l'architecture du système, mérite d'être évoquée à travers l'exemple des nombreux langages de programmation qui ont été développé à des fins de recherche en intelligence artificielle (en particulier le langage LISP). Les langages de représentation utiles à ces sytèmes, que ce soit pour représenter les connaissances des agents, leurs préférences, ou encore la façon dont ils communiquent, ont également fait l'objet de nombreuses recherches en intelligence artificielle.

Bien que toutes ces problématiques soient liées, nous avons choisi de nous intéresser plus spécifiquement dans ce manuscrit à la *problématique de l'interaction*, et à la *façon de représenter les préférences* de chacun des agents. Comme nous l'avons évoqué plus haut, les agents d'un système peuvent avoir besoin de communiquer afin de résoudre des différences d'opinions et des conflits d'intérêts, de travailler ensemble afin de résoudre des dilemmes, de trouver des preuves, ou tout simplement d'échanger des informations. De plus, chaque agent a des préférences au sein d'un système multi-agent, des désirs sur les états du monde qu'il souhaite atteindre, et ceux qu'il souhaite éviter. Le "sort" de chacun des agents, c'est-à-dire la satisfaction de ses préférences, dépend alors non seulement de ses propres décisions, mais aussi des décisions prises par les autres membres du sytème. La décision "optimale" pour un individu dépend donc généralement de ses propres actions, mais également de ce que font les autres agents. Comme chacun n'est pas totalement maître de son sort, on dit que les agents sont en *interaction stratégique*. Plusieurs cadres de travail existent pour traiter cette problématique.

Cadre de travail choisi

La théorie des jeux est probablement le modèle formel le plus abouti pour l'étude des interactions stratégiques entre agents. Plusieurs raisons nous ont amené à choisir ce cadre de travail. Tout d'abord les jeux permettent de décrire des situations sociales très différentes : les marchés en économie peuvent être vus comme des jeux dans lesquels les participants sont des producteurs ou des consommateurs ; et, plus généralement, une partie d'échecs, la formation d'une coalition gouvernementale ou encore une négociation sont autant de jeux différents obéissant à des règles spécifiques. Ensuite, ce cadre de travail est, et a été, l'objet de nombreuses recherches, et est très riche en résultats. C'est également un cadre de travail très intuitif à manipuler : il est en effet facile de visualiser les interactions entre agents.

Informellement, un jeu consiste en un ensemble d'agents (ou joueurs), et pour chaque agent, la donnée d'un ensemble de stratégies possibles et une fonction d'utilité associant une valeur réelle à chaque combinaison possible de stratégies, représentant ses préférences. Des hypothèses sur les croyances de l'agent au cours du jeu doivent être en outre spécifiées pour les jeux dynamiques à information incomplète.

Nous étudierons ici les interactions entre joueurs et la représentation de leurs préférences en nous plaçant dans un cadre simple : les jeux statiques à information complète. Un jeu est statique si les agents choisissent leur stratégie en parallèle et en une seule étape, sans observer les choix des autres joueurs. Il est à information complète si chaque joueur connaît exactement l'état du monde, les préférences et les actions disponibles pour chacun des joueurs.

Plusieurs modes de représentation sont utilisés en théorie des jeux, notamment les formes extensives et les formes normales qui coïncident dans le cas des jeux statiques. Les fonctions d'utilité sont généralement décrites explicitement dans ces représentations, en énumérant toutes les valeurs pour chaque combinaison de stratégies. Le nombre de valeurs numériques à spécifier, c'est-à-dire le nombre de combinaisons de stratégies possibles, est alors exponentiel en fonction du nombre de joueurs, ce qui rend cette représentation explicite des préférences des joueurs déraisonnable lorsque le nombre de joueurs est grand. Ceci devient encore plus problématique lorsque l'ensemble des stratégies disponibles pour un agent consiste à assigner une valeur à chacune de ses variables à partir d'un domaine fini donné (ce qui est le cas dans beaucoup de domaines réalistes). Dans ce cas, la représentation explicite des fonctions d'utilité est de taille exponentielle en fonction du nombre d'agents ($n \times 2^n$ valeurs pour n agents ayant chacun deux stratégies disponibles) et en fonction du nombre de variables contrôlées par les agents ($2 \times 2^p \times 2^p$ valeurs pour deux agents contrôlant chacun p variables booléennes). Ainsi, spécifier les préférences des joueurs de façon explicite est clairement peu raisonnable, tout d'abord car cela nécessiterait une quantité d'espace exponentielle, puis parce qu'étudier ces jeux (en calculant par exemple des concepts de solution comme les équilibres de Nash en stratégies pures) exigerait d'accéder à toutes ces valeurs d'utilité au moins une fois, et serait donc exponentiel en temps en fonction du nombre d'agents et du nombre de variables.

Une solution pour répondre à ces besoins consiste à utiliser un langage permettant une représentation concise des relations de préférences, ou des fonctions d'utilité, sur un ensemble de conséquences qui possède une structure combinatoire (c'est-à-dire un produit cartésien de domaines de valeurs finis pour un ensemble fini de variables). Ces langages ont été activement étudiés ces dernières années, spécialement dans la communauté d'intelligence artificielle. Ils exploitent dans une large mesure des propriétés structurelles des relations de préférences (comme l'indépendance préférentielle entre variables). Partant de là, puisque la spécification d'un jeu statique nécessite la description des préférences des agents, il apparaît naturel de représenter de tels jeux en utilisant des langages de représentation compacte de préférences. Il existe déjà plusieurs cadres répondant aux problèmes que nous avons posés plus haut, notamment les langages graphiques que nous présenterons dans le chapitre 4 (page 121).

Proposition et méthodologie

Comme nous l'avons vu, notre objectif ici est de spécifier de façon concise et efficace les interactions entre agents rationnels au sein d'un système multi-agents. Pour cela, nous avons choisi de nous appuyer sur la théorie des jeux, qui est un modèle formel abouti pour l'étude de ces interactions. Pourtant, une des principales lacunes de cette théorie est la façon de représenter les utilités des joueurs, coûteuse en place mémoire et en temps d'exécution. Nous avons vu qu'une solution pour pallier ces problèmes est d'utiliser un langage de représentation compacte de préférences. La première étape consiste donc à choisir un tel langage, puis d'étudier les jeux pouvant être formalisés dans ce cadre : étude des différents concepts de solution de ces jeux, étude de la complexité des problèmes associés. L'étape suivante consiste à essayer de généraliser ces jeux en introduisant de nouvelles représentations compactes de préférences.

Choix d'un langage de représentation compacte de préférences

Afin de trouver un langage de représentation compacte de préférence répondant à nos besoins, nous devons commencer par répondre à une question préliminaire : les buts des agents doivent-ils être exprimés avec des préférences numériques ou ordinales ? Un langage de représentation de préférences doit être aussi proche que possible de la façon dont les individus "connaissent" leurs préférences et les expriment en langage naturel. Dans ce cadre, le problème conceptuel de l'utilité cardinale est qu'il n'existe pas d'échelle objective de la mesure de l'utilité, et qu'il est donc difficile pour un individu d'évaluer numériquement l'utilité apportée par la satisfaction de chacun de ses désirs. Dans le cadre de l'utilité ordinale, il est demandé au consommateur de pouvoir classer raisonnablement les mondes disponibles en fonction de l'utilité apportée, ce qui semble a priori plus aisé à effectuer pour de nombreux individus.

Les notions que nous voulons étudier dans un jeu représentent un autre critère de choix entre préférences numériques ou ordinales. En effet, quelques notions (telles que équilibres de Nash en stratégies pures et stratégies dominées) peuvent être définies avec des préférences ordinales, tandis que d'autres (telles que équilibres de Nash en stratégies mixtes) ont besoin de préférences cardinales. Ici, nous avons choisi de nous restreindre aux calculs de stratégies pures, et *nous avons donc choisi de représenter les préférences des joueurs de façon ordinale*.

Le langage de représentation compacte de préférences ordinal le plus simple et le plus intuitif paraît être la *logique propositionnelle*. En effet, utiliser la logique propositionnelle pour représenter les préférences des joueurs permet non seulement de simplifier les jeux, mais aussi d'utiliser les propriétés et les outils bien connus de cette logique pour étudier les caractéristiques de tels jeux. Nous avons donc choisi ici d'étudier le cas où chaque agent contrôle un ensemble fini de variables *binaires*. Ces jeux, appelés *jeux booléens*, ont été introduits par [Harrenstein *et al.*, 2001; Harrenstein, 2004a; Dunne et van der Hoek, 2004]. Un jeu booléen est un jeu à deux joueurs, chacun d'entre eux contrôlant un ensemble de variables propositionnelles, et à somme nulle. La fonction d'utilité du joueur 1 (et donc celle du joueur 2 qui est son opposé) est représentée par une formule de la logique propositionnelle, appelée *forme booléenne* du jeu, qui doit être satisfaite pour que le joueur soit également satisfait. Chaque joueur a donc des préférences *dichotomiques* (soit il est satisfait, soit il ne l'est pas, sans niveau intermédiaire).

Ces trois restrictions (deux joueurs, somme nulle, préférences binaires) limitent fortement l'expressivité des jeux booléens. Nous avons donc commencé par généraliser ces jeux en étudiant des jeux à n joueurs et à somme non nulle, puis nous avons introduit des préférences non binaires.

Etude des jeux booléens

Les jeux booléens peuvent être facilement transformés en des jeux à n joueurs à somme non nulle en définissant les préférences des agents comme étant un n-uplet de formules propositionnelles. Des outils simples issus de la logique propositionnelle nous permettent alors de caractériser certaines propriétés du jeu, comme les équilibres de Nash en stratégies pures et les stratégies dominées, et de calculer la complexité des problèmes associés.

Deux autres notions ont alors été étudiées : les dépendances entre joueurs (si le but, et donc la satisfaction, d'un joueur i dépend de variables contrôlées par le joueur j, alors i aura besoin de j pour satisfaire son but), et les coalitions de joueurs (une coalition dans un jeu booléen est efficace si elle peut garantir à tous ses membres que leurs buts sont satisfaits). Ces deux nouvelles notions nous ont permis de faciliter encore le calcul des concepts de solution tels que les équilibres de Nash en stratégies pures.

Introduction de nouvelles préférences

Tandis qu'une simple formule propositionnelle φ ne peut pas exprimer plus qu'une relation de préférence binaire sur les interprétations (les modèles de φ sont strictement meilleurs que les modèles de $\neg\varphi$), exprimer des préférences (non binaires) dans un cadre propositionnel est possible en utilisant un autre langage de représentation compacte de préférences. Nous avons choisi de nous restreindre encore aux préférences ordinales, et nous avons integré deux de ces langages aux jeux booléens : tout d'abord, les *buts à priorité* puis les *CP-nets*.

Plan du manuscrit

Après avoir donné dans le chapitre 1 quelques éléments sur la théorie des jeux, nous donnerons dans le chapitre 2 une description (simplifiée) des jeux booléens puis nous montrerons qu'ils peuvent facilement être généralisés de manière à représenter des jeux avec un nombre arbitraire de joueurs et à somme non nulle, mais en gardant l'hypothèse que les préférences de chaque joueur sont représentées par une formule propositionnelle unique, ce qui ne permet de représenter que des utilités binaires. Nous verrons comment introduire la notion de dépendance entre les joueurs, et comment des outils simples issus de la logique propositionnelle permettent de caractériser certaines propriétés du jeu, puis nous donnerons quelques résultats de complexité algorithmique. Nous introduirons ensuite dans le chapitre 3 deux langages de représentation de préférences afin d'enrichir encore les jeux booléens avec des préférences non dichotomiques : les buts à priorité puis les CP-nets. Dans le chapitre 4, nous exposerons quelques-uns des travaux utilisant des langages de représentation compacte de préférences dans des jeux, en montrant le lien avec nos solutions. Nous étudierons ensuite dans le chapitre 5 les coalitions efficaces dans les jeux booléens. Nous étudierons tout d'abord les propriétés des fonctions d'effectivité associées aux jeux booléens, puis la notion de coalition efficace : nous donnerons une caractérisation exacte des ensembles de coalitions correspondant à des ensembles de coalitions efficaces associées à un jeu booléen, et nous ferons le lien entre coalition efficace et noyau avant de conclure.

La figure 1 donne un guide de lecture des chapitres : une flèche d'un chapitre vers un autre indique que la lecture du premier est nécessaire à la compréhension du second.

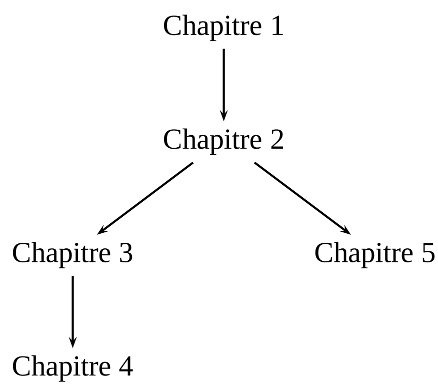


***Figure 1* —** Guide de lecture

Notations

Soit $V = \{a, b, \ldots\}$ un ensemble fini de variables propositionnelles et L_V le langage propositionnel construit à partir de V, des connecteurs habituels et des constantes booléennes $\top$ (*vrai*) et $\perp$ (*faux*). Les formules de L_V seront notées φ, ψ etc. On note $Var(\varphi)$ l'ensemble des variables présentes dans la formule φ.

Un littéral est, soit une variable de V, soit sa négation. Une conjonction finie de littéraux est appelée *terme* ou *cube*, et une disjonction finie de littéraux est appelée une *clause*. On note $Lit(\varphi)$ l'ensemble des littéraux formant la formule φ. Une formule φ est en DNF si c'est une disjonction de termes.

2^V est l'ensemble des interprétations pour V avec la convention suivante : soit M une interprétation pour V et pour tout $x \in V$, M donne la valeur *vrai* à x si $x \in M$ et *faux* sinon. Soit M une interprétation pour V et $\psi \in L_V$, la conséquence logique $M \models \psi$ est définie de la manière usuelle.

Soit $X \subseteq V$. 2^X est l'ensemble des *X-interprétations*. Une *interprétation partielle* de L_V est une X-interprétation pour $X \subseteq V$. Les interprétations partielles sont représentées par une liste de variables de X, le symbole $\bar{}$ représentant la négation d'une variable. Par exemple, si $X = \{a, b, d\}$, la X-interprétation $M = \{a, d\}$ sera notée $a\bar{b}d$. Si $Var(\varphi) \subseteq X$, alors $Mod_X(\varphi)$ représente l'ensemble des X-interpretations satisfaisant φ.

Si $\{V_1, \ldots, V_p\}$ est une partition de V et si $\{M_1, \ldots, M_p\}$ sont des interprétations partielles, avec $M_i \in 2^{V_i}$, $(M_1, \ldots, M_p)$ représente alors l'interprétation $M_1 \cup \ldots \cup M_p$.

Rappellons que, quelle que soit la formule φ, une interprétation qui rend φ vrai est un *modèle* de φ.

L'interprétation partielle d'une formule φ par une X-interprétation M_X sera noté : $(\varphi)_{M_X} = \varphi_{v \in M_X \leftarrow \top, v \in X \setminus M_X \leftarrow \perp}$.

Nous aurons également besoin dans ce document de plusieurs notions d'impliquants premiers. Les définitions suivantes sont reprises du rapport de synthèse [Marquis, 2000].

Intuitivement, un impliquant premier d'une formule propositionnelle ψ est un des plus petits termes dont tous les modèles sont des modèles de ψ.

Définition 1. *Soit ψ une formule propositionnelle.*

- *Un terme α est un* impliquant *de ψ ssi $\alpha \models \psi$.*
- *Un terme α est un* impliquant premier *de ψ ssi*
  - *α est un impliquant de ψ, et*
  - *pour chaque impliquant α' de ψ, si $\alpha \models \alpha'$, alors $\alpha' \models \alpha$.*

On notera $PI(\psi)$ l'ensemble des impliquants premiers de ψ.

Un L-impliquant (resp. L-impliquant premier) est un impliquant (resp. impliquant premier) dont tous les littéraux appartiennent à l'ensemble L.

Définition 2. *Soit $L \subseteq V$ et soit ψ une formule propositionnelle de L_V.*

- *Un terme α est un* L-impliquant *de ψ ssi $\alpha \models \psi$ et $Lit(\alpha) \subseteq L$.*
- *Un terme α est un* L-impliquant premier *de ψ ssi*
  - *α est un L-impliquant de ψ, et*
  - *pour chaque L-impliquant α' de ψ, si $\alpha \models \alpha'$, alors $\alpha' \models \alpha$.*

  On notera $PI_L(\psi)$ l'ensemble des L-impliquants premiers de ψ.

La notion de projection d'une formule sur un ensemble de variables correspond à l'utilisation de l'opérateur *forget*, qui a été étudié par [Lang *et al.*, 2002a, 2003].

On dit que l'on "oublie complètement" la variable x dans une formule φ si et seulement si on s'intéresse à la formule (notée $\forall x : \varphi$) : $\varphi_{x\leftarrow\top} \wedge \varphi_{x\leftarrow\bot}$.

On peut aussi "oublier partiellement" x dans φ en s'intéressant à la formule (notée $\exists x : \varphi$) : $\varphi_{x\leftarrow\top} \vee \varphi_{x\leftarrow\bot}$.

On note que l'on a :

$$\forall x : \varphi \equiv \neg\exists x : \neg\varphi$$

Par exemple, soit la formule φ suivante :

$$\varphi = (a \wedge \neg b \wedge c) \vee (b \wedge \neg c) \vee (a \wedge b \wedge d)$$

La projection de φ sur la variable c sera calculée comme suit :

$$\begin{aligned} \exists c : \varphi &= [(a \wedge \neg b \wedge \top) \vee (b \wedge \bot) \vee (a \wedge b \wedge d)] \vee [(a \wedge \neg b \wedge \bot) \vee (b \wedge \top) \vee (a \wedge b \wedge d)] \\ &= (a \wedge \neg b) \vee b \vee (a \wedge b \wedge d) \end{aligned}$$

$$\begin{aligned} \forall c : \varphi &= [(a \wedge \neg b \wedge \top) \vee (b \wedge \bot) \vee (a \wedge b \wedge d)] \wedge [(a \wedge \neg b \wedge \bot) \vee (b \wedge \top) \vee (a \wedge b \wedge d)] \\ &= ((a \wedge \neg b) \vee (a \wedge b \wedge d)) \wedge (b \vee (a \wedge b \wedge d)) \\ &= a \wedge b \wedge d \end{aligned}$$

1 Eléments de théorie des jeux

La théorie des jeux est un outil mathématique permettant d'étudier les comportements - prévus, réels, ou justifiés a posteriori - d'individus face à des situations d'antagonisme.

Si ses précurseurs furent Cournot [Cournot, 1838] et Edgeworth [Edgeworth, 1897], c'est la parution en 1944 de l'ouvrage de John Von Neumann et Oskar Morgenstern, *The Theory of Games and Economic Behaviour* [von Newmann et Morgenstern, 1944], qui instaura véritablement la théorie des jeux comme étant une nouvelle discipline. Dans cet ouvrage, Von Neumann et Morgenstern proposent une solution dans le cas particulier d'un *jeu à somme nulle* (ce qui est gagné par l'un est perdu par l'autre, et réciproquement). En 1950, John Nash [Nash, 1950] a montré comment les idées développées par Cournot dès 1838 pouvaient servir de base pour construire une théorie de l'équilibre pour des jeux à somme non nulle, qui généralise la solution proposée par Von Neumann et Morgenstern. Les économistes, premiers à s'approprier cet outil, ont été depuis rejoints par, entre autres, les sociologues, les chercheurs en sciences politiques, les philosophes, ou encore les informaticiens.

La théorie des jeux étudie des situations dans lesquelles le sort de chaque participant dépend non seulement des décisions qu'il prend, mais également des décisions prises par d'autres participants. Le choix optimal pour un agent (appelé **joueur**) dépend donc généralement des choix des autres agents. Comme chaque joueur n'est pas totalement maître de son sort, on dit que les agents sont en situation d'*interaction stratégique*. On suppose dans un jeu en interaction stratégique que les joueurs se connaissent : ils savent combien il y a de joueurs, et qui ils sont. Du fait que le gain de chacun dépend en partie des actions des autres, un joueur ne peut pas se contenter de choisir ses propres plans d'actions, en négligeant ce que font les autres. Il doit au contraire se faire une idée aussi précise que possible des stratégies choisies, ou susceptibles d'être choisies, par les autres joueurs. Pour cela, on admet que **les agents sont rationnels**, c'est-à-dire que chaque joueur s'efforce de prendre les meilleures décisions pour lui-même, et sait que les autres joueurs font de même.

Notre objectif ici n'est pas de donner un état de l'art exhaustif sur la théorie des jeux, nous voulons juste introduire quelques concepts qui nous seront utiles dans la suite de ce manuscrit. Nous allons donc tout d'abord présenter une taxonomie partielle des jeux en Section 1.1, puis les types de représentation des jeux en Section 1.2, et enfin nous présenterons en Section 1.3 quelques concepts de solution.

1.1 Taxonomie partielle des jeux

Nous allons présenter ici quatre types de jeux, les jeux statiques, dynamiques, coopératifs et non coopératifs. Un jeu peut réunir plusieurs de ces caractéristiques : il peut être statique et coopératif, statique et non coopératif, dynamique et coopératif ou encore dynamique et non coopératif.

1.1.1 Jeux statiques et dynamiques

La première distinction que nous allons faire est celle entre jeu statique et dynamique. Un jeu est statique lorsque tous les joueurs jouent simultanément en une seule étape, alors qu'il est dynamique lorsque le jeu se déroule en plusieurs étapes (un ou plusieurs joueurs peuvent jouer à chaque étape).

1.1.1.1 Jeux statiques

Un jeu est dit **statique** lorsque les joueurs choisissent **simultanément** leurs actions, et reçoivent ensuite leurs gains respectifs.

Définition 1.1. *Un* **jeu statique** *est un ensemble de règles qui encadre le comportement des joueurs et qui détermine les gains des joueurs selon les actions entreprises. Formellement, un jeu G est constitué de :*

* *un* **ensemble de joueurs** $N = \{1, \ldots, n\}$,
* **l'ensemble des profils de stratégies ou issues du jeu** $S = S_1 \times \ldots \times S_n$, *où* S_i. *représente l'ensemble des choix possibles (stratégies) du joueur i. On suppose dans la suite que les* S_i *sont finis.*
* $\forall i \in N$, *une* **fonction d'utilité** *qui représente les gains du joueur en fonction de l'issue du jeu :* $u_i : S \to \mathbb{R}$. *Le joueur i préfère strictement l'issue s à l'issue* s' *si* $u_i(s) > u_i(s')$. *Si* $u_i(s) = u_i(s')$, *i est indifférent entre ces deux issues.*

$\forall i \in N$, S_i *représente toutes les* **stratégies** s_i *disponibles pour le joueur i. Un* **profil de stratégies**, *appelé aussi* **issue** *du jeu, est une combinaison de stratégies individuelles* $s = (s_1, \ldots, s_n) \in S_1 \times \ldots \times S_n$.

Soit G un jeu statique, avec $N = \{1, \ldots, n\}$, $s = (s_1, \ldots, s_n)$ et $s' = (s'_1, \ldots, s'_n)$ deux profils de stratégies. On note s_{-i} le *profil de stratégies s privé de la stratégie du joueur i* : $s_{-i} = (s_1, s_2, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$. On note (s_{-i}, s'_i) *le profil de stratégies s dans lequel on a remplacé la stratégie du joueur i par celle du profil* s' : $(s_{-i}, s'_i) = (s_1, s_2, \ldots, s_{i-1}, s'_i, s_{i+1}, \ldots, s_n)$. On note $S_I = \times_{i \in I} S_i$ l'ensemble des stratégies pour $I \subseteq N$.

Le dilemme du prisonnier, que nous allons présenter maintenant, est un exemple célèbre de la théorie des jeux.

Exemple 1.1. *Deux suspects sont retenus dans des cellules séparées, et ne peuvent donc pas communiquer. La police ne dispose pas d'éléments de preuve suffisants pour obtenir leur condamnation, l'aveu d'au moins un des deux est donc indispensable. La police propose à chacun d'entre eux le marché suivant :*

* *si vous avouez et que votre complice n'avoue pas, vous aurez une remise de peine, tandis que votre complice aura la peine maximale (10 ans) ;*

* *si vous avouez tous les deux, vous serez condamnés à une peine plus légère (5 ans) ;*
* *si aucun de vous n'avoue, la peine sera minimale (6 mois), faute d'éléments au dossier.*

On peut formaliser cette situation par un jeu à 2 joueurs, chaque joueur ayant 2 stratégies possibles : avouer (dénotée par A) ou se taire (dénotée par T).

* $N = \{1,2\}$,
* *le prisonnier 1 a deux stratégies possibles :* $s_{1_1} = A$ *et* $s_{1_2} = T$*. Il en est de même pour le joueur 2 :* $s_{2_1} = A$ *et* $s_{2_2} = T$.
* *Ce jeu a donc 4 profils de strategies possibles : AA,* AT, TA *et* TT.
* *On peut donc calculer les utilités de chacun des joueurs pour chacune des issues possibles du jeu :*
  * $u_1(AA) = u_2(AA) = -5$,
  * $u_1(TT) = u_2(TT) = -0.5$,
  * $u_1(AT) = u_2(TA) = 0$,
  * $u_2(AT) = u_1(TA) = -10$.

La bataille des sexes est également un jeu célèbre en théorie des jeux :

Exemple 1.2. *Lucas et Elsa veulent aller au cinéma. Ils ont le choix entre un film d'horreur et une comédie romantique. Pour les deux, ce qui compte avant tout, c'est d'être ensemble. Néanmoins, Elsa a une préférence pour le film d'horreur et Lucas pour la comédie romantique.*

On peut formaliser cette situation par un jeu à 2 joueurs, chaque joueur ayant 2 stratégies possibles : aller voir un film d'horreur (dénotée par H) ou une comédie romantique (dénotée par R).

* $N = \{1,2\}$,
* *Elsa* (1) *a 2 stratégies possibles :* $s_{1_1} = H$ *et* $s_{1_2} = R$*. Lucas* (2) *a les même possibilités :* $s_{2_1} = H$ *et* $s_{2_2} = R$.
* *On peut donc calculer les utilités de chacun des joueurs pour chacune des issues possibles du jeu :*
  * $u_1(H,H) = u_2(R,R) = 2$,
  * $u_1(R,R) = u_2(H,H) = 1$,
  * $u_1(H,R) = u_1(R,H) = u_2(H,R) = u_2(R,H) = 0$.

1.1.1.2 Jeux dynamiques

Un jeu dynamique est un jeu qui se déroule en **plusieurs étapes**. On se place ici dans le cadre des **jeux dynamiques en information complète**, c'est-à-dire que l'on admet que toutes les actions passées sont *observables* et *connues* de tous les joueurs. Dans ce cadre, en intervenant à des étapes antérieures du jeu, certains joueurs ont le pouvoir d'affecter directement les gains d'autres joueurs de manière irréversible.

Un jeu est **en information parfaite** si chaque joueur connaît l'ensemble des actions choisies par *tous* les joueurs qui sont intervenus avant qu'il ne sélectionne sa stratégie[1], et qu'il connaît toutes leurs stratégies possibles. Il est le seul joueur à prendre une décision à cette étape[2]. Si plusieurs joueurs

[1]Dans le cadre des jeux statiques, la notion d'information parfaite n'a aucun sens : les joueurs jouent simultanément et en une seule étape, et n'ont donc pas à connaître les actions déjà réalisées.

[2]Par exemple, le jeu d'échecs est un jeu à information parfaite.

choisissent leurs actions simultanément à une étape donnée, ou si les joueurs ne connaissent pas toutes les stratégies des autres joueurs[3], le jeu est dit **en information imparfaite**. Ces actions ne sont pas connues et chacun des joueurs intervenant à cette étape se comporte un peu comme dans un jeu statique, à la différence que dans ce cas, l'histoire du jeu influence le choix de chacun.

Définissons le cadre conceptuel général des jeux dynamiques. Soit G un jeu dynamique. On désigne par a^t **le vecteur des actions choisies à l'étape** t **du jeu** par les participants qui interviennent à cette étape (un seul joueur dans le cas d'un jeu en information parfaite). Soit t une **étape** quelconque du jeu. On définit l'**histoire** $h^t = (a^0, a^1, \ldots, a^{t-1})$ du jeu à l'étape t par la séquence de toutes les décisions prises par les joueurs intervenant lors des étapes antérieures. On suppose que tous les joueurs connaissent l'histoire du jeu à chaque étape, c'est-à-dire que toutes les actions passées sont observables et connues par tous les participants. Le reste du jeu (toutes les étapes r telles que $r > t$) est appelé **sous-jeu** de G, et est noté $G(h^t)$. Puisque l'histoire h^t du jeu à chaque étape t est connue, le sous-jeu se déroulant à partir de t peut être vu comme un jeu à part entière induit par l'histoire h^t. L'histoire h^t impose des restrictions sur les choix offerts au joueur i. Soit $A_i(h^t)$ l'**ensemble des actions** auxquelles le joueur i a accès à l'étape t du jeu lorsque l'histoire est donnée par h^t. Si $A_i(h^t)$ est vide, le joueur i n'intervient pas à l'étape considérée[4]. Soit H^t l'**ensemble de toutes les histoires possibles** jusqu'à l'étape t. $A_i(H^t) = \bigcup_{h^t \in H^t} A_i(h^t)$ désigne alors l'ensemble de toutes les actions possibles pour le joueur i à l'étape t selon les histoires possibles.

Nous pouvons à présent définir une stratégie pure d'un jeu dynamique admettant T étapes. Une **stratégie pure pour le joueur** i est définie par une suite de T applications S_i^t de H^t vers $A_i(H^t)$. En d'autres termes, une stratégie pure est une suite de règles de sélection d'une action particulière par le joueur i à chaque étape du jeu compte tenu de l'histoire qui s'est déroulée jusqu'alors. Par exemple, les actions du joueur i à l'étape 0 sont $a_i^0 = s_i^0(h^0)$, celles de l'étape 1 sont $a_i^1 = s_i^1(h^1)$, de l'étape 2 $a_i^2 = s_i^2(h^2)$, et ainsi de suite.

Nous donnerons un exemple de jeu dynamique en Section 1.3.4 (page 27).

1.1.2 Jeux coopératifs et non coopératifs

Comme nous l'avons vu, une caractéristique fondamentale des jeux est que le gain obtenu par un joueur dépend de ses choix, mais aussi des choix effectués par les autres joueurs. Il convient alors de distinguer deux grandes familles de jeux : les jeux coopératifs et les jeux non coopératifs.

Un jeu est **coopératif** lorsque les joueurs peuvent passer entre eux des accords qui les lient de manière contraignante. On dit alors qu'ils forment une **coalition** dont les membres agissent de concert. Dans le cas contraire, c'est-à-dire lorsque les joueurs n'ont pas la possibilité de former des coalitions, le jeu est **non coopératif**.

1.1.2.1 Jeux coopératifs

Un jeu coopératif (appelé aussi jeu coalitionnel) est un jeu dans lequel les joueurs peuvent former des coalitions et agir de concert.

[3]Les jeux de cartes sont généralement des jeux à information imparfaite : à la belote par exemple, un joueur ne connaît pas toutes les stratégies possibles de ses adversaires, il n'a pas une connaissance parfaite du jeu.

[4]La description des ensembles $A_i(h^t)$ à chaque étape t pour tout joueur i fait partie de la spécification des règles du jeu.

Définition 1.2. *Une* **coalition** *est un sous-ensemble de joueurs : $C \subseteq N = \{1, \dots, N\}$. Si C est une coalition d'un seul joueur ($C = \{i\}$), C est appelé* **singleton**. *Si C est la coalition formée de tous les joueurs ($C = N$), C est appelé* **grande coalition**.

On dit qu'un jeu coopératif est à **utilité non transférable** s'il n'est pas possible d'additionner les utilités des joueurs et de les redistribuer aux membres d'une coalition. Chaque membre d'une coalition essaie d'optimiser le montant obtenu par chacun individuellement.

Définition 1.3. *Un* **jeu coopératif à utilité non transférable** *est une paire (N, v) où*

- ∗ *N est un ensemble de joueurs*
- ∗ *v est une* **fonction caractéristique** *qui associe un vecteur $v(C) \in \mathbb{R}^C$ à chaque coalition C de N, chaque élément v_i du vecteur $v(C)$ correspondant à l'utilité obtenue par le joueur i dans la coalition C.*

La bataille des sexes peut être transformée en un jeu coopératif à utilité non transferable [Luce et Raiffa, 1957] :

Exemple 1.2 (page 11) – suite : *La bataille des sexes peut être formalisée par le jeu coopératif à utilité non transférable à 2 joueurs suivant :*

- ∗ $N = \{1, 2\}$,
- ∗ *v définie par :* $\begin{cases} v(12) = \{(v_1, v_2); 1 \leq v_1 \leq 2, 1 \leq v_2 \leq 2\} \\ v(1) = \{(v_1); 0 \leq v_1 \leq 2\} \\ v(2) = \{(v_2); 0 \leq v_2 \leq 2\} \end{cases}$

En effet, si Lucas et Elsa se coordonnent, ils sont sûrs d'aller au même endroit, et donc auront chacun au moins une utilité de 1. Par contre, s'ils ne se coordonnent pas, ils ont toutes les chances de ne pas être au même endroit, et donc de n'en retirer aucune satisfaction (même s'ils peuvent avoir de la chance et que l'un des deux se dévoue, en espérant que l'autre n'ait pas eu la même idée).

On dit qu'un jeu coopératif est à **utilité transférable** s'il est possible d'additionner les utilités des joueurs et de les redistribuer aux membres d'une coalition (il existe une "monnaie" commune à tous avec laquelle on peut effectuer des transferts).

Définition 1.4. *Un* **jeu coopératif à utilité transférable** *est une paire (N, v) où*

- ∗ *N est un ensemble de joueurs*
- ∗ *v est une* **fonction caractéristique** *qui associe une valeur $v(C) \in \mathbb{R}$ à chaque coalition C de N.*

Pour chaque coalition C, $v(C)$ est le paiement total que peuvent se partager les joueurs appartenant à C, indépendamment du comportement des joueurs n'appartenant pas à C.

Un jeu coopératif à utilité transférable est :

- ∗ **symétrique** si la valeur d'une coalition ne dépend que de sa taille : il existe une fonction f telle que $\forall C \subseteq N$, $v(C) = f(|C|)$;
- ∗ **monotone** si $B \subseteq C \Rightarrow v(B) \leq v(C)$;
- ∗ **superadditif** si $B \cap C = \varnothing \Rightarrow v(B \cup C) \geq v(B) + v(C)$;

* **simple** si pour toute coalition C, soit $v(C) = 1$[5] (**coalition gagnante**), soit $v(C) = 0$ (**coalition perdante**), et $v(N) = 1$.

Un joueur i dans un jeu coopératif à utilité transférable :

* a **un droit de veto** s'il appartient à toutes les coalitions gagnantes ($v(C) = 1 \Rightarrow i \in C$) ;
* est un **dictateur** si une coalition est gagnante si et seulement si il en fait partie ($v(C) = 1 \Leftrightarrow i \in C$).

Exemple 1.3. *Présentons ici plusieurs jeux coopératifs à utilité transférable à 3 joueurs satisfaisant différences propriétés*[6] *:*

* Majorité simple. *Une coalition est gagnante si et seulement si elle comprend au moins deux membres*
$$\Rightarrow \begin{cases} v(1) = v(2) = v(3) = 0 \\ v(1,2) = v(2,3) = v(1,3) = v(1,2,3) = 1 \end{cases}$$
* Unanimité. *Une coalition est gagnante si et seulement si elle comprend tous les membres*
$$\Rightarrow \begin{cases} v(1,2,3) = 1 \\ \forall C \subset N, v(C) = 0 \end{cases}$$
* Le joueur 2 a un droit de veto. *Une coalition est gagnante si et seulement si elle comprend au moins deux membres, dont le joueur 2 : 2 peut empêcher une coalition de gagner, mais ne peut pas pour autant gagner seul*
$$\Rightarrow \begin{cases} v(1) = v(2) = v(3) = v(1,3) = 0 \\ v(1,2) = v(2,3) = v(1,2,3) = 1 \end{cases}$$
* Le joueur 2 est dictateur. *Une coalition est gagnante si et seulement si elle comprend le joueur 2*
$$\Rightarrow \begin{cases} v(1) = v(3) = v(1,3) = 0 \\ v(2) = v(1,2) = v(2,3) = v(1,2,3) = 1 \end{cases}$$

1.1.2.2 Jeux non coopératifs

Les jeux non coopératifs se divisent en deux grandes familles : les jeux à somme nulle, et ceux à somme non nulle. En économie, cette notion simplificatrice de jeu à somme nulle est importante : ces jeux correspondent à l'absence de production, ou de destruction, de produits.

Les **jeux à somme nulle** sont tous les jeux où la somme "algébrique" des gains des joueurs est constante : ce que gagne l'un est nécessairement perdu par un autre. Stricto sensu, il est possible que les jeux ne soient pas à somme nulle, mais à somme constante, et cela n'a aucune importance en pratique : l'enjeu est de répartir entre tous les joueurs un total de gains préalablement fixé. Les échecs, le poker ou encore le jeu **matching pennies**, présenté ci-dessous, sont des jeux à somme nulle, les gains d'un joueur étant très exactement les pertes d'un autre joueur, tandis que le dilemme du prisonnier est un jeu non coopératif à somme non nulle[7].

Exemple 1.4. *Deux joueurs, Robin et Annelise, annoncent simultanément pile ou face.*

* *Si les annonces sont identiques, Robin donne 15 euros à Annelise.*

[5] La valeur 1 est choisie arbitrairement, il faut juste que ce soit la même pour toutes les coalitions gagnantes.

[6] Par souci de simplifier les notations, on écrira $v(i, j)$ au lieu de $v(\{i, j\})$.

[7] Les deux prisonniers sont enfermés dans des cellules séparées, ne peuvent pas communiquer, et donc ne peuvent pas passer un accord et former une coalition, mais ce que gagne l'un n'est pas forcément perdu par l'autre.

* *Si les annonces ne concordent pas, c'est Annelise qui doit donner 15 euros à Robin.*

On peut formaliser cette situation par un jeu à 2 joueurs, chaque joueur ayant 2 stratégies possibles : annoncer pile (dénotée par P) ou face (dénotée par F).

* $N = \{1,2\}$,
* *Annelise* (1) *a 2 stratégies possibles :* $s_{1_1} = P$ *et* $s_{1_2} = F$. *Robin* (2) *a les même possibilités :* $s_{2_1} = P$ *et* $s_{2_2} = F$.
* *On peut donc calculer les utilités de chacun des joueurs pour chacune des issues possibles du jeu :*
  * $u_1(P,P) = u_1(F,F) = u_2(P,F) = u_2(F,P) = 15$, *et*
  * $u_1(P,F) = u_1(F,P) = u_2(P,P) = u_2(F,F) = -15$.

Ce jeu est bien un jeu à somme nulle : tout ce qui est gagné par Annelise est perdu par Robin, et vice-versa.

En 1944, John von Neumann et Oskar Morgenstern [von Newmann et Morgenstern, 1944] ont démontré que tout jeu à somme nulle à n joueurs est une forme généralisée des jeux à somme nulle à 2 joueurs, et qu'il est possible de ramener tout jeu à somme non nulle à n joueurs à un jeu à somme nulle à $n+1$ joueurs, le $n+1^{eme}$ joueur représentant le gain ou la perte globale.

Les jeux à somme nulle à 2 joueurs constituent donc une partie essentielle de la théorie mathématique des jeux.

On peut noter ici que si les jeux à somme nulle et à deux joueurs sont des jeux non coopératifs, les jeux à somme nulle et à n joueurs peuvent être coopératifs, si par exemple $n-1$ joueurs se liguent contre le n^{eme} joueur pour le faire perdre, et se partager les gains.

Exemple 1.5. *Tristan, Aguirre et Matisse jouent à Pile ou Face. Si deux d'entre eux annoncent la même chose, ils gagnent et le troisième perd. Si les trois ont la même annonce, aucun d'entre eux ne gagne.*

On peut formaliser cette situation par le jeu à somme nulle et à utilités non transférables à 3 joueurs suivant :

* $N = \{1,2,3\}$,
* *Tristan* (1) *a 2 stratégies possibles :* $s_{1_1} = P$ *et* $s_{1_2} = F$. *Aguirre* (2) *et Matisse* (3) *ont les mêmes possibilités :* $s_{2_1} = s_{3_1} = P$ *et* $s_{2_2} = s_{3_2} = F$.
* *On peut donc calculer les utilités de chacun des joueurs pour chacune des issues possibles du jeu :*
  * $u_1(P,P,P) = u_1(F,F,F) = u_2(P,P,P) = u_2(F,F,F) = u_3(P,P,P) = u_3(F,F,F) = 0$,
  * $u_1(P,P,F) = u_2(P,P,F) = u_1(P,F,P) = u_3(P,F,P) = u_2(F,P,P) = u_3(F,P,P) = u_1(F,F,P) = u_2(F,F,P) = u_1(F,P,F) = u_3(F,P,F) = u_2(P,F,F) = u_3(P,F,F) = 1$.
  * $u_3(P,P,F) = u_2(P,F,P) = u_1(F,P,P) = u_3(F,F,P) = u_2(F,P,F) = u_1(P,F,F) = -2$.

Ce jeu peut être vu comme un jeu coopératif, 2 joueurs ont intéret à se mettre d'accord pour avoir une chance de battre le 3^{eme}. *Il est à utilités non transférables car si un joueur gagne, il ne peut pas donner une partie de sa victoire, ou de ce qu'il a gagné, à un autre joueur.*

Ce jeu peut aussi être vu comme un jeu à somme nulle et à utilités transférables si on suppose par exemple que Tristan, Aguirre et Matisse misent à l'origine 25 euros chacun ; que s'ils ont tous les trois la même annonce ils récupèrent leur mise, mais si l'annonce d'un joueur est différente de celle des deux autres, il perd sa mise tandis que les deux autres gagnent 37.5 euros chacun.

1.1.3 Récapitulatif

Nous avons donc vu dans cette section une taxonomie de quelques jeux, que nous récapitulons dans le tableau représenté Figure 1.1, dans lequel nous avons placé les quelques exemples déjà étudiés.

	Somme nulle			Somme non nulle		
	¬Coop.	Coop.		¬Coop.	Coop.	
		Util. transf.	Util. ¬transf.		Util. transf.	Util. ¬transf.
Statique	Ex. 1.4	Ex. 1.5	Ex. 1.5	Ex. 1.1	Ex. 1.3	Ex. 1.2
Dynamique				Ex. 1.12		

Figure 1.1 — Taxonomie des jeux

Comme c'est visible sur ce tableau, certains cas ne sont pas illustrés par des exemples. Ces derniers concernent les jeux dynamiques, que nous n'étudierons pas dans la suite de ce rapport. Nous avons donc omis de présenter exhaustivement cette catégorie de jeux.

Nous allons à présent voir comment l'on peut représenter ces jeux.

1.2 Représentation des jeux

Un jeu stratégique peut être représenté de deux façons différentes mais équivalentes : sous forme normale (dite aussi stratégique) et sous forme extensive (dite aussi développée).

1.2.1 Forme extensive

Un jeu sous **forme extensive** est défini par un **arbre** de décision décrivant les actions possibles des joueurs à chaque étape du jeu, la séquence de tours de jeu des joueurs ainsi que l'information dont ils disposent à chaque étape pour prendre leur décision. Chaque nœud de l'arbre spécifie le joueur qui doit choisir une action (ou stratégie) à ce moment du jeu, ainsi que l'information dont il dispose. Les gains que chaque joueur peut réaliser après avoir suivi un des chemins possibles au sein de l'arbre, correspondant à chaque profil de stratégies, sont associés à chaque feuille de l'arbre. Par exemple, la forme extensive du dilemme du prisonnier est représentée Figure 1.2 (page suivante).

Exemple 1.1 (page 10) – suite : *Une forme extensive du dilemme du prisonnier est la suivante :*

Chaque nœud de cet arbre correspond à un joueur, et chaque branche partant de ce nœud correspond à une stratégie possible de ce joueur.
Dans les feuilles, le premier élément de chaque couple représente l'utilité du prisonnier 1, tandis que le second élément représente celle du prisonnier 2.
Le cercle en pointillé entourant les deux occurrences du joueur 2 signifie que ce dernier ne sait pas dans quelle situation il se trouve, il ne sait pas si son complice a choisi d'avouer ou de se taire.

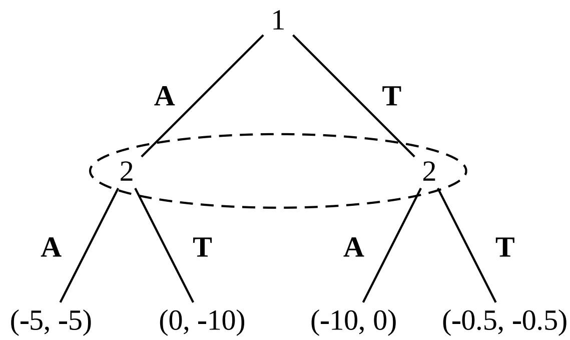


***Figure 1.2* —** Forme extensive du dilemme du prisonnier

Ce jeu a une seconde forme extensive, plaçant le second joueur en haut de l'arbre. Ces deux représentations sont équivalentes car les deux joueurs jouent simultanément.

La forme extensive permet une description "dynamique" du jeu parce qu'elle spécifie les séquences de décisions prises par les joueurs. Pourtant, lorsqu'un jeu implique de multiples joueurs et de multiples choix, l'arbre peut devenir complexe à représenter. Or, lorsque les stratégies des agents sont choisies en parallèle, c'est-à-dire sans que l'un des joueurs observe les décisions des autres, comme c'est le cas dans le dilemme du prisonnier, cette construction basée sur un modèle dynamique n'est pas nécessaire. Dans ce cas, on simplifiera la présentation du jeu en utilisant une représentation sous forme normale, absolument équivalente à la forme extensive associée.

1.2.2 Forme normale

Un jeu sous **forme normale** est la donnée de l'ensemble des joueurs, de l'ensemble des stratégies pour chaque joueur et des paiements associés à toute combinaison possible de stratégies. On peut alors représenter ces jeux **sous forme matricielle**, en associant à chaque profil de stratégies s un n-uplet donnant l'utilité obtenue par chaque joueur dans l'ordre : $(u_1(s), u_2(s), \ldots, u_n(s))$. Par exemple, la forme normale du dilemme du prisonnier est représentée Figure 1.3.

Exemple 1.1 (page 10) – suite : *La forme normale du dilemme du prisonnier est représentée Figure 1.3.*

1 \ 2	A	T
A	(-5, -5)	(0, -10)
T	(-10, 0)	(-0.5, -0.5)

***Figure 1.3* —** Forme normale du dilemme du prisonnier

Comme précédemment, le premier élément de chaque couple représente l'utilité du prisonnier 1, tandis que le second élément représente celle du prisonnier 2.

Cette représentation sous forme matricielle permet de représenter des jeux ayant un nombre de joueurs et un nombre de stratégies pour chaque joueur raisonnable : la taille de la matrice est exponentielle en fonction du nombre d'agents, et du nombre de choix possible pour chaque agent. Par exemple, si n agents ont chacun le choix entre deux actions possibles, il faudra spécifier $n \times 2^n$ valeurs numériques. Ces descriptions, sous forme extensive ou normale, supposent que les ensembles d'actions sont finis.

A chaque jeu sous forme extensive correspond un jeu sous forme normale dans lequel les joueurs choisissent simultanément les stratégies qu'ils mettront en œuvre. En revanche, un jeu sous forme normale peut correspondre à plusieurs jeux sous forme extensive différente, comme nous l'avons vu pour le dilemme du prisonnier.

1.3 Concepts de solution

Etudions un autre exemple classique de jeu statique, qui suit la même logique que le dilemme du prisonnier, le **jeu de la tirelire** :

Exemple 1.6. *On propose à Jérémie et Léonore le jeu suivant : ils ont chacun la possibilité de mettre 0 ou 100 euros dans une tirelire. Une fois qu'ils ont tous deux pris une décision, sans connaître la décision de l'autre, le contenu de la tirelire est multiplié par 1.5 et est réparti en part égale entre les deux joueurs.*

Une fois ce jeu formalisé, si on choisit Jérémie comme étant le joueur 1, Léonore le joueur 2, on obtient la forme normale représentée Figure 1.4.

1 \ 2	0	100
0	(0, 0)	(75, -25)
100	(-25, 75)	(50, 50)

Figure 1.4 — Forme normale du jeu de la tirelire

Dans ce cas de figure, que vont faire Jérémie et Léonore ? Mettons nous à la place de Jérémie, qui tient le raisonnement suivant : " Si Léonore ne met rien dans la tirelire, il est optimal pour moi que je ne mette rien, sinon je perdrai 25 euros. Si elle dépose 100 euros, je gagne 75 euros si je ne mets rien, et 50 si je mets 100 euros. Donc, dans les deux cas de figure, j'ai intérêt à ne rien mettre dans la tirelire." Si Léonore tient le même raisonnement, le résultat sera un gain nul pour chacun d'entre eux. Même s'ils se mettent d'accord au début du jeu pour mettre tous les deux 100 euros, il reste "optimal" de ne rien mettre dans la tirelire s'ils sont motivés par la recherche de leur seul intérêt personnel.

La logique qui se trouve derrière ce jeu, et qui est la même que celle se trouvant derrière le dilemme du prisonnier, montre qu'un groupe d'individus ne va pas nécessairement se comporter dans l'intérêt du groupe si chacun peut obtenir pour lui-même un résultat meilleur en choisissant pour son propre compte. Sachant cela, nous allons à présent présenter quelques concepts permettant de prédire l'issue d'un jeu (ce que l'on appelle des "concepts de solution").

L'analyse d'un jeu permet de prédire l'**équilibre** qui émergera si les joueurs sont rationnels. Par équilibre, nous entendons un état ou une situation dans lequel aucun joueur ne souhaite modifier son comportement compte tenu du comportement des autres joueurs. De façon plus précise, un équilibre est une combinaison de stratégies telle qu'aucun des joueurs n'a d'intérêt à changer sa stratégie compte tenu des stratégies des autres joueurs. Une fois que l'équilibre a été atteint dans un jeu (et peu importe la manière dont il a été obtenu), il n'y a aucune raison de le quitter.

Nous allons présenter ici plusieurs concepts de solution : les deux premiers, équilibres de Nash et stratégies dominées, s'appliquent à des jeux statiques non coopératifs ; le concept de noyau (core) s'applique à des jeux statiques coopératifs, et enfin les équilibres parfaits de Selten pour les jeux dynamiques.

1.3.1 Equilibres de Nash

L'**équilibre de Nash**, introduit par John Nash en 1950 [Nash, 1950], est un concept fondamental en théorie des jeux. Il décrit une issue du jeu dans laquelle aucun joueur ne souhaite modifier sa stratégie étant donnée la stratégie de chacun de ses rivaux.

Les jeux pour lesquels il est possible de calculer les équilibres de Nash sont représentés Figure 1.5.

	Somme nulle			Somme non nulle		
	¬Coop.	Coop.		¬Coop.	Coop.	
		Util. transf.	Util. ¬transf.		Util. transf.	Util. ¬transf.
Statique	Nash			Nash		
Dynamique						

Figure 1.5 **—** Taxonomie des jeux : Domaine d'application des équilibres de Nash

1.3.1.1 Equilibre de Nash en stratégies pures

Définition 1.5. *Soit G un jeu non coopératif à n joueurs, avec $N = \{1, \ldots, n\}$ l'ensemble de joueurs. Un profil de stratégie $s = (s_1, s_2, \ldots, s_n)$ est un* **équilibre de Nash en stratégies pures (PNE)** *si et seulement si*

$$\forall i \in N, \forall s_i' \in S_i, u_i(s) \geq u_i(s_i', s_{-i})$$

En d'autres termes, un équilibre de Nash est un profil de stratégie dont aucun joueur n'a intérêt à dévier s'il suppose que les autres joueurs ne dévieront pas non plus.

Exemple 1.1 (page 10) – suite : *Le profil de stratégies AA est un équilibre de Nash en stratégies pures du dilemme du prisonnier. En effet, on peut vérifier dans la matrice des paiements représentée Figure 1.3 (page 17) que l'on a : $u_1(AA) \geq u_1(TA)$ et $u_2(AA) \geq u_2(AT)$.*

AA est le seul équilibre de Nash de ce jeu. En effet, AT ne peut pas être un PNE car AA est une meilleure stratégie pour le joueur 2 : $u_2(AA) > u_2(AT)$; TA non plus car AA est une meilleure stratégie pour le joueur 1 : $u_1(AA) > u_1(TA)$; et de même pour TT car AT est une meilleure stratégie pour le joueur 1 : $u_1(TT) < u_1(AT)$.

Ce jeu n'a qu'un équilibre de Nash en stratégies pures. Pourtant, cette unicité n'est pas toujours garantie :

Exemple 1.2 (page 11) – suite : *Le jeu de la bataille des sexes a deux équilibres de Nash en stratégies pures : HH et RR. En effet, on peut vérifier dans la matrice des paiements représentée Figure 1.6 que l'on a :* $u_1(H,H) \geq u_1(R,H)$, $u_2(H,H) \geq u_2(H,R)$ *;* $u_1(R,R) \geq u_1(H,R)$ *et* $u_2(R,R) \geq u_2(R,H)$.

1 \ 2	H	R
H	(2, 1)	(0, 0)
R	(0, 0)	(1, 2)

Figure 1.6 **—** Forme normale de la bataille des sexes

De la même façon, l'existence d'un équilibre de Nash en stratégies pures n'est pas garantie non plus, comme on peut le constater facilement dans le jeu matching pennies de l'exemple 1.4 (page 14).

1.3.1.2 Equilibres de Nash en stratégies mixtes

Dans ce genre de situation, il existe une autre méthode pour obtenir un équilibre de Nash, et donc un équilibre : il est possible d'élargir la définition d'une stratégie, et d'y inclure non seulement les actions pures (telles qu'annoncer pile ou face), mais aussi les probabilités de choisir l'une ou l'autre de ces actions. Chaque joueur associe une probabilité p_i (positive ou nulle) à la stratégie s_i, et vise à maximiser ses gains espérés en choisissant la meilleure loterie possible, c'est-à-dire la meilleure **stratégie mixte**. Une stratégie mixte est donc une stratégie définissant les probabilités avec lesquelles les joueurs choisissent chacune de leurs stratégies pures.

Définition 1.6. *Une* **stratégie mixte** *pour le joueur i est une distribution de probabilité sur* S_i. $\Sigma_i = \Delta(S_i)$ *représente l'ensemble des stratégies mixtes du joueur i. La fonction* $\sigma_i : S_i \rightarrow \mathbb{R}$ *associe à la stratégie pure* s_i *sa probabilité d'être jouée.*

On peut noter ici qu'une stratégie pure s_i correspond à la stratégie mixte s_i associée à une probabilité de 1.

Définition 1.7. *Un* **équilibre de Nash en stratégies mixtes** *est un profil de stratégies mixtes* $\sigma \in \Sigma$ *tel que*

$$\forall i \in N, \forall \sigma_i' \in \Sigma_i, u_i(\sigma) \geq u_i(\sigma_i', \sigma_{-i})$$

Un équilibre en stratégies mixtes est donc une situation dans laquelle tous les joueurs choisissent leur stratégies mixtes de façon à rendre leurs adversaires indifférents entre les gains espérés de chacune de leurs stratégies pures.

Exemple 1.4 (page 14) – suite : *Le jeu matching pennies n'a aucun équilibre de Nash en stratégies pures.*

Supposons que la distribution de probabilités pour Annelise est la suivante : $\sigma_1(P) = x$ *et* $\sigma_1(F) = 1-x$ *; et que celle de Robin est* $\sigma_2(P) = y$ *et* $\sigma_2(F) = 1-y$*.*

Dans ce cas, si on suppose que Robin choisit $\sigma_2(P) = 1$ *(il choisit toujours P), d'après la Figure 1.7 son gain espéré est de :* $15(1-x) - 15x = 15(1-2x)$*. De même, si Robin choisit* $\sigma_2(F) = 1$*, son gain espéré est de :* $15(2x-1)$*. On constate alors que si Annelise choisit une probabilité* $\sigma_1(P) = x = 1$ *(elle jouera toujours P), alors Robin a tout intérêt de choisir toujours* $\sigma_2(F) = 1$ *(*$y = 0$*). Par contre, si Robin choisit* $\sigma_2(F) = 1$*, Annelise choisira* $\sigma_1(F) = 1$ *(*$x = 0$*). Cette situation n'est donc pas un équilibre.*

En raisonnant de cette façon, on constate que pour ne pas influencer le choix de Robin, Annelise a donc tout intérêt à choisir $x = 1/2$*. Le gain espéré de Robin sera alors toujours* 0 *; et de même, Robin a intérêt à choisir* $y = 1/2$*.*

On vérifie que ce jeu est un équilibre de Nash en stratégies mixtes : si Annelise choisit $\sigma_1(P) = x = 1/2$*, l'utilité espérée de Robin est de* 0 *qu'il choisisse* $\sigma_2(P) = 1$ *(*$15(1-2x)$*) ou* $\sigma_2(F) = 1$ *(*$15(2x-1)$*). Annelise n'a donc pas intérêt à dévier. En effectuant le même raisonnement pour Robin, on constate que s'il choisit* $y = 1/2$*, l'utilité espérée d'Annelise est de* 0 *également, et Robin n'a pas non plus intérêt à dévier.*

Vérifions à présent que cet équilibre de Nash en stratégies mixtes est le seul : supposons que $y = 1/2$ *et* $x \neq 1/2$*. Dans ce cas, si* $x > 1/2$*, Robin a intérêt à choisir* $y = 0$*, et si* $x < 1/2$*, Robin choisira* $y = 1$*. Donc Annelise est sûre que Robin déviera. De même, si* $x = 1/2$ *et* $y \neq 1/2$*, Annelise aura tout intérêt à dévier.*

Ce jeu a donc un équilibre de Nash mixte : $\sigma_1(F) = \sigma_1(P) = 1/2$*,* $\sigma_2(F) = \sigma_2(P) = 1/2$*.*

1 \ 2	*P*	*F*
P	(15, -15)	(-15, 15)
F	(-15, 15)	(15, -15)

***Figure 1.7* —** Forme normale du jeu matching pennies

Quelques propriétés, classiques en théorie des jeux, découlent de ces définitions (voir par exemple [Osborne et Rubinstein, 1994; Hillas et Kohlberg, 2002]) :

- Tout équilibre de Nash en stratégies pures est un équilibre de Nash en stratégies mixtes.
- Tout jeu fini a au moins un équilibre de Nash en stratégies mixtes.

1.3.2 Stratégies dominées

Les jeux pour lesquels il est possible de calculer les stratégies dominées sont représentés Figure 1.8 (page suivante).

Comme le montre l'exemple suivant, qui présente le **jeu de la soirée**, les conditions d'existence d'un equilibre de Nash en stratégies pures sont parfois trop faibles.

	Somme nulle			Somme non nulle		
	$\neg$Coop.	Coop.		$\neg$Coop.	Coop.	
		Util. transf.	Util. $\neg$transf.		Util. transf.	Util. $\neg$transf.
Statique	Strat. Dom			Strat. Dom		
Dynamique						

***Figure 1.8* —** Taxonomie des jeux : Domaine d'application des stratégies dominées

Exemple 1.7. *Agathe et Iwan sont invités à une soirée. La raison principale qui les motive à y aller est de pouvoir se voir là-bas : si l'autre n'y va pas, il leur est indifférent d'y aller (dénoté par A) ou pas (dénoté par P).*

Une fois ce jeu formalisé, si on choisit Agathe comme étant le joueur 1 et Iwan le joueur 2, on obtient la forme normale représentée Figure 1.9.

1 \ 2	A	P
A	(1, 1)	(0, 0)
P	(0, 0)	(0, 0)

***Figure 1.9* —** Forme normale du jeu de la soirée

Ce jeu a 2 *équilibres de Nash en stratégies pures : AA et PP. Pourtant, un seul de ces équilibres est intéressant : si Agathe et Iwan veulent se voir, et qu'ils ont l'occasion de le faire à cette soirée, ils iront tous les deux.*

On voit sur cet exemple que l'un des équilibres de Nash en stratégies pures, PP, n'est pas une stratégie optimale. En effet, pour chacun des deux joueurs, la stratégie A permet toujours d'obtenir une utilité au moins aussi bonne que la stratégie P. Il semble alors assez naturel qu'Agathe et Iwan choisissent cette stratégie. On dit que A est une stratégie (faiblement) dominante pour chacun des deux joueurs de ce jeu.

Définissons les notions de stratégies strictement et faiblement dominées :

Définition 1.8. *La stratégie s_i du joueur i est dite* **strictement dominée** *s'il existe une autre stratégie s'_i telle que, quelles que soient les stratégies des autres joueurs, s'_i assure au joueur i une utilité strictement plus grande que s_i. Donc :*

$s_i \in S_i$ *est* **strictement dominée** *si*

$$\exists s'_i \in S_i \text{ telle que } \forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) < u_i(s'_i, s_{-i})$$

Définition 1.9. *La stratégie s_i du joueur i est dite* **faiblement dominée** *s'il existe une autre stratégie s'_i telle que, quelles que soient les stratégies des autres joueurs, s'_i assure au joueur i une utilité au* *moins aussi grande que s_i, et qu'il existe au moins une combinaison des stratégies des autres joueurs telle que l'utilité du joueur i avec s'_i soit strictement plus grande que celle avec s_i. Donc : $s_i \in S_i$ est* **faiblement dominée** *si $\exists s'_i \in S_i$ telle que*

$$\forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) \leq u_i(s'_i, s_{-i})$$

et que

$$\exists s'_{-i} \in S_{-i} \text{ telle que } u_i(s_i, s'_{-i}) < u_i(s'_i, s'_{-i})$$

Dans l'exemple 1.7 (page ci-contre), chaque joueur a une stratégie faiblement dominée (P) et, Agathe et Iwan étant deux joueurs rationnels, il est facile de voir qu'ils joueront leur stratégie dominante, à savoir A. Pourtant, l'existence d'une solution aussi simple est rare. Il est souvent nécessaire de faire appel à d'autres manières de raisonner dans l'espoir de trouver un équilibre au jeu. Par exemple, si le joueur 1 possède une stratégie strictement ou faiblement dominante, on peut s'attendre à ce qu'il choisisse cette stratégie. Comme le joueur 2 est capable d'anticiper ce choix, il choisit alors sa meilleure stratégie contre la stratégie dominante du premier.

Exemple 1.8. *Soit un jeu G à deux joueurs, chacun des joueurs ayant 3 stratégies possibles, représenté par la forme normale donnée Figure 1.10.*

1 \ 2	G	M	D
H	(4, 3)	(5, 1)	(6, 2)
M	(2, 1)	(8, 4)	(3, 6)
B	(3, 0)	(9, 6)	(2, 8)

Figure 1.10 **—** Forme normale du jeu présentant l'élimination des stratégies dominées

Il est tout d'abord possible de remarquer que M est une stratégie strictement dominée par D pour le joueur 2. Dans ce cas, il est possible de penser que le joueur 2 ne retiendra jamais cette stratégie. On peut donc éliminer la stratégie M de la matrice.

Ceci étant fait, on remarque alors que, dans la matrice résultante, H est "devenue" une stratégie strictement dominante pour le joueur 1. Dès lors, celui-ci devrait jouer H. Le joueur 2 étant rationnel, il est capable d'anticiper ce raisonnement, il sait donc qu'il est optimal pour lui de choisir G.

En conséquence, une fois que l'on a éliminé toute les stratégies dominées, on obtient un profil de stratégies résultat : HG.

Le processus d'élimination qui vient d'être appliqué dans cet exemple est appelé **processus d'élimination des stratégies dominées**. Il demande un comportement assez sophistiqué, dans la mesure ou chaque joueur doit être capable de reconstituer les opérations auxquelles les autres joueurs procèdent et d'en déduire de nouvelles implications pour lui-même.

Toutefois, le processus de dominance successive admet également des limites :

Exemple 1.9. *Soit le jeu G à deux joueurs représenté par la forme normale donnée Figure 1.11 (page suivante).*

1 \ 2	G	D
H	(3, 6)	(7, 1)
M	(5, 1)	(8, 0)
B	(6, 0)	(6, 2)

***Figure 1.11* —** Forme normale du jeu présentant les limites de l'élimination des stratégies dominées

Dans ce jeu, il est clair que H est strictement dominé par M pour le joueur 1*. On élimine donc la stratégie H. On ne peut pas aller plus loin car il n'y a plus de stratégie strictement ou faiblement dominée dans la matrice résultante. Cela revient à dire que, dans ce cas, le processus d'élimination des stratégies dominées ne conduit pas à un résultat unique.*

Nous pouvons énoncer ici quelques propriétés, classiques en théorie des jeux (voir par exemple [Osborne et Rubinstein, 1994; Hillas et Kohlberg, 2002]) :

- Comme nous l'avons vu dans l'exemple 1.9 (page précédente), le processus d'élimination des stratégies dominées ne conduit pas nécessairement à une solution unique ;
- L'ordre d'élimination des stratégies strictement dominées n'affecte pas le résultat final ;
- L'ordre d'élimination des stratégies faiblement dominées peut affecter le résultat final ;
- Une stratégie strictement dominée ne peut jamais être présente dans un équilibre de Nash en stratégies pures ;
- Une stratégie faiblement dominée peut apparaître dans un équilibre de Nash en stratégies pures.

On a pu constater que le concept de stratégies dominées pouvait conduire à des résultats alors que celui d'équilibre de Nash en stratégies pures aboutissait à une impasse. Mais l'inverse peut être vrai aussi : le processus d'élimination des stratégies dominées ne conduit pas forcément à un résultat, alors que l'on peut obtenir un équilibre de Nash, comme le montre l'exemple suivant :

Exemple 1.10. *Soit le jeu G à deux joueurs représenté par la forme normale donnée Figure 1.12.*

1 \ 2	A	B	C
D	(2, 3)	(1, 2)	(0, 2)
E	(1, 2)	(2, 3)	(1, 2)
F	(1, 1)	(3, 0)	(0, 2)

***Figure 1.12* —** Forme normale du jeu de l'exemple 1.10

Dans ce jeu, aucun joueur n'a de stratégie strictement ou faiblement dominée. Pourtant, il y a un équilibre de Nash : le profil de stratégies DA.

Etudions à présent des concepts de solution pour des jeux coopératifs (notion de cœur, appelé aussi noyau ou core).

1.3.3 Cœur

Les jeux pour lesquels il est possible de calculer le cœur sont représentés Figure 1.13.

	Somme nulle			Somme non nulle		
	¬Coop.	Coop.		¬Coop.	Coop.	
		Util. transf.	Util. ¬transf.		Util. transf.	Util. ¬transf.
Statique		cœur	cœur		cœur	cœur
Dynamique						

***Figure 1.13* —** Taxonomie des jeux : Domaine d'application du cœur

Le cœur, ou noyau, d'un jeu est un concept de solution pour les jeux coopératifs. Informellement, une issue d'un jeu appartient au cœur de ce jeu si aucune coalition ne peut améliorer le paiement (utilité) de *tous* ses membres, et donc aucune coalition n'a intérêt à dévier (c'est-à-dire changer sa stratégie commune).

On peut introduire ici deux définitions du cœur d'un jeu coalitionnel, une pour les jeux à utilités transférables, et une pour les jeux dont les utilités ne sont pas transférables :

1.3.3.1 Utilités transférables

On note $(x_i)_{i\in N}$ le **profil de paiement (ou répartition)** d'un jeu coopératif, x_i représentant le paiement obtenu par le joueur i. On note $x(C) = \sum_{i\in C} x_i$ la somme des paiements des membres de la coalition C (voir par exemple [Osborne et Rubinstein, 1994]).

On dit qu'un profil de paiement $(x_i)_{i\in N}$ est C-**réalisable** si $x(C) = v(C)$ (rappelons que la fonction v représente la fonction caractéristique du jeu, voir définition 1.4 (page 13)). Il est **réalisable** s'il est N-réalisable.

Définition 1.10. *Le* **cœur** *(ou* **noyau***) d'un jeu coalitionnel à utilités transférables* (N,v) *est l'ensemble des répartitions* $(x_i)_{i\in N}$ *réalisables telles que* $\forall C \subseteq N$, $x(C) \geq v(C)$, *ou, de manière équivalente, telles qu'il n'existe pas de coalition* C *et de répartition* C*-réalisable* $(y_i)_{i\in N}$ *où* $y_i > x_i$ *pour tout* $i \in C$.

Exemple 1.3 (page 14) – suite :

- ∗ Majorité simple. *Le profil de paiement doit respecter les équations suivantes :*

$$\begin{cases} \forall i, x_i \geq 0 \\ x_1 + x_2 + x_3 \geq 1 \text{ car on a ainsi } x(C) = x_1 + x_2 + x_3 \geq v(C) \\ x_1 + x_2 \geq 1 \\ x_1 + x_3 \geq 1 \\ x_2 + x_3 \geq 1 \end{cases} \Rightarrow \text{Cœur } = \varnothing$$

- ∗ Unanimité. *Le cœur réunit les profils de paiements tels que :*
  $\text{Cœur} = \{(x_1, x_2, x_3) : x_1 + x_2 + x_3 = 1, \forall i, x_i \geq 0\}$

* Le joueur 2 a un droit de veto. *Le profil de paiement doit respecter les équations suivantes :*
$$\begin{cases} \forall i, x_i \geq 0 \\ x_1+x_2+x_3 \geq 1 \\ x_1+x_2 \geq 1 \\ x_2+x_3 \geq 1 \end{cases} \Rightarrow \textit{Cœur} \ = \{(0,1,0)\}$$
* Le joueur 2 est dictateur. *Le profil de paiement doit respecter les équations suivantes :*
$$\begin{cases} \forall i, x_i \geq 0 \\ x_1+x_2+x_3 \geq 1 \\ x_1+x_2 \geq 1 \\ x_2+x_3 \geq 1 \\ x_2 \geq 1 \end{cases} \Rightarrow \textit{Cœur} \ = \{(0,1,0)\}$$

1.3.3.2 Utilités non transférables

Il existe plusieurs façons de définir le noyau d'un jeu coopératif à utilités non transférables : la première définition est semblable à celle du cœur pour les jeux avec utilités transférables. La seconde est la suivante :

Un profil de stratégies s est dans le noyau d'un jeu coalitionnel si et seulement si il n'existe pas de coalition C telle que tous les membres de cette coalition ont une stratégie commune qui leur permet à *tous* d'obtenir une meilleure utilité qu'avec s (voir par exemple [Aumann, 1967; Owen, 1982; Myerson, 1991]).

Définition 1.11. *Le* **cœur** *(ou* **noyau***) d'un jeu coalitionnel à utilités non transférables est l'ensemble des profils de stratégies $s = (s_1, \ldots, s_n)$ tels qu'il n'existe pas de coalition $C \subset N$ et de $s_C \in S_C$ tels que $\forall i \in C$, $\forall s_{-C} \in S_{-C}$, $u_i(s_C, s_{-C}) > u_i(s)$.*

Exemple 1.11. *Soit le jeu G à trois joueurs représenté par la forme normale donnée Figure 1.14.*

3 : E		
1 \ 2	C	D
A	(3, 3, 3)	(4, 1, 0)
B	(5, 0, 0)	(2, 5, 2)

3 : F		
1 \ 2	C	D
A	(3, 2, 1)	(1, 3, 2)
B	(1, 0, 1)	(4, 2, 1)

***Figure 1.14* —** Forme normale d'un jeu coopératif à utilités non transférables

On a ici : Cœur = $\{(ACE)\}$. En effet, il n'existe pas de coalition qui permette à tous les joueurs de cette coalition d'obtenir une meilleure utilité :

* $C = \{1\}$. *Si* 1 *dévie et joue B, il existe un* $s_{-C} = DE$ *tel que* $u_1(ACE) > u_1(BDE)$.
* $C = \{2\}$. *Si* 2 *dévie et joue D, il existe un* $s_{-C} = AE$ *tel que* $u_1(ACE) > u_1(ADE)$.
* $C = \{3\}$. *Si* 3 *dévie et joue F, il existe un* $s_{-C} = AC$ *tel que* $u_1(ACE) > u_1(ACF)$.
* *On raisonne de même pour* $C = \{1,2\}$, $C = \{1,3\}$, $C = \{2,3\}$ *et* $C = \{1,2,3\}$.

1.3.4 Equilibres parfaits de Selten

Les jeux pour lesquels il est possible de calculer l'équilibre parfait de Selten sont représentés Figure 1.15.

	Somme nulle			Somme non nulle		
	¬Coop.	Coop.		¬Coop.	Coop.	
		Util. transf.	Util. ¬transf.		Util. transf.	Util. ¬transf.
Statique						
Dynamique	Selten	Selten	Selten	Selten	Selten	Selten

***Figure 1.15* —** Taxonomie des jeux : Domaine d'application de l'équilibre parfait de Selten

Exemple 1.12. *Soit le jeu dynamique en information parfaite représenté sous forme extensive par la Figure 1.16.*

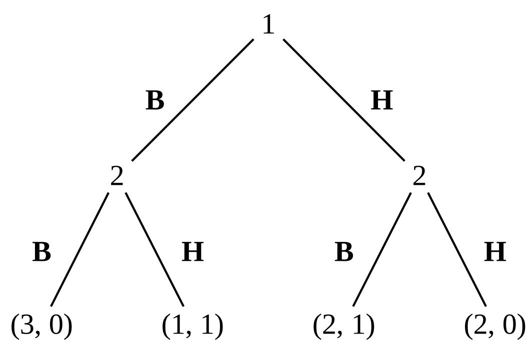


***Figure 1.16* —** Forme extensive d'un jeu dynamique en information parfaite

Le joueur 2 *a deux stratégies possibles : H et B. Le joueur* 2*, qui joue après* 1*, a* 2 *stratégies possibles selon le choix de* 1 *:* $s_2 = [s_2(H), s_2(B)]$.

Comment doit-on jouer ce jeu ? Le principe consiste à raisonner en remontant par **induction vers l'amont** (backward induction) : on raisonne en sens contraire de la manière dont le jeu va effectivement se dérouler.

Dans l'exemple 1.12, le joueur 1 se dit "si je joue H, alors 2 va jouer B de sorte que mon gain sera égal à 2 ; si je joue B, 2 jouera H et mon gain sera de 1". Dès lors, si 1 suppose que 2 choisira sa meilleure réponse à la seconde étape, il choisira H à la première. On admet que lorsque c'est au tour du joueur 2 de jouer, il choisit la meilleure réponse pour lui-même, à savoir $s_2(H) = B$ et $s_2(B) = H$. En conséquence, sachant que le joueur 2 se comporte de manière optimale à la seconde étape, le joueur 1 intègre cette information, et choisit une action optimale conditionnellement au comportement optimal de 2.

A chaque étape t, les stratégies du sous-jeu sont définies comme celles du jeu initial. La seule différence est que l'histoire à considérer pour les étapes précédentes est donnée par h^t : $s_i|h^t$ est la restriction de s_i imposée par l'histoire h^t. On dit alors que $s = (s_1, s_2, \ldots, s_n)$ est un **équilibre parfait de Selten** (ou subgame perfect Nash equilibrium, [Selten, 1965]) si, pour toute histoire h^t, la restriction $s|h^t$ est un équilibre de Nash du sous-jeu $G(h^t)$.

Exemple 1.13. *Soit le jeu dynamique représenté sous forme extensive par la Figure 1.17.*

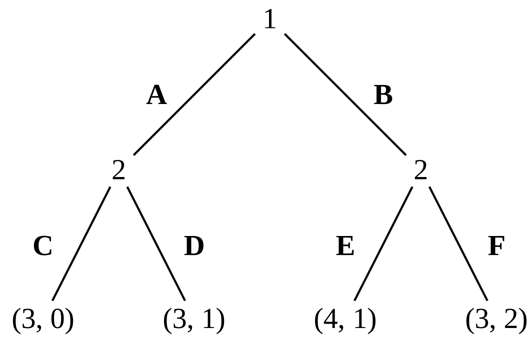
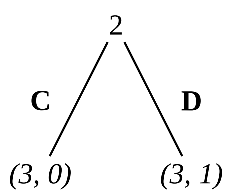


Figure 1.17 **—** Forme extensive d'un jeu dynamique

Ce jeu sous forme extensive a trois sous-jeux : le premier est le jeu lui-même, puisque tout jeu est un sous-jeu, le second commence après A, et le dernier après B.

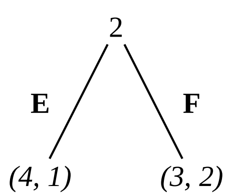
La forme extensive du sous-jeu commençant après A , et sa forme normale associée sont les suivantes :



C	*D*
(3, 0)	*(3, 1)*

Ce sous-jeu a un équilibre de Nash : D, que l'on notera $D(A)$, car 2 *jouera D si* 1 *joue A auparavant.*

La forme extensive du sous-jeu commençant après B , et sa forme normale associée sont les suivantes :



E	*F*
(4, 1)	*(3, 2)*

Ce sous-jeu a un équilibre de Nash : $F(B)$.

Dans le jeu complet, le joueur 1 *a* 2 *stratégies possibles : A et B. Par contre, le joueur* 2 *en a* 4 *: $C(A)E(B)$ qui signifie que* 2 *joue C si* 1 *joue A, et E si* 1 *joue B, $C(A)F(B)$, $D(A)E(B)$ et $D(A)F(B)$. La forme normale de ce jeu est :*

1 \ 2	$C(A)E(B)$	$C(A)F(B)$	$D(A)E(B)$	$D(A)F(B)$
A	*(3, 0)*	*(3, 0)*	*(3, 1)*	*(3, 1)*
B	*(4, 1)*	*(3, 2)*	*(4, 1)*	*(3, 2)*

Ce jeu a 3 *équilibres de Nash :*

* $(A, D(A)F(B))$ *est un équilibre parfait de Selten :* $D(A)$ *et* $F(B)$ *sont des équilibres de Nash des sous-jeux associés.*
* $(B, C(A)F(B))$ *n'est pas un équilibre parfait de Selten :* $C(A)$ *ne sera pas joué dans le sous jeu commençant après A, ce n'est donc pas un équilibre de Nash de ce sous-jeu.*
* $(B, D(A)F(B))$ *est un équilibre parfait de Selten :* $D(A)$ *et* $F(B)$ *sont des équilibres de Nash des sous-jeux associés.*

Ce jeu a donc 2 *équilibres parfaits de Selten :* $(A, D(A)F(B))$ *et* $(B, D(A)F(B))$.

1.3.5 Récapitulatif

Nous pouvons à présent récapituler les concepts de jeu que nous avons vu en fonction des types de jeux auxquels ils peuvent être appliqués dans le tableau représenté Figure 1.18.

	Somme nulle			Somme non nulle		
	¬Coop.	Coop.		¬Coop.	Coop.	
		Util. transf.	Util. ¬transf.		Util. transf.	Util. ¬transf.
Statique	Nash / Dom	cœur	cœur	Nash / Dom	cœur	cœur
Dynamique	Selten	Selten	Selten	Selten	Selten	Selten

Figure 1.18 **—** Taxonomie des jeux : Domaine d'application des concepts de solution

2 Jeux booléens

Comme nous l'avons vu dans la section 1.2 (page 16), les deux modes de représentation usuels des jeux (forme extensive et forme normale) coïncident pour les jeux statiques. Cette représentation ne fait pas l'économie de la description explicite de la fonction d'utilité de chaque agent. Or, cette description est de taille exponentielle en fonction du nombre d'agents : par exemple, si n agents ont chacun un choix entre deux actions possibles, il faudra spécifier $n \times 2^n$ valeurs numériques ; si deux agents contrôlent chacun un ensemble de p variables booléennes (il suffit de penser à de telles variables comme à des boutons que l'agent peut choisir d'enfoncer ou non), chaque agent a 2^p stratégies possibles et il faudra donc expliciter $2 \times (2^p)^2 = 2^{2p+1}$ valeurs numériques.

Cette explosion combinatoire est encore plus flagrante lorsqu'à la fois l'ensemble des agents et l'ensemble des stratégies pour chacun des agents sont de taille importante. Il devient alors déraisonnable de spécifier les fonctions d'utilité de manière explicite, en listant les valeurs pour chaque combinaison de stratégies. Il est tout aussi déraisonnable de penser pouvoir calculer des propriétés du jeu en appliquant un algorithme nécessitant une énumération explicite des combinaisons de stratégies. Pensons par exemple au calcul des équilibres de Nash en stratégies pures : ce calcul nécessite, dans le cas des jeux précédents, et dans le pire des cas, un temps de calcul de l'ordre de $n \times 2^n$ (pour le jeu à n joueurs avec 2 actions chacun) et de $2 \times 2^p \times 2^{2p} = 2^{3p+1}$ (pour le jeu à deux joueurs contrôlant chacun p variables booléennes).

D'un autre côté, une sous-branche de l'intelligence artificielle s'intéresse aux langages de représentation compacte de préférences (ordinales ou numériques). Ces langages permettent une représentation concise de relations de préférences, ou de fonctions d'utilité, sur un ensemble de conséquences qui possède une structure combinatoire (c'est-à-dire un produit cartésien de domaines de valeurs finis pour un ensemble fini de variables), en exploitant dans une large mesure des propriétés structurelles des relations de préférences (comme l'indépendance préférentielle entre variables). En particulier, lorsque les variables en jeu sont binaires, ces langages sont fondés sur la logique propositionnelle, dont ils héritent l'expressivité et les méthodes algorithmiques (pour la déduction et la recherche de modèles, notamment). L'expressivité et le pouvoir de concision des langages de représentation logique de préférences sont étudiés dans [Coste-Marquis *et al.*, 2004] et leur complexité algorithmique dans [Lang, 2004].

Partant de là, puisque la spécification d'un jeu statique nécessite la description des préférences des agents, il apparaît naturel de représenter de tels jeux en utilisant des langages de représentation compacte de préférences. Il existe déjà plusieurs cadres répondant aux problèmes que nous avons posés plus haut, notamment les langages graphiques que nous présenterons dans le chapitre 4 (page 121). Nous avons choisi ici d'étudier le cas où chaque agent contrôle un ensemble fini de variables *binaires* :

les *jeux booléens* [Harrenstein *et al.*, 2001; Harrenstein, 2004a]. Cela permet non seulement de simplifier les jeux, mais aussi d'utiliser un langage logique de représentation compacte des préférences.

Un jeu booléen est un jeu à deux joueurs et à somme nulle, la fonction d'utilité du joueur 1 (et donc celle du joueur 2 qui est son opposé) est représentée par une formule de la logique propositionnelle, appelée *forme booléenne* du jeu. Après avoir donné une description (simplifiée) des jeux booléens, nous montrerons que ces jeux booléens peuvent facilement être généralisés de manière à représenter des jeux avec un nombre arbitraire de joueurs et à somme non nulle, mais en gardant l'hypothèse que les préférences de chaque joueur sont représentées par une formule propositionnelle unique, ce qui ne permet de représenter que des utilités binaires. Nous verrons alors comment des outils simples issus de la logique propositionnelle permettent de caractériser certaines propriétés du jeu, et nous donnerons quelques résultats de complexité algorithmique[1].

2.1 Introduction aux jeux booléens

Dans cette section, nous allons faire un rapide état de l'art des jeux booléens tels qu'ils ont été introduits par Harrenstein, van der Hoek, Meyer et Witteveen dans [Harrenstein *et al.*, 2001; Harrenstein, 2004a].

Un jeu booléen sur un ensemble de variables propositionnelles V est un jeu à deux joueurs, 1 et 2, à somme nulle, ayant les spécificités suivantes :

- les actions que peuvent entreprendre les deux joueurs consistent à donner une valeur de vérité à des variables de V ;
- les fonctions d'utilité des deux joueurs sont représentées au moyen d'une formule propositionnelle φ formée sur les variables V, appelée **forme booléenne** du jeu.

φ représente le but du joueur 1 : l'utilité de celui-ci est $+1$[2] lorsque φ est satisfaite[3] (et alors le joueur 1 gagne), et -1 sinon (et c'est alors le joueur 2 qui gagne). Les **utilités** sont donc **binaires** : il n'y en a que 2 possibles, 1 et -1.

Le jeu étant à somme nulle, nous avons $u_2 = -u_1$ et le but du joueur 2 n'est autre que $\neg\varphi$. Le jeu n'a donc que deux issues possibles : la victoire de 1 ou celle de 2.

Pour construire ces jeux booléens, [Harrenstein *et al.*, 2001; Harrenstein, 2004a] commencent par définir deux jeux booléens atomiques, dénotés par **1** et **0**. Le premier est gagné par 1, tandis que le second est gagné par 2, sans qu'aucun des joueurs n'ait à prendre la moindre décision. Des jeux booléens plus complexes sont ensuite construits récursivement à partir de ces jeux atomiques et d'un ensemble de variables propositionnelles, que l'on appellera variables de décision binaires. Chaque variable de décision est contrôlée de manière exclusive par l'un des deux joueurs. Pour tous jeux booléens g_0 et g_1, et pour toute variable de décision a, il existe un autre jeu booléen dénoté $a(g_0, g_1)$. Dans le jeu $a(g_0, g_1)$, c'est au joueur qui contrôle la variable a de décider de la valeur à laquelle il veut l'assigner. S'il choisit d'assigner a à vrai (resp. faux), alors le jeu continue avec g_1 (resp. g_0). Un jeu booléen est alors défini formellement comme suit :

[1] Les résultats donnés dans ce chapitre ont fait l'objet de plusieurs publications : [Bonzon *et al.*, 2005, 2006b, 2007a].
[2] On notera alors $u_1 = 1$.
[3] Une formule booléenne est satisfaite si et seulement si la formule est vraie.

Définition 2.1. *Soit A un ensemble fini de variables propositionnlles, et $\{0,1\}$ les deux joueurs. L'ensemble des* **formes booléennes sur** *A est le plus petit ensemble $B(A)$ tel que :*

- $\{0,1\} \subseteq B(A)$
- *si $a \in A$ et $g,h \in B(A)$, alors $a(g,h) \in B(A)$.*

Un **jeu booléen sur** *A est un couple (g,π), où g est une forme booléenne, et π est une* **fonction d'affectation de contrôle** *$(\pi : A \rightarrow \{0,1\})$, qui associe à chaque variable le joueur qui la contrôle (chaque variable étant contrôlée par un et un seul joueur).*

Exemple 2.1. *Soit $V = \{a,b,c\}$ un ensemble de variables propositionnelles. Soit 1 et 2 deux joueurs ayant pour buts : $\varphi_1 = (a \leftrightarrow b) \vee (\neg a \wedge b \wedge \neg c)$, $\varphi_2 = \neg\varphi_1 = (\neg a \wedge b \wedge c) \vee (a \wedge \neg b)$.*
Le joueur 1 contrôle les variables a et c, tandis que 2 contrôle la variable b.

La représentation proposée par Harrenstein dans [Harrenstein, 2004a] de ce jeu booléen est donnée en figure 2.1. Sur cette figure, la flèche gauche partant du nœud x représente la mise à faux de x, tandis que la flèche droite représente la mise à vrai de x.

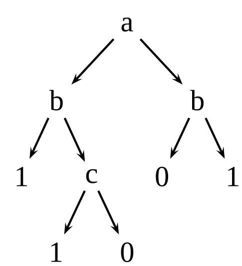


Figure 2.1 **—** Jeu booléen $(a,(b(\mathbf{1},c(\mathbf{1},\mathbf{0})),b(\mathbf{0},\mathbf{1})))$

Comme l'ont constaté Dunne et Van der Hoek [Dunne et van der Hoek, 2004], cette construction basée sur un modèle dynamique n'est pas nécessaire. En effet, l'hypothèse disant que les stratégies des agents sont choisies en parallèle (c'est-à-dire sans que l'un observe la décision de l'autre) est implicite. Cette forme extensive, sous forme d'arbre, est donc inutile. La représentation sous forme normale, qui correspond à celle d'un jeu statique, est ici plus simple, et a l'avantage de montrer clairement quel joueur gagnera à chaque issue du jeu .

Exemple 2.1 – suite *La forme normale du jeu booléen $(a,(b(\boldsymbol{1},c(\boldsymbol{1},\boldsymbol{0})),b(\boldsymbol{0},\boldsymbol{1})))$ est représentée Figure 2.2 (page suivante).*

[Harrenstein, 2004a, Section 8.4 (page 191)] introduit également des "jeux d'évaluation distribués" qui généralisent les jeux booléens en introduisant un nombre quelconque de joueurs et des préférences non binaires représentées par des ensembles de formules. Nous ne considèrerons pas cette généralisation dans ce chapitre, mais elle sera étudiée plus en avant dans le chapitre 3 (page 69).

2.2 Jeux booléens à *n* joueurs : définitions et exemples

Avant de revenir plus en détail aux jeux booléens tels qu'ils ont été définis dans [Harrenstein *et al.*, 2001; Harrenstein, 2004a], nous allons d'abord les généraliser en nous intéressant à des jeux à *n*

1 \ 2	b	$\overline{b}$
ac	(1, 0)	(0, 1)
$a\overline{c}$	(1, 0)	(0, 1)
$\overline{a}c$	(0, 1)	(1, 0)
$\overline{ac}$	(1, 0)	(1, 0)

***Figure 2.2* —** Forme normale du jeu booléen $(a, (b(\mathbf{1}, c(\mathbf{1}, \mathbf{0})), b(\mathbf{0}, \mathbf{1})))$

joueurs et à somme non nulle. Nous verrons ensuite que le cadre étudié par [Harrenstein *et al.*, 2001; Harrenstein, 2004a] est un cas particulier de ce cadre plus général.

Commençons par formaliser la notion de jeu booléen à n joueurs. Etant donné un ensemble de variables propositionnelles V, un jeu booléen sur V est un jeu à n joueurs pour lequel les actions disponibles de chaque joueur consistent à assigner une valeur de vérité à toutes les variables d'un sous-ensemble donné de V. Les préférences de chaque joueur i sont représentées par une formule propositionnelle φ_i formée sur les variables de V.

Définition 2.2. *Un* **jeu booléen à** n **joueurs** *est un 5-uplet* $(N, V, \pi, \Gamma, \Phi)$*, avec*

- $N = \{1, 2, \ldots, n\}$ *l'ensemble des joueurs (appelés aussi agents) ;*
- V *un ensemble de variables propositionnelles ;*
- $\pi : N \mapsto V$ *une fonction d'affectation de contrôle ;*
- $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$ *l'ensemble des contraintes, chaque* γ_i *étant une formule propositionnelle de* $L_{\pi(i)}$ *satisfiable ;*
- $\Phi = \{\varphi_1, \ldots, \varphi_n\}$ *l'ensemble des buts, chaque* φ_i *étant une formule de* L_V *satisfiable.*

Un 4-uplet (N, V, π, Γ)*, avec* N, V, π, Γ *definis comme ci-dessus est appelé un* **pré-jeu booléen**.

La *fonction d'affectation de contrôle* π associe à chaque joueur les variables qu'il contrôle. On note π_i l'*ensemble des variables contrôlées* par le joueur i. Chaque variable est contrôlée par un et un seul joueur. Ainsi, $\{\pi_1, \ldots, \pi_n\}$ forme une partition de V.

Chaque γ_i représente ici les contraintes d'un agent sur l'ensemble des variables qu'il contrôle. Ce choix de représentation est assez intuitif : en règle générale, les contraintes d'une personne reposent sur les actions qu'il lui est possible d'effectuer. Si une de mes contraintes pèse sur une action d'un autre joueur, je ne peux pas être assurée de la respecter, ce n'est pas de mon ressort. Ainsi, cette représentation des contraintes permet de respecter l'indépendance des agents : chaque agent gère ses variables, et les contraintes qui y pèsent, sans être tributaire d'un autre agent pour cela.

L'utilisation des jeux booléens permet d'avoir une représentation compacte du problème. Pour illustrer ce propos, nous allons utiliser une variante de l'exemple 1.1 (page 10) du dilemne des prisonniers : nous considérons ici n prisonniers qui ne peuvent bénéficier que d'un seul type de remise de peine afin de simplifier le problème.

Exemple 2.2. *Dans le jeu simplifié du prisonnier à n joueurs, n détenus (notés* $1, \ldots, n$*) sont emprisonnés dans des cellules séparées. La police fait à chacun d'eux le même marché :*

> *"Tu as le choix entre couvrir tes complices en te taisant (noté T_i, $i = 1, \ldots, n$) ou les trahir en les dénonçant (noté $\neg T_i$, $i = 1, \ldots, n$). Si tu les dénonces, tu auras une remise de peine et tes partenaires purgeront le maximum (sauf si l'un d'eux t'a dénoncé aussi, auquel cas il bénéficiera comme toi d'une remise de peine). Mais si vous vous couvrez mutuellement, vous aurez tous une remise de peine[4]."*

La représentation de ce jeu en forme normale pour $n = 3$ est la suivante :

$3 : T_3$		
1 \ 2	T_2	$\neg T_2$
T_1	*(1, 1, 1)*	*(0, 1, 0)*
$\neg T_1$	*(1, 0, 0)*	*(1, 1, 0)*

$3 : \neg T_3$		
1 \ 2	T_2	$\neg T_2$
T_1	*(0, 0, 1)*	*(0, 1, 1)*
$\neg T_1$	*(1, 0, 1)*	*(1, 1, 1)*

Dans cette forme normale, les n-uplets (ici des triplets) donnent le résultat obtenu par les n joueurs dans l'ordre : (résultat joueur 1, résultat joueur 2, ...). Le 0 (resp. 1) signifie que le joueur concerné perd (resp. gagne).

On constate ici que pour n prisonniers, on aura une matrice à n dimensions, chaque dimension étant égale à 2, donc 2^n n-uplets à spécifier[5].

Or, ce jeu peut être traduit très simplement par le jeu booléen $G = (N, V, \pi, \Gamma, \Phi)$ suivant :

- $N = \{1, 2, \ldots, n\}$,
- $V = \{T_1, \ldots, T_n\}$,
- $\forall i \in \{1, \ldots, n\}, \pi_i = \{T_i\}$,
- $\forall i \in \{1, \ldots, n\}, \gamma_i = \top$, *et*
- $\forall i \in \{1, \ldots, n\}, \varphi_i = (T_1 \wedge T_2 \wedge \ldots \wedge T_n) \vee \neg T_i$.

L'utilisation des jeux booléens permet donc de réduire de manière très significative la taille de la représentation du problème.

Définition 2.3. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen. Une* **stratégie** *s_i pour un joueur i de G est une π_i-interprétation satisfaisant γ_i. L'***ensemble des stratégies du joueur** *i est représenté par $S_i = \{s_i \in 2^{\pi_i} | s_i \models \gamma_i\}$.*

Un **profil de stratégies** *s pour G est un n-uplet $s = (s_1, \ldots, s_n)$, avec pour tout i, $s_i \in S_i$. $S = S_1 \times \ldots \times S_n$ est l'ensemble de tous les profils de stratégies.*

En d'autres mots, une stratégie pour le joueur i est une affectation à vrai ou faux des variables qu'il contrôle. Pour chaque joueur i, γ_i est l'ensemble des contraintes restreignant l'ensemble des profils de stratégies possibles pour ce joueur.

Pour tout ensemble non vide de joueurs (appelé **coalition**) $I \subseteq N$, la projection de s sur I est définie par $s_I = (s_i)_{i \in I}$. Si $I = \{i\}$, la projection de s sur $\{i\}$ est dénotée par s_i au lieu de $s_{\{i\}}$.

[4] L'emploi de préférences binaires ne nous permet pas ici d'exprimer le fait qu'un prisonnier préfère la situation dans laquelle il dénonce ses complices tandis que les autres le couvrent à la situation dans laquelle tout le monde couvre tout le monde, elle-même préférée à la situation dans laquelle tout le monde dénonce. Pour faire cela, nous aurons besoin d'un langage plus sophistiqué, cf. Chapitre 3 (page 69).

[5] Ou un arbre binaire à 2^n feuilles si on utilise une représentation sous forme extensive.

Comme décrit en section 1.1.1.1 (page 10), les notations suivantes sont usuelles en théorie des jeux. Soit $s = (s_1, \dots, s_n)$ et $s' = (s'_1, \dots, s'_n)$ deux profils de stratégies.

On note s_{-i} le *profil de stratégies s privé de la stratégie du joueur i* : $s_{-i} = (s_1, s_2, \dots, s_{i-1}, s_{i+1}, \dots, s_n)$. On note (s_{-i}, s'_i) *le profil de stratégies s dans lequel on a remplacé la stratégie du joueur i par celle du profil s'* : $(s_{-i}, s'_i) = (s_1, s_2, \dots, s_{i-1}, s'_i, s_{i+1}, \dots, s_n)$. Similairement, si $I \subseteq N$, on note $s_{-I} = s_{N \setminus I}$.

π_I représente l'ensemble des variables contrôlées par I, et $\pi_{-I} = \pi_{N \setminus I}$. Si $I = \{i\}$, on note π_{-i} *l'ensemble des variables contrôlées par tous les joueurs sauf le joueur i* : $\pi_{-i} = V \setminus \pi_i$. $\{\pi_1, \dots, \pi_n\}$ formant une partition de V, un profil de stratégies s est une interpretation pour V, *i.e.* $s \in 2^V$. π^{-1} représente la fonction inverse de π.

L'ensemble des stratégies de $I \subseteq N$ est dénoté par $S_I = \times_{i \in I} S_i$, et l'ensemble des buts de $I \subseteq N$ par $\Phi_I = \bigwedge_{i \in I} \varphi_i$.

Etudions l'exemple suivant pour illustrer ces notions.

Exemple 2.3. *Soit* $G = (N, V, \pi, \Gamma, \Phi)$ *un jeu booléen, avec*

* $V = \{a, b, c\}$, $N = \{1, 2\}$,
* $\pi_1 = \{a, b\}$ *et* $\pi_2 = \{c\}$.
* $\gamma_1 = \neg a \vee \neg b$, $\gamma_2 = \top$
* $\varphi_1 = (a \leftrightarrow b) \vee (\neg a \wedge b \wedge \neg c)$,
* $\varphi_2 = (\neg a \wedge b \wedge c) \vee (a \wedge \neg b)$,

Le joueur 1 *a trois stratégies possibles :* $s_{1_1} = a\overline{b}$, $s_{1_2} = \overline{a}b$, $s_{1_3} = \overline{a}\overline{b}$. *La stratégie ab ne satisfait pas les contraintes du joueur* 1, *et donc n'est pas une stratégie acceptable.*

Le joueur 2 *a deux stratégies possibles :* $s_{2_1} = c$ *ou* $s_{2_2} = \overline{c}$.

G possède donc 6 profils de stratégies : $S = \{a\overline{b}c, a\overline{b}\overline{c}, \overline{a}bc, \overline{a}b\overline{c}, \overline{a}\overline{b}c, \overline{a}\overline{b}\overline{c}\}$.

Les profils de stratégies $\overline{a}\overline{b}c$, $\overline{a}b\overline{c}$ *et* $\overline{a}\overline{b}\overline{c}$ *donnent la victoire à* 1, *tandis que* $a\overline{b}c$, $a\overline{b}\overline{c}$ *et* $\overline{a}bc$ *permettent à* 2 *de gagner.*

Le but φ_i du joueur i est une relation de préférence dichotomique et compacte, correspondant donc à une fonction d'utilité binaire[6] : un joueur i est satisfait, et a une utilité de 1, si et seulement si son but φ_i est satisfait. Il a une utilité de 0 sinon. Les buts $\{\varphi_i, i = 1, \dots, n\}$ jouent donc le rôle des fonctions d'utilité.

Définition 2.4. *Pour chaque joueur i, la* **fonction d'utilité** *induite par le but de ce joueur est la fonction* $u_i : S \to \{0, 1\}$ *telle que :*

$$u_i(s) = \begin{cases} 0 \text{ si } s \models \neg\varphi_i \\ 1 \text{ si } s \models \varphi_i \end{cases}$$

Nous avons :

* s est au moins aussi bon que s' pour i, dénoté par $s \succeq_i s'$, si $u_i(s) \geq u_i(s')$, ou de façon équivalente si $s \models \neg\varphi_i$ implique $s' \models \neg\varphi_i$;
* s est strictement meilleur que s' pour i, dénoté par $s \succ_i s'$, si $u_i(s) > u_i(s')$, ou, de façon équivalente, $s \models \varphi_i$ et $s' \models \neg\varphi_i$.

[6]Nous verrons comment lever cette contrainte dans le chapitre 3 (page 69).

* i est indifferent entre s et s', dénoté par $s \sim_i s'$, si $s \geq_i s'$ et $s' \geq_i s$, ou de façon équivalente si $s \models \varphi_i$ si et seulement si $s' \models \varphi_i$.

Définissons à présent la notion de stratégie gagnante pour un joueur.

Définition 2.5. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen, avec $\Phi = \{\varphi_1, \dots, \varphi_n\}$, et $N = \{1, \dots, n\}$. La stratégie s_i est une* **stratégie gagnante** *pour le joueur i si, quels que soient les choix effectués par ses adversaires, i est sûr de gagner en choisissant cette stratégie.*

$$\forall s_{-i} \in S_{-i}, (s_{-i}, s_i) \models \varphi_i$$

2.3 Graphe de dépendance entre les joueurs

De nombreuses structures graphiques sont cachées dans un jeu booléen : la satisfaction de chaque joueur i dépend des joueurs qui contrôlent des variables de φ_i. Il est donc possible de représenter graphiquement les dépendances entre joueurs en utilisant la nature syntactique des buts. Intuitivement, si le but d'un joueur i ne dépend d'aucune variable contrôlée par le joueur j, alors la satisfaction de i ne dépendra pas directement de j. Toutefois, ce n'est qu'une condition suffisante : il se peut que l'expression syntactique de φ_i contienne une variable contrôlée par j , mais que cette variable ne joue aucun rôle dans la satisfaction de φ_i, comme c'est le cas pour la variable y dans $\varphi_i = x \wedge (y \vee \neg y)$. Nous utiliserons donc une notion plus forte d'indépendance entre une formule et une variable que la simple notion d'indépendance syntactique [Lang et Marquis, 1998a; Lang *et al.*, 2003].

Définition 2.6. *Une formule propositionnelle φ est* **indépendante** *d'une variable propositionnelle x s'il existe une fornule ψ logiquement équivalente à φ et dans laquelle x n'apparaît pas*[7].

Définition 2.7. *Soit $G = (N, V, \Gamma, \pi, \Phi)$ un jeu booléen. L'ensemble des* **variables utiles** *pour un joueur i, dénoté par $RV_G(i)$, est l'ensemble de toutes les variables $v \in V$ telles que φ_i n'est pas indépendante de v.*

Pour faciliter les notations, l'ensemble des variables utiles pour un joueur i dans un jeu booléen G sera noté RV_i au lieu de $RV_G(i)$. On peut à présent facilement définir l'ensemble des *joueurs utiles* pour un joueur i comme étant l'ensemble des joueurs contrôlant au moins une variable de RV_i.

Définition 2.8. *Soit $G = (N, V, \Gamma, \pi, \Phi)$ un jeu booléen. L'ensemble des* **joueurs utiles** *pour un joueur i, dénoté par*[8] *RP_i, est l'ensemble des joueurs $j \in N$ tels que j contrôle au moins une variable utile de i : $RP_i = \bigcup_{v \in RV_i} \pi^{-1}(v)$.*

Exemple 2.4. *Axel, Kevin et Virginie sont invités à une fête. Axel veut y aller. Kevin aussi, mais si et seulement si Axel y va. Quant à Virginie, elle voudrait y aller, que Kevin y aille mais pas Axel. Cette situation peut être modélisée par le jeu booléen $G = (N, V, \Gamma, \pi, \Phi)$, défini par*

* $N = \{1, 2, 3\}$, *Axel étant le joueur* 1, *Kevin le* 2 *et Virginie* 3 *;*

[7] Il existe une caractérisation sémantique équivalente de l'indépendance entre une formule et une variable [Lang *et al.*, 2003] : φ est indépendante de x s'il existe une interprétation s telle que $s \models \varphi$ et $\mathrm{switch}(s, x) \models \varphi$, où $\mathrm{switch}(s, x)$ est obtenue en changeant la valeur de x dans s, et en laissant les valeurs des autres variables inchangées.

[8] Comme précédemment, l'ensemble des joueurs utiles pour un joueur i dans un jeu booléen G devrait être noté $RP_G(i)$. Pour faciliter les notations nous écrirons simplement RP_i.

- $V = \{a, b, c\}$, *où a signifie "1 va à la fête", et de même pour b et 2, et pour c et 3 ;*
- *pour chaque* i, $\gamma_i = \top$ *;*
- $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\pi_3 = \{c\}$ *;*
- $\varphi_1 = a$,
- $\varphi_2 = a \leftrightarrow b$ *et*
- $\varphi_3 = \neg a \wedge b \wedge c$.

On peut voir que la satisfaction d'Axel ne dépend que de lui-même, celle de Kevin depend de la décision d'Axel et de la sienne, tandis que celle de Virginie dépend des décisions de tous les joueurs.

On a : $RV_1 = \{a\}$, $RV_2 = \{a, b\}$, $RV_3 = \{a, b, c\}$, $RP_1 = \{1\}$, $RP_2 = \{1, 2\}$, $RP_3 = \{1, 2, 3\}$.

Cette relation entre les joueurs peut être vue comme un graphe orienté contenant un nœud pour chaque joueur, et un arc de i à j si $j \in RP_i$, c'est-à-dire si j est un joueur utile pour i.

Définition 2.9. *Le* **graphe de dépendance d'un jeu booléen** $G = (N, V, \Gamma, \pi, \Phi)$ *est un graphe orienté* $\mathcal{P} = \langle N, R \rangle$*, avec* $\forall i, j \in N$, $(i, j) \in R$ *(dénoté par* $R(i, j)$*) si* $j \in RP_i$.

$R(i)$ est donc l'ensemble des joueurs nécessaires à i pour satisfaire son but : $j \in R(i)$ si et seulement si $j \in RP_i$. Il faut pourtant remarquer que $j \in R(i)$ n'implique pas que i *a besoin de* j pour que son but soit satisfait. Par exemple, si $\pi_1 = \{a\}$, $\pi_2 = \{b\}$ et $\varphi_1 = a \vee b$, alors $2 \in R(1)$. Pourtant, 1 a une stratégie lui permettant de satisfaire son but (en mettant a à vrai) et n'a donc pas besoin de 2.

La fermeture transitive de R est notée R^*. $R^*(i, j)$ représente le chemin de i à j dans R. $R^*(i)$ représente donc tous les joueurs qui ont une influence directe ou indirecte sur i. $R^{*-1}(i)$ représente tous les joueurs sur lesquels i a une influence directe ou indirecte.

Exemple 2.4 (page précédente), suite : *Le graphe de dépendance* $\mathcal{P}$ *induit par G est représenté figure 2.3.*

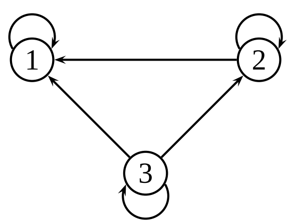


Figure 2.3 — Graphe de dépendance du jeu de la fête

- $R^{-1}(1) = \{1, 2, 3\}$, $R^{-1}(2) = \{2, 3\}$, $R^{-1}(3) = \{3\}$.
- $R^*(1) = \{1\}$, $R^*(2) = \{1, 2\}$ *et* $R^*(3) = \{1, 2, 3\}$.
- $R^{*-1}(1) = \{1, 2, 3\}$, $R^{*-1}(2) = \{2, 3\}$ *et* $R^{*-1}(3) = \{3\}$.

Propriété 2.1. *Tout graphe de dépendance* $\mathcal{P} = \langle N, R \rangle$ *représente au moins un jeu booléen* $G = (N, V, \Gamma, \pi, \Phi)$.

> *Preuve :*
> Soit $\mathcal{P} = \langle N, R \rangle$ un graphe de dépendance. Soit $G = (N, V, \Gamma, \pi, \Phi)$ un jeu booléen, avec $V = \{v_1, \ldots, v_n\}$, $\forall i \in N$, $\pi_i = \{v_i\}$, et pour tout i, $\gamma_i = \top$.
> Les buts de G sont construits comme suit : $\forall i \in N$, $\forall j$ tel que $j \in R(i)$, alors $\varphi_i =$

$\bigwedge_j v_j$. Si $\nexists j$ tel que $j \in R(i)$, alors $\varphi_i = \top$.
Le jeu construit de cette façon est un jeu booléen.

■

On introduit à présent la notion d'ensemble stable afin d'exploiter au mieux ce graphe de dépendance. Un ensemble de nœuds B est stable pour une relation R si tous les arcs de R partant d'un nœud de B atteignent un autre nœud de B. L'ensemble des joueurs utiles d'un ensemble stable B sont les joueurs dans B.

Définition 2.10. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen. $B \subseteq N$ est* **stable**[9] *pour R si et seulement si $R(B) \subseteq B$, c'est-à-dire $\forall j \in B, \forall i$ tels que $i \in R(j)$, alors $i \in B$.*

On remarque facilement que $\varnothing$ et N sont des ensembles stables, et que l'ensemble des ensembles stables pour un jeu booléen est fermé pour l'union et l'intersection. Ces quatre propriétés caractérisent complètement l'ensemble des coalitions stables pour un jeu booléen.

Propriété 2.2. *Soit $\mathcal{C} \subset 2^N$. Il existe un jeu booléen G tel que $\mathcal{C}$ est l'ensemble des ensembles stables de G si et seulement si $\mathcal{C}$ satisfait les quatre propriétés suivantes :*

1. *$\varnothing \in \mathcal{C}$;*
2. *$N \in \mathcal{C}$;*
3. *Si $B, B' \in \mathcal{C}$ alors $B \cup B' \in \mathcal{C}$;*
4. *Si $B, B' \in \mathcal{C}$ alors $B \cap B' \in \mathcal{C}$.*

Preuve : Grâce à la propriété 2.1 (page ci-contre), il suffit de montrer qu'il existe une relation R sur N telle que $\mathcal{C}$ est l'ensemble des ensembles stables pour R si et seulement si $\mathcal{C}$ satisfait (1)–(4).
$\Rightarrow$ $\varnothing$ et N sont évidemment stables pour R.

Si B et B' sont stables, alors $R(B) \subseteq B$ et $R(B') \subseteq B'$. Donc, $R(B) \cup R(B') \subseteq B \cup B'$. Par définition, $R(B) = \{j | \exists i \in B : j \in RP_i\}$ et $R(B') = \{j | \exists i \in B' : j \in RP_i\}$. Donc, $R(B) \cup R(B') = \{j | \exists i \in B \cup B' : j \in RP_i\} = R(B \cup B')$. Alors, $R(B \cup B') \subseteq B \cup B'$, et $B \cup B'$ est un ensemble stable.

Si B et B' sont stables, alors $R(B) \subseteq B$ et $R(B') \subseteq B'$. Donc, $R(B) \cap R(B') \subseteq B \cap B'$. Par définition, $R(B) = \{j | \exists i \in B : j \in RP_i\}$ et $R(B') = \{j | \exists i \in B' : j \in RP_i\}$. Donc, $R(B) \cap R(B') = \{j | \exists i \in B \cap B' : j \in RP_i\} = R(B \cap B')$. Alors, $R(B \cap B') \subseteq B \cap B'$, et $B \cap B'$ est un ensemble stable.

$\Leftarrow$ Soit $\mathcal{C}$ un ensemble de coalitions satisfaisant (1)–(4). Pour tout $i \in N$, soit X_i le plus petit ensemble de $\mathcal{C}$ contenant i : $X_i = \bigcap \{B \in \mathcal{C} | i \in B\}$ (puisque $\mathcal{C}$ est fermé pour $\cap$, nous avons $X_i \in \mathcal{C}$). Construisons à présent R tel que pour tout $i, j \in N$, $(i, j) \in R$ si et seulement si $j \in X_i$.

On veut à présent vérifier que pour tout $B \in \mathcal{C}$, B est stable pour R. Supposons que $B \in \mathcal{C}$ et que B ne soit pas stable (il existe un $i \in B$ tel qu'il existe un

[9] La notion d'ensemble stable que nous définissons ici n'est pas la même que celle utilisée en théorie de graphe : nous ne parlons pas ici des ensembles de sommets deux-à-deux non adjacents.

$j \in R(i)$ et $j \notin B$). Comme $j \in R(i)$, on sait par construction de R que $j \in X_i$, et donc $j \in \bigcap\{B \in \mathcal{C} | i \in B\}$, donc $j \in B$, et donc B est stable pour R, ce qui est en contradiction avec l'hypothèse effectuée.

On montre ensuite que si $B \notin \mathcal{C}$, alors B n'est pas stable pour R. Puisque $\mathcal{C}$ est fermé pour $\cup$, $\bigcup_{i \in B} X_i \neq B$. Donc il existe un $i \in B$ tel qu'il existe $k \in X_i$ et que $k \notin B$. Par construction de R, $R(B) \not\subseteq B$, et B n'est pas stable pour R.

■

On définit à présent la projection d'un jeu booléen G sur un ensemble stable de joueurs $B \subseteq N$, afin de pouvoir décomposer un jeu booléen en plusieurs jeux :

Définition 2.11. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen, et $B \subseteq N$ un ensemble stable pour R. La* **projection** *de G sur B est définie par $G_B = (B, V_B, \pi_B, \Gamma_B, \Phi_B)$, où $V_B = \cup_{i \in B} \pi_i$, $\pi_B : B \to V_B$ telle que $\pi_B(i) = \{v | v \in \pi_i\}$, $\Gamma_B = \{\gamma_i | i \in B\}$, et $\Phi_B = \{\varphi_i | i \in B\}$.*

Propriété 2.3. *Si B est un ensemble stable, $G_B = (B, V_B, \pi_B, \Gamma_B, \Phi_B)$ est un jeu booléen.*

Preuve : Soit $G_B = (B, V_B, \pi_B, \Gamma_B, \Phi_B)$.

* $B \subseteq N$ est l'ensemble des joueurs,
* $V_B = \cup_{i \in B} \pi_i$. Nous avons évidemment $V_B \subseteq V$, et donc V_B est l'ensemble des variables.
* Nous devons vérifier que chaque but φ_i pour $i \in B$ est une formule de L_{V_B}, ou est logiquement équivalente à une formule de L_{V_B}.

  Supposons que $\exists i \in B$, $\exists v \in Var(\varphi_i)$ tel que $v \notin V_B$. Donc, $\forall j \in B$, $v \notin \pi_j$. Soit $k \in N \setminus B$ tel que $v \in \pi_k$. Nous savons que $v \in Var(\varphi_i)$, donc soit φ_i est indépendant de v, et est donc logiquement équivalente à une formule dans laquelle v n'apparaît pas, soit φ_i n'est pas indépendante de v, et dans ce cas $v \in RV_i$, et par définition $k \in RP_i$. Donc, $k \in R(i)$, mais $k \notin B$: c'est en contradiction avec le fait que B est stable.
* $\pi_B : B \to V_B$ tel que $\pi_B(i) = \{v | v \in \pi_i\}$ est l'ensemble des fonctions d'affectation de contrôle des joueurs de B.
* $\Gamma_B = \{\gamma_i | i \in B\}$ est l'ensemble des contraintes des joueurs de B. Comme chaque γ_i est une formule propositionnelle de L_{π_i}, Γ_B est bien défini pour le jeu G_B.

■

Exemple 2.5. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ le jeu booléen défini par $V = \{a, b, c\}$, $N = \{1, 2, 3\}$, $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\pi_3 = \{c\}$, pour chaque i, $\gamma_i = \top$, $\varphi_1 = a \leftrightarrow b$, $\varphi_2 = a \leftrightarrow \neg b$ et $\varphi_3 = \neg c$.*

Nous avons : $RV_1 = \{a, b\}$, $RV_2 = \{a, b\}$, $RV_3 = \{c\}$, $RP_1 = \{1, 2\}$, $RP_2 = \{1, 2\}$, $RP_3 = \{3\}$.

Le graphe de dépendance $\mathcal{P}$ de G est représenté figure 2.4 (page ci-contre).

Les ensembles de joueurs $B = \{1, 2\}$ et $C = \{3\}$ sont stables. On peut décomposer G en 2 jeux booléens :

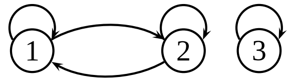


Figure 2.4 **—** Graphe de dépendance de l'exemple 2.5

* $G_B = (B, V_B, \pi_B, \Gamma_B, \Phi_B)$, *avec* $B = \{1,2\}$, $V_B = \{a,b\}$, $\pi_1 = a$, $\pi_2 = b$, $\gamma_1 = \gamma_2 = \top$, $\varphi_1 = a \leftrightarrow b$, $\varphi_2 = a \leftrightarrow \neg b$.
* $G_C = (C, V_C, \pi_C, \Gamma_C, \Phi_C)$, *avec* $C = \{3\}$, $V_C = \{c\}$, $\pi_3 = c$, $\gamma_3 = \top$, $\varphi_3 = \neg c$.

Cette décomposition nous permet ainsi d'étudier deux jeux plus simples, G_B *et* G_C*, plutôt que d'étudier le jeu G. Nous verrons dans la suite que nous pouvons ainsi calculer les équilibres de Nash sur les sous-jeux plutôt que sur le jeu complet.*

2.4 Concepts de solution : équilibres de Nash et stratégies dominées[10]

2.4.1 Equilibres de Nash

La définition des équilibres de Nash en stratégie pures (PNE) pour les jeux booléens à n joueurs est la même que la définition classique en théorie des jeux (voir Chapitre 1, Définition 1.5 (page 19)), en ayant à l'esprit que les fonctions d'utilité sont induites par les buts des joueurs $\varphi_1, \ldots, \varphi_n$.

Rappelons qu'un équilibre de Nash est un profil de stratégies tel que la stratégie de chaque joueur est une réponse optimale aux stratégies des autres joueurs, et étudions un exemple simple.

Exemple 2.6. *Soit* $G = (N, V, \pi, \Gamma, \Phi)$ *un jeu booléen avec*

* $V = \{a,b,c\}$, $N = \{1,2,3\}$, *pour tout* i, $\gamma_i = \top$,
* $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\pi_3 = \{c\}$,
* $\varphi_1 = (\neg a \vee (a \wedge b \wedge \neg c))$,
* $\varphi_2 = (a \leftrightarrow (b \leftrightarrow c))$ *et*
* $\varphi_3 = ((a \wedge \neg b \wedge \neg c) \vee (\neg a \wedge b \wedge c))$.

On peut à présent construire la forme normale de G :

stratégie de 3 : c		
1 \ 2	$\overline{b}$	b
$\overline{a}$	(1, 1, 0)	(1, 0, 1)
a	(0, 0, 0)	(0, 1, 0)

stratégie de 3 : $\overline{c}$		
1 \ 2	$\overline{b}$	b
$\overline{a}$	(1, 0, 0)	(1, 1, 0)
a	(0, 1, 1)	(1, 0, 0)

On constate tout d'abord que le joueur 1 *a une stratégie gagnante. En effet, s'il choisit d'instancier la variable a à faux, alors il est sûr de gagner quels que soient les choix de* 2 *et* 3.

[10] Certains des résultats de cette section ont été obtenus en collaboration avec Bruno Zanuttini.

Etudions à présent les équilibres de Nash. Pour cela, il faut étudier chaque profil de stratégies.

abc :	1, *qui contrôle a, préfère* $\overline{a}bc$
$ab\overline{c}$:	2, *qui contrôle b, préfère* $a\overline{b}\overline{c}$
$a\overline{b}c$:	2 , *qui contrôle b, préfère* abc
$\overline{a}bc$:	2, *qui contrôle b, préfère* $\overline{a}\overline{b}c$
$a\overline{b}\overline{c}$:	1, *qui contrôle a, préfère* $\overline{a}\overline{b}\overline{c}$
$\overline{a}b\overline{c}$:	3, *qui contrôle c, préfère* $\overline{a}bc$
$\overline{a}\overline{b}c$:	*Equilibre de Nash (aucun joueur ne préfère un autre profil)*
$\overline{a}\overline{b}\overline{c}$:	2, *qui contrôle b, préfère* $\overline{a}b\overline{c}$

Le profil de stratégies $\overline{a}\overline{b}c$ *est donc le seul équilibre de Nash du jeu G.*

Donnons à présent une caractérisation simple des équilibres de Nash aux stratégies pures.

Propriété 2.4. *Soit $s \in S$. s est un équilibre de Nash en stratégies pures de G si et seulement si pour tout $i \in N$, on a :*

- *soit* $s \models \varphi_i$,
- *soit* $s_{-i} \models \neg\varphi_i$.

Cette caractérisation, simple, permet de calculer plus facilement les équilibres de Nash en stratégies pures des jeux booléens. Ainsi, un profil de stratégies s sera un PNE si et seulement si, pour chacun des joueurs i, soit s permet d'obtenir le but du joueur i, soit le joueur i ne peut obtenir son but si les autres joueurs maintiennent leurs stratégies.

Preuve : s est un équilibre de Nash en stratégies pures de G si et seulement si pour tout $i \in N$ et pour tout $s'_i \in S_i$ on a $u_i(s) \geq u_i(s_{-i}, s'_i)$. Or, puisque $u_i(s)$ et $u_i(s_{-i}, s'_i)$ ne peuvent prendre que deux valeurs, cette inégalité est équivalente à

$$\forall i \in N \text{ et } \forall s'_i \in S_i, u_i(s) = 1 \text{ ou } u_i(s_{-i}, s'_i) = 0$$

c'est-à-dire

$$\forall i \in N \quad \text{soit} \quad u_i(s) = 1 \tag{2.1}$$

$$\text{soit} \quad \forall s'_i \in S_i, u_i(s_{-i}, s'_i) = 0 \tag{2.2}$$

2.1 est équivalente à $s \models \varphi_i$.
2.2 est équivalente à $\forall s'_i \in S_i$, $(s_{-i}, s'_i) \models \neg\varphi_i$, donc $s_{-i} \models \neg\varphi_i$.

■

Exemple 2.6 (page précédente), suite : *Nous avons vu que le seul équilibre de Nash de ce jeu est* $s = \overline{a}\overline{b}c$. *En effet, nous avons :*

- $s = \overline{a}\overline{b}c \models \varphi_1 = (\neg a \vee (a \wedge b \wedge \neg c))$
- $s = \overline{a}\overline{b}c \models \varphi_2 = (a \leftrightarrow (b \leftrightarrow c))$
- $s_{-3} = \overline{a}\overline{b} \models \neg\varphi_3 = ((\neg a \vee b \vee c) \wedge (a \vee \neg b \vee \neg c))$

On peut vérifier qu'aucun autre profil de stratégies ne satisfait l'une ou l'autre de ces conditions.

Cette caractérisation des équilibres de Nash en stratégies pures peut être encore simplifiée. En effet, vérifier que $s_{-i} \models \neg\varphi_i$ revient à vérifier que s_{-i} peut inférer $\neg\varphi_i$ *quelles que soient* les valeurs prises par les variables contrôlées par le joueur i. Cela évoque la notion de projection d'une formule sur un ensemble de variables, qui correspond à l'utilisation de l'opérateur *forget*, notion qui a été étudiée par [Lin et Reiter, 1994; Lang *et al.*, 2002a, 2003], et qui est rappelée dans le chapitre de notations (page 7).

On généralise ici cette notion d'oubli en utilisant la notation suivante : $\forall i : \varphi \equiv \neg\exists i : \neg\varphi$ qui représente l'oubli de toutes les variables contrôlées par le joueur i.

Si l'on revient à la seconde équation de la propriété 2.4 (page précédente), on voit que l'on a :

$$
\begin{aligned}
s_{-i} \models \neg\varphi_i \quad & \Leftrightarrow \quad s_{-i} \models \forall i : \neg\varphi_i && (2.3)\\
& \Leftrightarrow \quad (s_i, s_{-i}) \models \forall i : \neg\varphi_i && (2.4)\\
& \Leftrightarrow \quad s \models \forall i : \neg\varphi_i && (2.5)
\end{aligned}
$$

L'équation 2.3 est équivalente à l'équation 2.4 car les variables contrôlées par le joueur i ont disparu de $\forall i : \neg\varphi_i$.

Cela nous permet d'établir le corollaire suivant, qui simplifie la caractérisation des PNE que nous avons donnée dans la propriété 2.4 (page ci-contre) :

Corollaire 2.1. *Soit $s \in S$. Alors s est un équilibre de Nash en stratégies pures pour G si et seulement si :*

$$s \models \bigwedge_{i \in N} (\varphi_i \vee (\forall i : \neg\varphi_i))$$

Preuve :

$$
\begin{aligned}
& \forall i \in N, s \models \varphi_i \text{ ou } s_{-i} \models \neg\varphi_i\\
\Leftrightarrow \quad & \forall i \in N, s \models \varphi_i \text{ ou } s \models \forall i : \neg\varphi_i\\
\Leftrightarrow \quad & \forall i \in N, s \models \varphi_i \vee (\forall i : \neg\varphi_i)\\
\Leftrightarrow \quad & s \models \textstyle\bigwedge_{i \in N} (\varphi_i \vee (\forall i : \neg\varphi_i))
\end{aligned}
$$

■

Exemple 2.6 (page 41), suite : *Comme précédemment, le seul équilibre de Nash de ce jeu est $\overline{a}\overline{b}c$. Calculons $\forall 3 : \neg\varphi_3$:*

$$
\begin{aligned}
\forall 3 : \neg\varphi_3 \quad & = \quad \neg\varphi_{3_{c\leftarrow\top}} \wedge \neg\varphi_{3_{c\leftarrow\bot}}\\
& = \quad ((\neg a \vee b \vee \top) \wedge (a \vee \neg b \vee \bot)) \wedge ((\neg a \vee b \vee \bot) \wedge (a \vee \neg b \vee \top))\\
& = \quad (a \vee \neg b) \wedge (\neg a \vee b)
\end{aligned}
$$

On a donc bien :

$$\overline{a}\overline{b}c \models \varphi_1 \wedge \varphi_2 \wedge \forall 3 : \neg\varphi_3$$

Le graphe de dépendance entraîne également quelques propriétés sur les équilibres de Nash.

Propriété 2.5. *Soit G un jeu booléen. Si $\forall i \in N$, $i \notin RP_i$ alors tout $s \in S$ est un PNE.*

Si un joueur ne contrôle aucune des variables qui lui sont utiles, il n'a pas d'influence sur son propre but, et donc aucune préférence pour une ou l'autre de ses stratégies. Si c'est le cas pour tous les agents, aucun profil de stratégies n'est préféré, et tous sont donc des équilibres de Nash en stratégies pures.

> *Preuve :* Puisque $\forall i \in N$, $i \notin RP_i$, aucun joueur i n'a d'influence sur son propre but :
> $\forall s_i \in S_i$, $(s_i, s_{-i}) \models \varphi_i$ ou $s_{-i} \models \neg\varphi_i$. Donc, $\forall i \in N$, soit $s_{-i} \models \varphi_i$, soit $s_{-i} \models \neg\varphi_i$.
> La propriété 2.4 (page 42) s'applique pour tout s, et s est donc un PNE.

■

Propriété 2.6. *Soit G un jeu booléen tel que $\forall i \in N$, $|RP_i| = 1$, et $\exists i \in N$ tel que $RP_i = \{i\}$.*

s est un PNE si et seulement si $\forall i \in N$ tels que $RP_i = \{i\}$, $s \models \varphi_i$.

Si chaque joueur d'un jeu booléen ne dépend que d'un seul joueur, et qu'au moins un joueur ne dépend que de lui-même, les joueurs tels que $RP_i = \{i\}$ seront les seuls à avoir une influence sur leurs buts, et donc à pouvoir s'assurer que ces derniers sont satisfaits. Un profil de stratégies s sera donc un PNE si et seulement s'il satisfait les buts de ces joueurs.

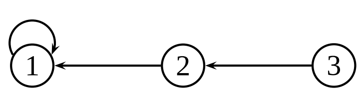
> *Preuve :*
> $\Rightarrow$ Supposons que s est un PNE et $\exists i$ tel que $RP_i = \{i\}$ et $s \models \neg\varphi_i$.
> Par définition d'un PNE, nous avons $s_{-i} \models \neg\varphi_i$. Mais nous savons que $RP_i = \{i\}$, donc $\forall v \in Var(s_{-i})$, $v \notin Var(\varphi_i)$. $s_{-i} \models \neg\varphi_i$ n'est alors pas possible. Nous avons une contradiction.
> $\Leftarrow$ Supposons à présent que pour tout i tel que $RP_i = \{i\}$ alors $s \models \varphi_i$, et s n'est pas un PNE.
> Si s n'est pas un PNE, alors $\exists i \in N$ tel que $s \models \neg\varphi_i$ et $s_{-i} \models \varphi_i$. Sachant que $s \models \neg\varphi_i$, on a donc $RP_i \neq \{i\}$. $\forall v \in \pi_i$, $v\neg \in Var(\varphi_i)$. Donc si $s \models \neg\varphi_i$, alors $s_{-i} \models \neg\varphi_i$ et s est un PNE ; nous avons une contradiction.

■

Exemple 2.7. *Soit $G = (N, V, \Gamma, \pi, \Phi)$ un jeu booléen défini par :*

* $V = \{a, b, c\}$, $N = \{1, 2, 3\}$,
* *pour tout i*, $\gamma_i = \top$,
* $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\pi_3 = \{c\}$,
* $\varphi_1 = a$, $\varphi_2 = \neg a$ *et* $\varphi_3 = b$.

Nous avons : $RV_1 = \{a\}$, $RV_2 = \{a\}$, $RV_3 = \{b\}$, $RP_1 = \{1\}$, $RP_2 = \{1\}$ *et* $RP_3 = \{2\}$.



Ce jeu a 4 PNEs : $\{abc, a\bar{b}c, ab\bar{c}, a\bar{b}\bar{c}\}$. *$s$ est un PNE si et seulement si* $s \models \varphi_1$.

Propriété 2.7. *Soit G un jeu booléen tel que les buts des joueurs soient tous des clauses. Alors G a toujours un équilibre de Nash en stratégies pures.*

De plus, s est un PNE si et seulement si pour tout $i \in N$ *tel que* $Lit(\varphi_i) \cap \pi_i \neq \emptyset$, $s \models \varphi_i$.

Si les buts des joueurs sont tous des clauses, un joueur peut donc soit satisfaire son but seul, s'il contrôle une variable de son but, soit il n'a aucune influence sur sa satisfaction. Tout profil de stratégies qui satisfait tous les joueurs contrôlant une variable de leur but est donc un équilibre de Nash en stratégies pures.

> *Preuve :* Soit $i \in N$. Si les buts de tous les joueurs sont des clauses, deux cas sont possibles :
> - $Lit(\varphi_i) \cap \pi_i \neq \emptyset$. Dans ce cas, i est sûr de pouvoir satisfaire son but, en affectant comme il le souhaite une des variables qu'il contrôle.
> - $Lit(\varphi_i) \cap \pi_i = \emptyset$. i ne peut rien faire pour satisfaire son but, il est totalement dépendant des autres joueurs.
>
> Il existe donc $s \in S$ tel qu'on ait pour tout $i \in N$ soit $s \models \varphi_i$, soit $s_{-i} \models \neg\varphi_i$. s est donc un PNE, et il satisfait pour tout $i \in N$ tel que $Lit(\varphi_i) \cap \pi_i \neq \emptyset$, $s \models \varphi_i$.
> Supposons qu'il existe $i \in N$ tel que $Lit(\varphi_i) \cap \pi_i \neq \emptyset$ et $s \models \neg\varphi_i$. Dans ce cas le joueur i a intéret à changer sa stratégie pour satisfaire son but, s n'est donc pas un PNE.

■

Propriété 2.8. *Soit G un jeu booléen tel que les buts des joueurs soient tous des cubes. G a toujours alors un équilibre de Nash en stratégies pures.*

Si les buts des joueurs sont tous des cubes, un joueur peut instancier les variables qu'il contrôle de façon à satisfaire son but au mieux, et il n'a pas d'influence sur le reste. Le profil de stratégies construit de cette façon est un PNE.

> *Preuve :* Soit $s \in S$ tel que pour tout $i \in N$, $s \models (\varphi_i)_{\pi_i}$. On a alors pour tout $i \in N$, soit $s \models \varphi_i$, soit $s_{-i} \models \neg\varphi_i$. En effet, si un joueur i a instancié toutes ses variables au mieux et que son but n'est quand même pas satisfait, c'est qu'un autre joueur contrôle une variable de son but, et l'instancie d'une façon non satisfaisante pour i.

■

Si pour un jeu G, la partie irréflexive d'un graphe de dépendance $\mathcal{P}$ est acyclique (c'est-à-dire qu'il n'y a pas de cycles de longueur ≥ 2), nous pouvons utiliser une procédure inspirée de la “forward sweep procedure” (procédure de recherche en avant) [Boutilier *et al.*, 1999] pour trouver un équilibre de Nash en stratégies pures. Regardons comment procéder sur un exemple.

Exemple 2.4 (page 37), suite : *La partie irréflexive du graphe de dépendance* $\mathcal{P}$ *de G, représenté figure 2.3 (page 38), est acyclique.*

$RP_1 = \{1\}$, *et donc un profil de stratégies* $s = (s_1, s_2, s_3)$ *sera un PNE seulement si le but du joueur* 1 *est satisfait, et donc si* $s_1 = a$.

2 *peut alors choisir sa stratégie, car son but dépend seulement de* 1 *et de lui-même* ($RP_2 = \{1,2\}$). *Donc s sera un PNE seulement si* $(s_1,s_2) \models (\varphi_2)_{s_1}$, *c'est-à-dire si* $s_2 = b$.

Enfin, le joueur 3 *a fait le même raisonnement que* 1 *et* 2, *et donc sait que son but ne sera jamais satisfait quoiqu'il fasse.*

Donc G a 2 *PNEs :* $\{abc, ab\overline{c}\}$.

Propriété 2.9. *Soit G un jeu booléen tel que la partie irréflexive du graphe de dépendance* $\mathcal{P}$ *de G soit acyclique. On sait alors que :*

1. *G a un PNE.*
2. *s est un PNE de G si et seulement si pour tout* $i \in N$,
   * *soit* $(s_{R^*(i)\setminus\{i\}}, s_i) \models \varphi_i$,
   * *soit* $s_{R^*(i)\setminus\{i\}} \models \neg\varphi_i$.

Preuve :

1. On peut supposer sans perte de généralité que les joueurs sont numérotés de sorte que pour tout $i \in N$, si i dépend de j, alors $j < i$. Soit $s = (s_1, \ldots, s_N)$ défini inductivement comme suit : pour $i = 1, \ldots, N$, si $(\varphi_i)_{s_1,\ldots,s_{i-1}}$ est satisfiable, alors on prend s_i tel que $(s_1, \ldots, s_i) \models \varphi_i$. Un tel s_i existe car i ne dépend pas de k pour tout $k > i$. Si $(\varphi_i)_{s_1,\ldots,s_{i-1}}$ n'est pas satisfiable, on prend alors un s_i quelconque.

   On a donc construit un profil de stratégies s tel que pour tout i, $s \models \varphi_i$ ou $s_{-i} \models \neg\varphi_i$. D'après la propriété 2.4 (page 42), s est donc un PNE.

2. Il nous reste à prouver que s est un PNE si et seulement si $\forall i \in N$, soit $(s_{R^*(i)\setminus\{i\}}, s_i) \models \varphi_i$, soit $s_{R^*(i)\setminus\{i\}} \models \neg\varphi_i$.

   $\Rightarrow$ Supposons que s est un PNE et qu'il existe un joueur $i \in N$ tel que $(s_{R^*(i)\setminus\{i\}}, s_i) \not\models \varphi_i$ et $s_{R^*(i)\setminus\{i\}} \not\models \neg\varphi_i$.
   Nous avons alors $s \models \neg\varphi_i$, et, comme $\forall k \notin R^*(i)$, $(\varphi_i)_{s_k} = \varphi_i$, nous avons $s_{-i} \not\models \neg\varphi_i$. D'après la propriété 2.4 (page 42), s n'est pas un PNE.

   $\Leftarrow$ Suposons à présent que $(s_{R^*(i)\setminus\{i\}}, s_i) \models \varphi_i$ or $s_{R^*(i)\setminus\{i\}} \models \neg\varphi_i$.
   Si i est un joueur tel que $(s_{R^*(i)\setminus\{i\}}, s_i) \models \varphi_i$, on a évidemment $s \models \varphi_i$.
   Si $s_{R^*(i)\setminus\{i\}} \models \neg\varphi_i$, alors, comme $\forall k \notin RP_i, (\varphi_i)_{s_k} = \varphi_i$, on a $s_{-i} \models \neg\varphi_i$.
   Comme $\forall i \in N$, soit $s \models \varphi_i$, soit $s_{-i} \models \neg\varphi_i$, s est un PNE.

■

Exemple 2.4 (page 37), suite : *Dans cet exemple, nous avons* $R^*(1) = \{1\}$, $R^*(2) = \{1,2\}$, $R^*(3) = \{1,2,3\}$. *Pour* $s = abc$, *on a bien*

* $(s_{R^*(1)\setminus\{1\}}, s_1) = s_1 \models \varphi_1 = a$
* $(s_{R^*(2)\setminus\{2\}}, s_2) = (s_1, s_2) \models \varphi_2 = a \leftrightarrow b$
* $s_{R^*(3)\setminus\{3\}} = (s_1, s_2) \models \neg\varphi_3 = a \vee \neg b \vee \neg c$

Le raisonnement est le même pour $s = ab\overline{c}$.

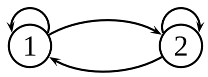
Quand la partie irréflexive du graphe de dépendance $\mathcal{P}$ de G n'est pas acyclique, l'existence d'un équilibre de Nash en stratégies pures n'est plus garanti, comme le montre l'exemple suivant.

Exemple 2.8. *Soit $G = (N,V,\pi,\Gamma,\Phi)$ un jeu booléen défini par :*

- $V = \{a,b\}$, $N = \{1,2\}$,
- $\pi_1 = \{a\}$, $\pi_2 = \{b\}$,
- $\forall i$, $\gamma_i = \top$,
- $\varphi_1 = a \leftrightarrow b$,
- $\varphi_2 = (a \wedge \neg b) \vee (\neg a \wedge b)$.

On a $RV_1 = \{a,b\}$, $RV_2 = \{a,b\}$, $RP_1 = \{1,2\}$, $RP_2 = \{1,2\}$.

Le graphe de dépendance $\mathcal{P}$ de G est le suivant :



Ce jeu n'a aucun PNE.

La propriété 2.9 (page ci-contre) entraîne le corollaire suivant :

Corollaire 2.2. *Si G est un jeu booléen tel qu'il existe un $j \in N$ tel que pour tous les $i \in N$, $RP_i = \{j\}$, alors s est un PNE si et seulement si $s \models \varphi_j$.*

Si tous les joueurs ne dépendent que d'un et un seul joueur j, tous les joueurs (sauf j) nont aucune influence sur leurs buts. Seul j, qui contrôle son but, a une préférence sur ses stratégies, et donc une influence sur le choix des PNEs.

Preuve : Soit G est un jeu booléen tel que $\exists j \in N$ tel que $\forall i \in N$ $RP_i = \{j\}$.

$\Rightarrow$ Soit s un PNE de G. Comme, $\forall i \in N$, $RP_i = \{j\}$, alors $\forall i \in N$, $R^*(i) = \{j\}$. D'après la propriété 2.9 (page précédente), on doit avoir $s_j \models \varphi_j$, et pour tout $i \neq j$, on peut avoir soit $(s_j, s_i) \models \varphi_i$, soit $s_j \models \neg\varphi_i$. La seule condition est donc que $s \models \varphi_j$.

$\Leftarrow$ Soit $s \models \varphi_j$. Si $\forall i$, $RP_i = \{j\}$, on sait que i n'a aucun pouvoir sur son but, et donc que s est équivalent à s_j. Donc, on a pour tout i, soit $s \models \varphi_i$, soit $s_{-i} \models \neg\varphi_i$, et s est un PNE.

■

Propriété 2.10. *Soit $G = (N,V,\pi,\Gamma,\Phi)$ un jeu booléen, $B \subseteq N$ un ensemble stable pour R, et $G_B = (B, V_B, \pi_B, \Gamma_B, \Phi_B)$ la projection de G sur B.*

Si s est un PNE pour G, alors s_B est un PNE pour G_B.

Preuve : Soit s un PNE de G. Alors, $\forall i \in N$, soit $s \models \varphi_i$, soit $s_{-i} \models \neg\varphi_i$.
Nous savons que $s = (s_B, s_{-B})$, et nous voulons vérifier que s_B est un PNE de G_B, c'est-à-dire que $\forall i \in B$, soit $s_B \models \varphi_i$, soit $s_{B-i} \models \neg\varphi_i$. Soit $i \in B$.

- $s = (s_B, s_{-B}) \models \varphi_i$. Comme $i \in B$, et que B est stable, nous savons que $\forall j \in R(i)$, $j \in B$. Donc tous les joueurs qui ont une influence sur i sont dans B, et donc s_{-B} n'a aucune influence sur φ_i : on a $s_B \models \varphi_i$.
- $s_{-i} \models \neg\varphi_i$. Comme précédemment, on sait que seuls les joueurs de B ont une influence sur φ_i. On a donc $s_{(B\setminus\{i\})} \models \neg\varphi_i$, ce qui correspond à $s_{B-i} \models \neg\varphi_i$.

■

Exemple 2.9. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen défini par*

- $V = \{a, b, c, d\}$, $N = \{1, 2, 3, 4\}$,
- $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\pi_3 = \{c\}$, $\pi_4 = \{d\}$,
- $\forall i$, $\gamma_i = \top$,
- $\varphi_1 = a \leftrightarrow b$, $\varphi_2 = b \leftrightarrow c$, $\varphi_3 = \neg d$, *et* $\varphi_4 = d \leftrightarrow (b \wedge c)$.

On a : $RP_1 = \{1, 2\}$, $RP_2 = \{2, 3\}$, $RP_3 = \{4\}$, $RP_4 = \{2, 3, 4\}$.

Le graphe de dépendance $\mathcal{P}$ de G est représenté figure 2.5.

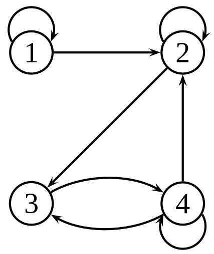


Figure 2.5 — Graphe de dépendance de l'exemple 2.9

L'ensemble de joueurs $B = \{2, 3, 4\}$ est stable. $G_B = (B, V_B, \Gamma_B, \pi_B, \Phi_B)$ est un jeu booléen, avec

- $V_B = \{b, c, d\}$,
- $\pi_2 = b$, $\pi_3 = c$, $\pi_4 = d$,
- $\gamma_2 = \gamma_3 = \gamma_4 = \top$,
- $\varphi_2 = b \leftrightarrow c$, $\varphi_3 = \neg d$, *et* $\varphi_4 = d \leftrightarrow (b \wedge c)$.

G a 2 PNEs : $\{abcd, \overline{a}\overline{b}\overline{c}\overline{d}\}$, et $\{bcd, \overline{b}\overline{c}\overline{d}\}$ sont les 2 PNEs de G_B (et dans cet exemple, G_B n'a aucun autre PNE).

Comme il est possible de le voir dans l'exemple 2.5 (page 40), la réciproque n'est pas toujours vraie : $C = \{3\}$ est stable, et le jeu booléen $G_C = (C, V_C, \pi_C, \Gamma_C, \Phi_C)$ a un équilibre de Nash en stratégies pures : $\{\overline{c}\}$, mais pourtant le jeu G n'a aucun PNE.

Toutefois, il existe des cas simples pour lesquels la réciproque est vraie.

Propriété 2.11. *Soit B et C deux ensembles stables de joueurs, et soit G_B et G_C deux jeux booléens associés.*

Si s_B est un PNE pour G_B et s_C un PNE pour G_C tels que[11] $\forall i \in B \cap C$, $s_{B,i} = s_{C,i}$, alors $s_{B \cup C}$ est un PNE pour le jeu $G_{B \cup C}$.

> *Preuve :* B et C sont stables, donc d'après la propriété 2.2 (page 39), $B \cup C$ est un ensemble stable, et d'après la propriété 2.3 (page 40), $G_{B \cup C}$ est un jeu booléen.
> Soit $i \in B \cup C$. 3 cas sont possibles (2 étant symétriques) :
> - $i \in B \setminus C$ (ou $i \in C \setminus B$). s_B est un PNE pour G_B, donc on a :
>   - soit $s_B \models \varphi_i$, et comme $i \notin C$, on a $(s_B, s_C) \models \varphi_i$;
>   - soit $s_{B-i} \models \neg\varphi_i$. Comme $i \notin C$, on a également $(s_{B-i}, s_C) \models \neg\varphi_i$, ce qui correspond à $s_{B \cup C - i} \models \neg\varphi_i$.

[11] $s_{B,i}$ représente la stratégie du joueur i pour le jeu G_B.

* $i \in B \cap C$. On a alors $\forall k \in B \cap C$, $s_{B,k} = s_{C,k}$. Plusieurs cas sont possibles :
    * $s_B \models \varphi_i$ et $s_C \models \varphi_i$. Puisque $\forall k \in B \cap C$, $s_{B,k} = s_{C,k}$, on a $(s_B, s_C) \models \varphi_i$, donc $s_{B \cup C} \models \varphi_i$.
    * $s_{B-i} \models \neg\varphi_i$ et $s_{C-i} \models \neg\varphi_i$. Pour les mêmes raisons que précédemment, on a $(s_{B-i}, s_{C-i}) \models \neg\varphi_i$, et donc $s_{B \cup C-i} \models \neg\varphi_i$.
    * $s_B \models \varphi_i$ et $s_{C-i} \models \neg\varphi_i$ (ou vice versa). On sait que $\forall k \in B \cap C$, $s_{B,k} = s_{C,k}$, et puisque les résultats obtenus par B et C sont différents, ces stratégies n'interviennent pas dans φ_i. Les joueurs dans $B \cap C$ ne contrôlent aucune variable de φ_i : $\forall v \in Var(\varphi_i)$, $\forall k \in B \cap C$, $v \notin RV_k$. Dans ce cas, puisque $s_B \models \varphi_i$, on sait que toutes les variables de φ_i sont contrôlées par des joueurs dans $B \setminus C$; et puisque $s_{C-i} \models \neg\varphi_i$, on sait que toutes les variables de φ_i sont contrôlées par des joueurs dans $C \setminus B$. Comme chaque variable est contrôlée par un et un seul joueur, c'est impossible.

  On sait donc que soit $s_{B \cup C} \models \varphi_i$, soit $s_{B \cup C-i} \models \neg\varphi_i$. $s_{B \cup C}$ est un équilibre de Nash de $G_{B \cup C}$.

■

Exemple 2.10. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen défini par*

* $V = \{a, b, c\}$, $N = \{1, 2, 3\}$,
* $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\pi_3 = \{c\}$,
* $\forall i$, $\gamma_i = \top$,
* $\varphi_1 = a \leftrightarrow c$, $\varphi_2 = b \leftrightarrow \neg c$, *et* $\varphi_3 = c$.

On a $RP_1 = \{1, 3\}$, $RP_2 = \{2, 3\}$, $RP_3 = \{3\}$. Le graphe de dépendance $\mathcal{P}$ de G est représenté figure 2.6.

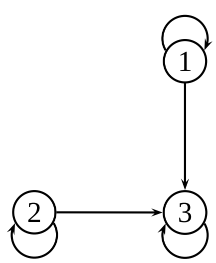


Figure 2.6 **—** Graphe de dépendance de l'exemple 2.10

L'ensemble des joueurs $A = \{1, 3\}$ et $B = \{2, 3\}$ sont stables. On a deux nouveaux jeux booléens :

* $G_A = (A, V_A, \Gamma_A, \pi_A, \Phi_A)$, *avec*
    * $A = \{1, 3\}$, $V_A = \{a, c\}$,
    * $\pi_1 = a$, $\pi_3 = c$,
    * $\gamma_1 = \gamma_3 = \top$,
    * $\varphi_1 = a \leftrightarrow c$ *et* $\varphi_3 = c$.

  G_A *a un PNE :* $\{ac\}$ *(dénoté par* $s_A = (s_{A,1}, s_{A,3})$*).*
* $G_B = (B, V_B, \pi_B, \Gamma_B, \Phi_B)$, *avec*
    * $B = \{2, 3\}$, $V_B = \{b, c\}$,

- $\pi_2 = b$, $\pi_3 = c$,
- $\gamma_1 = \gamma_3 = \top$,
- $\varphi_2 = b \leftrightarrow \neg c$, $\varphi_3 = c$.

G_B a un PNE : $\{\overline{b}c\}$ (dénoté par $s_B = (s_{B,2}, s_{B,3})$).

$A \cap B = \{3\}$. Mais nous avons $s_{A,3} = s_{B,3} = c$: $G_{A\cup B}$ a un PNE : $\{a\overline{b}c\}$.

On peut facilement généraliser la propriété 2.11 (page 48), avec p ensembles stables couvrant l'ensemble de tous les joueurs :

Propriété 2.12. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen, et soit $B_1 \dots B_p$ p ensembles stables de joueurs, tels que $B_1 \cup \dots \cup B_p = N$. Soit $G_{B_1}, \dots, G_{B_p}$ les p jeux booléens associés.*

Si $\exists s_{B_1} \dots s_{B_p}$ des PNEs de $G_{B_1}, \dots, G_{B_p}$ tels que $\forall i, j \in \{1, \dots p\}$, $\forall k \in B_i \cap B_j$, $s_{B_i,k} = s_{B_j,k}$, alors $s = (s_{B_1}, \dots, s_{B_p})$ est un PNE de G.

> *Preuve :* Soit B_1 et B_2 les deux premiers ensembles stables, et G_{B_1} et G_{B_2} les deux jeux booléens associés. On peut appliquer la propriété 2.11 (page 48), et montrer ainsi que $B_1 \cup B_2$ est un ensemble stable, que $G_{B_1\cup B_2}$ est un jeu booléen, et que $s_{B_1\cup B_2}$ est un PNE de $G_{B_1\cup B_2}$.
> On peut faire de même pour $B_1 \cup B_2$ et B_3, et ainsi de suite, jusqu'à trouver le résultat voulu.

■

Comme montré dans l'exemple 2.10 (page précédente), diviser un jeu booléen permet de faciliter le calcul des équilibres de Nash.

Si on essaie de calculer les équilibres de Nash dans le jeu original, nous devons vérifier soit $s \models \varphi_i$, soit $s_{-i} \models \neg\varphi_i$ pour chacun des 8 profils de stratégies s, et pour chacun des 3 joueurs i. Il faut donc faire 12 vérifications par joueur (8 pour chaque profil de stratégies pour vérifier $s \models \varphi_i$, et 4 pour chaque s_{-i} pour vérifier $s_{-i} \models \neg\varphi_i$), et donc 36 pour le jeu dans le pire des cas.

Le calcul des équilibres de Nash une fois le jeu divisé est beaucoup plus facile : d'après la propriété 2.9 (page 46), le calcul du PNE de G_B nécessite 6 vérifications pour le joueur 1 (4 pour calculer $(s_1, s_3) \models \varphi_1$, et 2 pour calculer $s_3 \models \neg\varphi_1$) ; et seulement 2 pour le joueur 3 (car $R^*(3) \setminus \{3\} = \varnothing$). On a donc à faire seulement 8 vérifications dans le pire des cas pour trouver les PNEs de G_B, et il en est de même pour G_C, qui a une configuration équivalente. Comme il faut également vérifier que l'instanciation des variables du joueur 3 sont les mêmes pour les 2 jeux, il faut faire 17 vérifications pour calculer les PNEs du jeu G.

2.4.2 Stratégies dominées

Comme nous l'avons vu dans le chapitre 1, section 1.3.2 (page 21), il arrive parfois que le calcul des équilibres de Nash donne des résultats mal adaptés. Nous allons donc étudier ici le principe de *dominance*. Dans le cadre des jeux booléens, la définition des stratégies dominées est la même que dans la littérature (Définitions 1.8 et 1.9 (page 22)).

L'idée principale de ce concept est que les stratégies dominées ne sont pas utiles et peuvent être éliminées itérativement, jusqu'à ce qu'un point fixe soit atteint. Ce processus repose sur l'hypothèse que chaque joueur se comporte de manière rationnelle et sait que les autres joueurs sont rationnels.

Comme nous l'avons vu dans la section 1.3.2 (page 21), une stratégie strictement dominée ne peut jamais être présente dans un équilibre de Nash, tandis qu'une stratégie faiblement dominée peut apparaitre dans un équilibre de Nash. Ce résultat reste valable dans le cas simple des jeux booléens.

Propriété 2.13.

* *Une stratégie strictement dominée n'est présente dans aucun équilibre de Nash.*
* *Une stratégie faiblement dominée peut être présente dans un équilibre de Nash.*

Preuve :

* Montrons qu'une stratégie strictement dominée n'est présente dans aucun équilibre de Nash. Soit s_i une stratégie du joueur i strictement dominée. Alors

$$\exists s'_i \in S_i \; telle\ que \; \forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) < u_i(s'_i, s_{-i})$$

  Donc, pour tout profil de stratégies $s = \{s_1, \ldots, s_i, \ldots, s_n\}$, on aura : $u_i(s) < u_i(s'_i, s_{-i})$.
  Or, s_i appartiendra à un équilibre de Nash si et seulement si il existe s tel que :

$$\forall s'_{-i} \in S_i, u_i(s) > u_i(s'_i, s_{-i})$$

  ce qui est en contradiction avec l'hypothèse précédente. s_i ne peut donc pas appartenir à un équilibre de Nash.
* Montrons à présent sur un contre-exemple qu'une stratégie faiblement dominée peut être présente dans un équilibre de Nash. Pour cela, étudions le jeu booléen $G = (N, V, \pi, \Gamma, \Phi)$ avec $V = \{a, b\}$, $N = \{1, 2\}$, pour tout i, $\gamma_i = \top$, $\pi_1 = \{a\}$, $\pi_2 = \{b\}$ et $\varphi_1 = \varphi_2 = a \wedge \neg b$.
  Ce jeu a deux équilibres de Nash : les profils de stratégies $a\overline{b}$ et $\overline{a}b$. Or, la stratégie $\neg a$, présente dans l'équilibre de Nash $\overline{a}b$, est faiblement dominée par la stratégie a.

■

La propriété suivante est également issue de la théorie classique des jeux.

Propriété 2.14.

* *L'ordre d'élimination des stratégies strictement dominées n'affecte pas le résultat final.*
* *L'ordre d'élimination des stratégies faiblement dominées peut affecter le résultat final.*

Le premier point de cette propriété, sur la dominance stricte, est évidemment applicable dans le cas simple des jeux booléens [Hillas et Kohlberg, 2002].

On aurait pu croire que l'ordre d'élimination des stratégies faiblement dominées dans le cas simple des jeux booléens n'affecte pas le résultat final, mais il n'en n'est rien. Nous en donnons ici un contre-exemple.

Exemple 2.11. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen avec*

- $V = \{a, b\}$, $N = \{1, 2\}$,
- *pour tout i,* $\gamma_i = \top$,
- $\pi_1 = \{a\}$, $\pi_2 = \{b\}$,
- $\varphi_1 = a \wedge b$ *et*
- $\varphi_2 = a \wedge \neg b$

On peut à présent construire la forme normale de G :

1 \ 2	$\overline{b}$	b
$\overline{a}$	(0, 0)	(0, 0)
a	(0, 1)	(1, 0)

Etudions les stratégies dominées de ce jeu :

- *Pour le joueur* 1*, la stratégie a domine faiblement la stratégie $\overline{a}$.*
- *Pour le joueur* 2*, la stratégie $\overline{b}$ domine faiblement la stratégie b.*

Eliminons dans un premier temps la stratégie dominée du joueur 1*. On élimine donc $\overline{a}$. Après cette élimination, la stratégie $\overline{b}$ du joueur* 2 *domine faiblement la stratégie b. On obtient donc une seule stratégie résultat : $a\overline{b}$.*

Eliminons à présent la stratégie dominée du joueur 2 *en premier. On élimine donc la stratégie b. Après cette élimination, le joueur* 1 *n'a plus de stratégie dominée. On obtient donc deux stratégies résultat : $a\overline{b}$ et $\overline{a}\overline{b}$.*

On voit ici que nous n'avons pas obtenu le même résultat suivant l'ordre d'élimination des stratégies faiblement dominées.

Reprenons l'exemple 2.6 (page 41) pour calculer l'élimination des stratégies dominées après avoir calculé les équilibres de Nash.

Exemple 2.6 – suite *Etudions les stratégies dominées de ce jeu :*

- *Pour le joueur* 1*, la stratégie $\overline{a}$ domine faiblement la stratégie a. Eliminons cette stratégie.*
- *Une fois cette stratégie éliminée, la stratégie c du joueur* 3 *domine faiblement sa stratégie $\overline{c}$. On élimine donc cette dernière.*
- *Enfin, pour le joueur* 2*, la stratégie $\overline{b}$ domine faiblement la stratégie b. On élimine également cette stratégie.*

On obtient ainsi un seul profil de stratégies résultat, $\overline{a}\overline{b}c$ qui permet aux joueurs 1 *et* 2 *de gagner.*

Nous avons vu que l'ordre d'élimination des stratégies faiblement dominées a un impact sur le résultat. On est donc en droit de se demander si on aurait pu trouver un autre résultat en éliminant les stratégies des joueurs dans un ordre différent.

Ici, c'est le seul ordre d'élimination possible. En effet, le joueur 1 *est le seul joueur qui a une stratégie dominante avant élimination d'autres stratégies. Et, une fois la stratégie dominée du joueur* 1 *éliminée, le joueur* 3 *est également le seul à avoir une stratégie dominante.*

Une seconde propriété se dégage immédiatement :

Propriété 2.15. *Dans un jeu booléen, si la stratégie s_i du joueur i domine strictement une autre stratégie s'_i, alors s_i est une stratégie gagnante pour ce joueur.*

> *Preuve :* s_i domine strictement s'_i. Donc,
>
> $$\begin{array}{rl} & \forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) > u_i(s'_i, s_{-i}) \\ \Leftrightarrow & \forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) = 1 \text{ et } u_i(s'_i, s_{-i}) = 0 \\ \Rightarrow & \forall s_{-i} \in S_{-i}, (s_i, s_{-i}) \models \varphi_i \\ \Rightarrow & s_i \text{ est une stratégie gagnante pour le joueur } i. \end{array}$$

■

L'utilisation de la notion de projection (voir le chapitre Notations (page 7)) nous permet ici aussi de donner une caractérisation logique des stratégies strictement dominées.

Propriété 2.16. *Soit G un jeu booléen. La stratégie s_i du joueur i domine strictement la stratégie s'_i si et seulement si :*

$$s_i \models (\forall -i : \varphi_i) \textit{ et } s'_i \models (\forall -i : \neg\varphi_i)$$

> *Preuve :* Supposons que la stratégie s_i du joueur i domine strictement s'_i.
>
> $$\begin{array}{rl} & \forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) > u_i(s'_i, s_{-i}) \\ \Leftrightarrow & \forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) = 1 \text{ et } u_i(s'_i, s_{-i}) = 0 \\ \Leftrightarrow & \forall s_{-i} \in S_{-i}, (s_i, s_{-i}) \models \varphi_i \text{ et } (s'_i, s_{-i}) \models \neg\varphi_i \\ \Leftrightarrow & s_i \models (\forall -i : \varphi_i) \text{ et } s'_i \models (\forall -i : \neg\varphi_i) \end{array}$$

■

Etudions cette caractérisation sur un exemple.

Exemple 2.12. *Soit $G = (N, V, \pi, \Gamma, \Phi)$ un jeu booléen. On donne*

* $V = \{a, b, c\}$, $N = \{1, 2\}$,
* *pour tout i,* $\gamma_i = \top$,
* $\pi_1 = \{a, b\}$, $\pi_2 = \{c\}$,
* $\varphi_1 = (\neg a \wedge \neg b \wedge c) \vee (a \wedge b)$, *et*
* $\varphi_2 = (a \wedge c) \vee (\neg b \wedge \neg c)$

On peut à présent construire la forme normale de G :

1 \ 2	$\overline{c}$	c
$\overline{a}\overline{b}$	*(0, 1)*	*(1, 0)*
$a\overline{b}$	*(0, 1)*	*(0, 1)*
$\overline{a}b$	*(0, 0)*	*(0, 0)*
ab	*(1, 0)*	*(1, 1)*

Etudions les stratégies dominées du joueur 1 *en utilisant les caractérisations de la propriété 2.16. Pour cela, commençons par calculer les formules qui nous seront utiles.*

- $\varphi_1 = (\neg a \wedge \neg b \wedge c) \vee (a \wedge b)$
- $\neg\varphi_1 = (a \vee b \vee \neg c) \wedge (\neg a \vee \neg b)$
- $\forall - 1 : \varphi_1 = a \wedge b = \varphi_1'$
- $\forall - 1 : \neg\varphi_1 = (a \vee b) \wedge (\neg a \vee \neg b) = \varphi_1''$

Nous devons ensuite étudier chaque stratégie du joueur 1 *pour voir si elle permet d'inférer* φ_1' *ou* φ_1''.

- $\overline{a}\overline{b} \not\models \varphi_1'$ *et* $\overline{a}\overline{b} \not\models \varphi_1''$
- $a\overline{b} \not\models \varphi_1'$ *et* $a\overline{b} \models \varphi_1''$
- $\overline{a}b \not\models \varphi_1'$ *et* $\overline{a}b \models \varphi_1''$
- $ab \models \varphi_1'$ *et* $ab \not\models \varphi_1''$

On peut à présent calculer les stratégies strictement dominées de 1.

ab domine strictement $a\overline{b}$ *et* $\overline{a}b$. *En effet, on a bien :*

- *pour* $a\overline{b}$: $ab \models \varphi_1'$ *et* $a\overline{b} \models \varphi_1''$
- *pour* $\overline{a}b$: $ab \models \varphi_1'$ *et* $\overline{a}b \models \varphi_1''$

Essayons à présent de caractériser les stratégies faiblement dominées. Cette fois ce n'est pas la notion de projection qui est utilisée mais celle d'interprétation partielle.

Propriété 2.17. *Soit G un jeu booléen. La stratégie s_i du joueur i domine faiblement la stratégie s_i' si et seulement si :*

$$(\varphi_i)_{s_i'} \models (\varphi_i)_{s_i} \text{ et } (\varphi_i)_{s_i} \not\models (\varphi_i)_{s_i'}$$

$(\varphi_i)_{s_i'} \models (\varphi_i)_{s_i}$ signifie que $\forall s_{-i}$, si l'instanciation partielle de φ_i par s_i' est satisfaite, alors il en est de même pour celle de φ_i par s_i. Donc, si $\forall s_{-i}$, (s_i', s_{-i}) permet de satisfaire φ_i, alors (s_i, s_{-i}) également.

$(\varphi_i)_{s_i} \not\models (\varphi_i)_{s_i'}$ signifie que $\exists s_{-i}$ tel que (s_i, s_{-i}) satisfait φ_i, tandis (s_i', s_{-i}) non.

Preuve : La stratégie s_i du joueur i domine faiblement s_i' si et seulement si :

$$\forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i}) \tag{2.6}$$
$$\text{et} \quad \exists s-i \in S_{-i}, u_i(s_i, s_{-i}) > u_i(s_i', s_{-i}) \tag{2.7}$$

L'équation 2.6 nous permet d'obtenir les équivalences suivantes :

$$\begin{aligned}
2.6 \quad &\Leftrightarrow \quad \forall s_{-i} \in S_{-i}, (u_i(s_i, s_{-i}) = 1 \text{ ou } u_i(s_i', s_{-i}) = 0) \\
&\Leftrightarrow \quad \forall s_{-i} \in S_{-i}, (s_i, s_{-i}) \models \varphi_i \text{ ou } (s_i', s_{-i}) \models \neg\varphi_i \\
&\Leftrightarrow \quad \forall s-i \in S_{-i}, \text{ si } (s_i', s_{-i}) \models \varphi_i \text{ alors } (s_i, s_{-i}) \models \varphi_i \\
&\Leftrightarrow \quad \forall s-i \in S_{-i}, \text{ si } s_{-i} \models (\varphi_i)_{s_i'} \text{ alors } s_{-i} \models (\varphi_i)_{s_i} \\
&\Leftrightarrow \quad (\varphi_i)_{s_i'} \models (\varphi_i)_{s_i}
\end{aligned}$$

Ensuite, il est possible de remarquer que l'on a : (2.7) $\Leftrightarrow \neg$(2.6) si l'on intervertit s_i et s_i'. Donc :

$$2.7 \Leftrightarrow (\varphi_i)_{s_i} \not\models (\varphi_i)_{s_i'}$$

∎

2.5 Cas particulier : Jeux à deux joueurs et à somme nulle

Dans cette section, on montre que les jeux booléens étudiés dans [Harrenstein *et al.*, 2001; Harrenstein, 2004a] qui nous ont inspirés au départ, sont un cas particulier des jeux booléens à n joueurs présentés dans la section 2.2 (page 33). Les définitions sont les mêmes, mis à part la notion de coalition qui n'a aucun sens ici.

Certains paramètres sont simplifiés. En effet, comme chaque variable de décision de V ne peut être contrôlée que par un seul des deux joueurs, on a $\pi_2 = V \setminus \pi_1$. D'autre part, on a $\varphi_2 \equiv \neg\varphi_1$ et $\gamma_1 = \gamma_2 = \top$.

Etudions à présent un exemple simple.

Exemple 2.13. *Soit* $V = \{a,b,c,d\}$*,* $N = \{1,2\}$*,* $\pi_1 = \{a,c\}$ *et* $\varphi_1 = (a \wedge \neg b) \vee (b \wedge d)$*.*

Le jeu booléen $G = (N,V,\pi_1,\Gamma,\Phi)$ *est totalement défini.*

En effet, on sait que $\pi_2 = V \setminus \pi_1$*, que* $\varphi_2 = \neg\varphi_1 = (\neg a \vee b) \wedge (\neg b \vee \neg d)$*, et que* $\gamma_1 = \gamma_2 = \top$*.*

On peut à présent construire la forme normale de G[12] *:*

1 \ 2	$\overline{b}\overline{d}$	$b\overline{d}$	$\overline{b}d$	bd
$\overline{a}\overline{c}$	0	0	0	1
$a\overline{c}$	1	0	1	1
$\overline{a}c$	0	0	0	1
ac	1	0	1	1

On constate ici que le joueur 2*, qui contrôle les variables* b *et* d*, a une stratégie gagnante. En effet, s'il choisit de mettre* b *à vrai et* d *à faux, il est sûr de gagner quels que soient les choix de* 1*.*

La propriété suivante donne une caractérisation simple des équilibres de Nash en stratégies pures dans les jeux booléens à deux joueurs et à somme nulle qui est obtenue par une simple transposition de résultats issus de la théorie des jeux à somme nulle.

Propriété 2.18. *Si G est un jeu booléen à deux joueurs et à somme nulle,* $s = (s_1,s_2)$ *est un équilibre de Nash en stratégies pures si et seulement si* s_1 *est une stratégie gagnante pour le joueur* 1*, ou* s_2 *est une stratégie gagnante pour le joueur* 2*.*

Preuve :

1. Soit $s = (s_1,s_2)$ un équilibre de Nash en stratégies pures.
   * Supposons que l'on a $u_1(s) = 1$. Le jeu étant à somme nulle, on a $u_2(s) = 0$. Comme s est un équilibre de Nash, $\forall s'_2, u_2(s) \geq u_2(s_1,s'_2)$ ce qui entraîne $\forall s'_2, u_2(s_1,s'_2) = 0$. Donc, $\forall s'_2, (s_1,s'_2) \models \neg\varphi_2$, ce qui implique que $\forall s'_2, (s_1,s'_2) \models \varphi_1$. s_1 est donc une stratégie gagnante pour le joueur 1.

[12] 1 signifie que 1 gagne et 2 perd, et vice-versa pour 0

  * On a $u_1(s) = 0$ et $u_2(s) = 1$. En faisant le même raisonnement que précédemment, on montre que s_2 est donc une stratégie gagnante pour le joueur 2.

2. Supposons que s_1 est une stratégie gagnante pour le joueur 1. On a donc : $\forall s_2, u_1(s_1, s_2) = 1$ et $\forall s_2, u_2(s_1, s_2) = 0$. Posons $s = (s_1, s_2)$. On a bien $\forall s'_1, u_1(s) \geq u_1(s'_1, s_2)$ et $\forall s'_2, u_2(s) \geq u_2(s_1, s'_2)$. s est donc bien un équilibre de Nash.

   On raisonne de la même façon si l'on suppose que s_2 est une stratégie gagnante pour le joueur 2.

■

Par conséquent, dans un jeu booléen à deux joueurs et à somme nulle, il existe un équilibre de Nash en stratégies pures si et seulement si l'un des deux joueurs a une stratégie gagnante.

La propriété 2.18 (page précédente) permet de déterminer facilement la complexité algorithmique du problème d'existence d'un équilibre de Nash en stratégies pures. On rappelle que $\Sigma_2^p = \mathsf{NP}^{\mathsf{NP}}$ est la classe des langages reconnaissables en temps polynomial par une machine de Turing non-déterministe munie d'oracles NP (voir [Papadimitriou, 1994a]). Les résultats de complexité suivants ont été trouvés avec l'aide de Bruno Zanuttini.

Propriété 2.19. *Le problème de l'existence d'un équilibre de Nash en stratégies pures dans un jeu booléen à deux joueurs et à somme nulle est Σ_2^p-complet.*

Preuve : L'appartenance à Σ_2^p est immédiate. La difficulté est obtenue par la réduction du problème consistant à décider la validité de $\mathtt{QBF}_{2,\exists}$. Etant donné une formule $Q = \exists A, \forall B \Phi$, où A et B sont des ensembles disjoints de variables et Φ est une formule propositionnelle de $L_{A \cup B}$, on définit le jeu booléen à 2 joueurs et à somme nulle suivant : $\varphi_1 = \Phi \vee (x \leftrightarrow y)$, où x, y sont des nouvelles variables (*i.e.* $x, y \notin A \cup B$), $\varphi_2 = \neg\varphi_1$, $\pi_1 = A \cup \{x\}$, $\pi_2 = B \cup \{y\}$ et $\gamma_1 = \gamma_2 = \top$. Ce jeu peut être évidemment construit en temps polynomial étant donné Q. D'après la propriété 2.18 (page précédente), ce jeu a un équilibre de Nash si et seulement si un des joueurs 1 ou 2 a une stratégie gagnante.
Supposons que Q est valide. Soit $M_A \in 2^A$ un modèle de Q. Alors, (M_A, x) et $(M_A, \overline{x})$ sont des stratégies gagnantes pour 1.
Supposons maintenant que Q n'est pas valide ; alors quel que soit $M_A \in 2^A$ que 1 choisit de jouer, 2 peut jouer $M_B \in 2^B$ tel que $(M_A, M_B) \not\models \Phi$. De plus, si 1 joue x (resp. $\overline{x}$), alors 2 peut jouer $\overline{y}$ (resp. y), avec dans les deux cas la victoire de 2. D'autre part, 2 n'a pas non plus de stratégie gagnante puisque 1 peut toujours rendre $x \leftrightarrow y$ vrai, et donc gagner.
Il existe donc une stratégie gagnante pour 1 (ou 2, au choix) si et seulement si Q est valide.

■

On en tire le corollaire suivant, concernant cette fois les jeux booléens à n joueurs.

Corollaire 2.3. *Le problème de l'existence d'un équilibre de Nash en stratégies pures dans un jeu booléen à n joueurs est Σ_2^p-complet.*

> *Preuve :* L'appartenance à Σ_2^p est immédiate ; la difficulté est obtenue, dès que $n = 2$ et que le jeu est à somme nulle par le théorème 2.19 (page précédente).

■

Ce résultat est à rapprocher de la complexité de la contrôlabilité – également un problème Σ_2^p-complet [Lang et Marquis, 1998b]. Or, le problème de l'existence d'un équilibre de Nash dans un jeu à plusieurs joueurs et à somme non nulle étant bien plus général que la contrôlabilité, le fait qu'il ne soit pas situé plus haut dans la hiérarchie des classes de complexité est plutôt une bonne nouvelle.

On peut expliquer intuitivement le fait que le problème de l'existence d'un équilibre de Nash soit au second niveau de la hiérarchie polynomiale par le fait que la résolution de ce problème comporte deux sources indépendantes de complexité NP-difficiles : la recherche du "bon" profil de stratégies, et la vérification qu'il constitue un équilibre de Nash en stratégies pures. Par comparaison, l'existence d'un profil de stratégies dont l'utilité cumulée est supérieure à une borne donnée est seulement un problème NP-complet.

Nous allons à présent essayer de simplifier le problème en étudiant les restrictions syntaxiques sur les formules représentant les buts des joueurs. Nous nous intéressons particulièrement aux formules en forme normale disjonctive (DNF). Rappelons que toute formule booléenne peut être mise en DNF, et que c'est donc une restriction syntaxique et non sémantique.

Tant que l'on considère des jeux à 2 joueurs et à somme nulle, et sachant que décider la validité de $\exists A, \forall B, \Phi$, une formule $\texttt{QBF}_{2,\exists}$, est Σ_2^p-complet même si Φ est en DNF, la propriété 2.19 (page ci-contre) reste correcte même si le but du joueur 1 est en DNF (le but du joueur 2 étant implicite).

Toutefois, lorsque nous représentons explicitement les buts de chaque joueur en DNF, la complexité du problème descend en NP, comme le montre la propriété suivante :

Propriété 2.20. *Soit G un jeu booléen. Si $\forall i \in N$, φ_i est en DNF, alors le problème de l'existence d'un équilibre de Nash en stratégie pure est NP-complet.*

> *Preuve :*
> Si φ_i est en DNF, alors $\exists i : \varphi_i$ peut être calculé en temps linéaire [Lang *et al.*, 2003, Propositions 17–18]. Alors, si chaque φ_i est en DNF, la formule $\psi \equiv \bigwedge_i(\varphi_i \vee (\neg\exists i : \varphi_i))$ peut être calculée en temps linéaire. Comme la propriété 2.4 (page 42) permet de trouver un profil de stratégies s et de vérifier que $s \models \psi$, le problème est en NP.
> La difficulté est obtenue par la réduction du complément du problème consistant à décider si une DNF $\Phi = \bigvee_{i=1}^k T_i$ est une tautologie, ce qui est un problème coNP-complet bien connu. Soit X l'ensemble des variables de Φ et soit $x, y \notin X$. On définit un jeu G à deux joueurs par :
> * $\varphi_1 = \bigvee_{i=1}^k (T_i \wedge x \wedge \neg y) \vee (T_i \wedge \neg x \wedge y)$,
> * $\varphi_2 = (x \wedge y) \vee (\neg x \wedge \neg y)$,
> * $\pi_1 = \{y\}$,
> * $\pi_2 = X \cup \{x\}$.

G peut être construit en temps linéaire et φ_1, φ_2 sont en DNF.

On a $\varphi_1 \equiv \Phi \wedge (x \neq y)$ et $\varphi_2 \equiv (x = y)$. Grâce à la propriété 2.4 (page 42) et à son corollaire 2.1 (page 43), nous savons que G a un équilibre de Nash si et seulement si $((\Phi \wedge (x \neq y)) \vee \neg\Phi) \wedge (x = y)$ est satisfiable.
En effet, comme y n'apparaît pas dans Φ, nous avons

$$\neg\exists y : (\Phi \wedge x \neq y) \equiv \neg(\Phi \wedge \exists y : x \neq y) \equiv \neg(\Phi \wedge \top) \equiv \neg\Phi$$

et nous avons aussi

$$\neg\exists X \cup \{x\} : (x = y) \equiv \perp$$

Comme $\Phi \wedge (x \neq y) \wedge (x = y)$ n'est pas satisfiable, le jeu G a un équilibre de Nash si et seulement si $\neg\Phi \wedge (x = y)$ est satisfiable, c'est-à-dire si et seulement si $\neg\Phi$ est satisfiable, étant donné que x et y n'apparaissent pas dans Φ.
Donc, G a un équilibre de Nash si et seulement si Φ n'est pas une tautologie.

■

Lorsqu'on considère uniquement des jeux à deux joueurs, la complexité de ce problème peut descendre à P pour quelques classes non triviales de formules : si les buts des joueurs sont représentés par des formules en DNF renommables en Horn, affines, 2CNF ou CNF monotones.

On ne développe pas ces démonstrations ici, car la résolution de la projection en est le résultat principal, et que les détails sont en grande partie les mêmes que dans [Zanuttini, 2003, Section 6].

Nous pouvons également déterminer la complexité algorithmique du problème de la dominance faible dans un jeu booléen.

Propriété 2.21. *Décider si une stratégie s'_i donnée est faiblement dominée est Σ_2^p-complet. La difficulté reste valable même si φ_i est restreint aux DNF.*

Preuve : L'appartenance à Σ_2^p est immédiate. La difficulté est obtenue cette fois encore par la réduction du problème consistant à décider la validité de $\mathtt{QBF}_{2,\exists}$.
Soit $Q = \exists A \forall B \Phi$, et a, b deux nouvelles variables. On définit :

* $\varphi_1 = (a \wedge \Phi) \vee (\neg a \wedge b)$, $\pi_1 = A \cup \{a\}$,
* $\pi_2 = B \cup \{b\}$ (φ_2 est quelconque et n'est pas définie ici car sa valeur n'intervient pas dans la démonstration).

Soit M'_A une A-interprétation, et soit $s'_1 = (M'_A, \overline{a})$. On a $(\varphi_1)_{s'_1} \equiv (b)$.

Supposons que Q est valide, avec $M_A \in 2^A$ un modèle de A, et $s_1 = (M_A, a)$. s_1 est alors une stratégie gagnante pour 1, contrairement à s'_1. Donc s_1 domine faiblement s'_1.

Supposons à présent que Q n'est pas valide. Soit $M_A \in 2^A$, et $s_1 = (M_A, \overline{a})$. Alors, $(\varphi_1)_{s_1} \equiv (b) \equiv (\varphi_1)_{s'_1}$, donc, d'après la propriété 2.17 (page 54), s_1 ne domine pas faiblement s'_1.
À présent, soit $s_1 = (M_A, a)$. Comme Q n'est pas valide, il existe $M_B \in 2^B$ tel que $(M_A, M_B) \not\models \Phi$. Donc $(M_B, b) \models (\varphi_1)_{s'_1}$ mais $(M_B, b) \not\models (\varphi_1)_{s_1}$, et d'après la propriété 2.17 (page 54), s_1 ne domine pas faiblement s'_1.

Donc, s_1' est faiblement dominée si et seulement si Q est valide.
On peut remarquer que si Φ est en DNF, alors $\exists A, \forall B, \Phi$ est toujours Σ_2^p-complet et que φ_1 peut être transformée en DNF efficacement.

■

Le problème de l'existence d'un équilibre de Nash, et son calcul, dans un jeu donné est une question importante, qui a attiré un grand nombre de chercheurs (par exemple [Deng *et al.*, 2002; McKelvey et McLennan, 1996; Koller *et al.*, 1996]). La plupart de ces travaux ont consisté à étudier des problèmes concernant les équilibres de Nash en *stratégies mixtes*. Dans ce contexte, l'existence d'un tel équilibre est toujours garantie, mais le problème de savoir s'il est possible de le calculer en temps polynomial est toujours ouvert [Papadimitriou, 2001].

Les premiers résultats de complexité ont été donnés par [Gilboa et Zemel, 1989], qui ont notamment montré que le problème consistant à trouver plus d'un équilibre de Nash en stratégies mixtes dans un jeu en forme normale est NP-difficile, ou encore que prouver l'existence d'un équilibre de Nash en stratégies mixtes satisfiant certaines propriétés dans un jeu à deux joueurs en forme normale est NP-difficile. Une réduction plus simple pour ce problème a été trouvée dans [Conitzer et Sandholm, 2003]. Des extensions de ces résultats à des jeux plus généraux ont été trouvés par [Meggido et Papadimitriou, 1991], et par [Papadimitriou, 1994b].

D'autres travaux étudient des problèmes plus proches des nôtres, à savoir l'existence d'un équilibre de Nash en *stratégies pures*. Par exemple, les jeux étudiés dans [Gottlob *et al.*, 2005] sont des jeux stratégiques sur lesquels il est imposé des limitations quantitatives et/ou qualitatives sur l'influence d'un agent sur les décisions des autres agents : par exemple, restreindre le nombre de joueurs ayant une influence sur un joueur donné à k (ce qui correspondrait dans nos jeux booléens à $RP_i \leq k$) ; ou encore définir un graphe de dépendance, et étudier le cas particulier où il est acyclique. Nous étudierons plus longuement ces cas particuliers, et nous les comparerons avec les nôtres dans le chapitre 4 (page 121). Dans ce contexte, le problème consistant à décider s'il existe un équilibre de Nash en stratégies pures est NP-complet pour chacune des deux restrictions décrites plus haut.

Le problème de l'existence d'un équilibre de Nash en stratégies pures pour plusieurs classes plus spécifiques de jeux a également été étudié. Par exemple, trouver un équilibre de Nash en stratégies pures a été montré polynomial pour quatre notions de jeux symétriques ayant un nombre constant d'actions, mais non-polynomial si le nombre d'actions est linéaire dans le nombre de joueurs [Brandt *et al.*, 2007] (voir chapitre 4 (page 121)).

D'autre part, les jeux graphiques d'actions (action graph games, AGGs, voir chapitre 4 (page 121)) sont des jeux totalement expressifs (tout jeu peut être représenté par un AGG) représentés par des graphes dont les nœuds correspondent à des actions, et les arcs aux dépendances entre les utilités des agents effectuant ces actions [Bhat et Leyton-Brown, 2004; Jiang et Leyton-Brown, 2006]. Si le problème consistant à décider s'il existe un équilibre de Nash en stratégies pures dans un AGG est NP-complet [Jiang et Leyton-Brown, 2006], il existe des classes de jeu pour lesquels ce problème est polynomial [Jiang et Leyton-Brown, 2007]. C'est le cas pour les AGGs ayant un nombre d'actions, et donc de nœuds, limité à k.

Les jeux de congestion (congestion games, voir chapitre 4 (page 121)) [Rosenthal, 1973], sont des jeux dans lesquels les actions disponibles consistent en un ensemble de ressources, et l'utilité de chaque joueur dépend du nombre de joueurs ayant sélectionné les mêmes ressources (c'est-à-dire ayant joué la même action). Les jeux de congestion ont toujours un équilibre de Nash en stratégies pures [Rosenthal, 1973], et il est montré dans [Fabrikant *et al.*, 2004] qu'un PNE peut être calculé en temps polynomial dans le cas où le réseau de ce jeu (c'est-à-dire le graphe dont les nœuds sont les actions possibles, et les arcs les ressources) est symétrique.

2.6 Jeux booléens et duopole de Stackelberg

Von Stackelberg a imaginé en 1934 une situation à deux joueurs dans laquelle un des deux joueurs a une idée précise du comportement de son concurrent : il connaît parfaitement sa fonction de réaction et il l'intègre dans son processus de décision. On appelle alors ce joueur le *leader* ou le *meneur*. Suite à sa décision, son concurrent réagit en maximisant son profit et donc en suivant sa fonction de réaction ; il se contente de "suivre" le comportement du leader et pour cette raison, on l'appelle le *suiveur* (*follower*). Dans ce cas, le suiveur considère que ses décisions n'ont aucun impact sur le comportement du meneur (voir par exemple [Osborne et Rubinstein, 1994]).

Une idée est alors d'étudier la dynamicité dans les jeux booléens à deux joueurs en suivant ce principe de "meneur-suiveur". Cette ébauche de travail a été effectuée au laboratoire ILLC à Amsterdam avec Ulle Endriss.

Dans cette section, nous allons procéder par étapes : nous allons commencer par étudier les jeux de type Stackelberg à deux joueurs et une variable par joueur, puis nous chercherons empiriquement les propriétés que doivent respecter les stratégies du meneur et du receveur pour être optimales. Nous généraliserons ensuite ces résultats à des jeux à deux joueurs et 3 variables, puis ainsi de suite jusqu'à prouver ce résultat, trouvé empiriquement, pour des jeux à deux joueurs et n variables par joueur.

2.6.1 2 joueurs, 1 variable chacun

Dans cette section, nous allons étudier des jeux booléens $G = (N, V, \Gamma, \pi, \Phi)$ tels que $N = \{1, 2\}$, $V = \{a, b\}$, $\gamma_1 = \gamma_2 = \top$, $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, les buts des joueurs étant quelconques. Supposons que le joueur 1 est le meneur, et donc joue en premier. La liste de toutes les issues possibles de ce type de jeux une fois que le joueur 1 a joué la stratégie s_1 ($s_1 = a$ ou $s_1 = \overline{a}$) est représentée sur la figure 2.7 (page suivante).

Chaque paire de couples du type $(x, y)(z, t)$ apparaissant dans la première colonne de ce tableau représente les issues du jeu pour le joueur 1 s'il joue s_1 selon la stratégie choisie par le joueur 2. Donc si le joueur 1 joue s_1 et que le joueur 2 joue s_2 ($s_2 = b$ ou $s_2 = \overline{b}$), l'issue du jeu sera[13] (x, y) ; si le joueur 2 joue $\overline{s}_2$ (si $s_2 = b$, alors $\overline{s}_2 = \overline{b}$ et vice-versa), l'issue du jeu sera (z, t). Dans ces couples, la notation _ signifie "0 ou 1".

La seconde colonne du tableau représente l'intérêt de s_1 pour le joueur 1 : c'est soit une stratégie gagnante (si l'issue est $(1, \_)(1, \_)$ par exemple, quel que soit le choix de 2, 1 est sûr de gagner s'il joue s_1), soit une stratégie perdante (si l'issue est $(0, \_)(0, \_)$ par exemple, 1 est sûr de perdre s'il joue s_1), soit le joueur 1 ne peut pas prévoir quelle sera son utilité (si l'issue est $(0, 1)(1, 1)$ par exemple, 1 ne sait pas quelle stratégie jouera 2).

[13]Comme précédemment, cela signifie que le joueur 1 aura l'utilité x, et le joueur 2 l'utilité y.

$(1,\_)(1,\_)$	stratégie gagnante	$s_1 \models \varphi_1$
$(0,0)(1,1)$	stratégie gagnante	$s_1 \models \varphi_1 \leftrightarrow \varphi_2$
$(1,1)(0,0)$	stratégie gagnante	$s_1 \models \varphi_1 \leftrightarrow \varphi_2$
$(0,\_)(0,\_)$	stratégie perdante	$s_1 \models \neg\varphi_1$
$(1,0)(0,1)$	stratégie perdante	$s_1 \models \neg(\varphi_1 \leftrightarrow \varphi_2)$
$(0,1)(1,0)$	stratégie perdante	$s_1 \models \neg(\varphi_1 \leftrightarrow \varphi_2)$
$(1,0)(0,0)$	imprévisible	$s_1 \models \neg\varphi_2$
$(0,0)(1,0)$	imprévisible	$s_1 \models \neg\varphi_2$
$(0,1)(1,1)$	imprévisible	$s_1 \models \varphi_2$
$(1,1)(0,1)$	imprévisible	$s_1 \models \varphi_2$

***Figure 2.7* —** Issues possibles du jeu de type Stackelberg à 2 joueurs et 1 variable chacun

Enfin, dans la troisième colonne se trouve une description des effets de s_1 sur les buts des joueurs (par exemple, si l'issue est $(1,\_)(1,\_)$, on sait que $s_1 \models \varphi_1$).

A partir de ce tableau, qui couvre tous les cas possibles, on déduit que la stratégie s_1 est optimale pour le joueur 1 si et seulement si

* soit c'est une stratégie gagnante,
* soit[14] $\overline{s}_1$ est une stratégie perdante (auquel cas s_1 est forcément meilleure),
* soit il est impossible de choisir entre s_1 et $\overline{s}_1$, les deux étant imprévisibles (dans ce cas, les deux stratégies sont optimales).

Une fois que le joueur 1 a joué, le joueur 2 peut en toute connaissance de cause jouer à son tour, et il choisira une stratégie qui ne le satisfait pas uniquement s'il ne peut pas être satisfait.

On obtient donc l'assertion suivante :

Assertion 2.1. *Soit un jeu booléen $G = (N,V,\Gamma,\pi,\Phi)$ tel que $N = \{1,2\}$, $V = \{a,b\}$, $\gamma_1 = \gamma_2 = \top$, $\pi_1 = \{a\}$, $\pi_2 = \{b\}$ du type Stackelberg.*

* **La stratégie s_1 est optimale pour le meneur** *de G si et seulement si :*

$$\exists s_2 \; tq \; s_1 \models \varphi_1 \vee ((\varphi_1 \leftrightarrow \varphi_2) \wedge (\varphi_1)_{s_2}) \qquad \textit{(stratégie gagnante)}$$

ou

$$\exists s_2 \; tq \; \overline{s}_1 \models \neg\varphi_1 \vee (\neg(\varphi_1 \leftrightarrow \varphi_2) \wedge (\varphi_2)_{s_2}) \qquad \textit{(stratégie non perdante)}$$

ou

$$\begin{pmatrix} s_1 \models \varphi_2 \\ ou \\ s_1 \models \neg\varphi_2 \end{pmatrix} \; et \; \begin{pmatrix} \overline{s}_1 \models \varphi_2 \\ ou \\ \overline{s}_1 \models \neg\varphi_2 \end{pmatrix} \qquad \textit{(peu importe)}$$

* *Si le joueur 1 joue s_1,* **la stratégie s_2 est optimale pour le suiveur** *si et seulement si*

$$s_2 \models \neg(\varphi_2)_{s_1} \Rightarrow \overline{s}_2 \models \neg(\varphi_2)_{s_1}$$

[14] Si $s_1 = a$, alors $\overline{s}_1 = \overline{a}$, et vice-versa.

2.6.2 2 joueurs, 3 variables

On étudie à présent des jeux booléens $G = (N, V, \Gamma, \pi, \Phi)$ tels que $N = \{1,2\}$, $V = \{a,b,c\}$, $\gamma_1 = \gamma_2 = \top$, $\pi_1 = \{a,c\}$, $\pi_2 = \{b\}$. Supposons que le joueur 1 est le meneur, et donc qu'il joue en premier. Le joueur 2 répond en instanciant b, et c'est ensuite encore au joueur 1 de jouer.

Le joueur 2 se retrouve donc ici dans la même position que le joueur 1 dans la section précédente : il est meneur d'un jeu booléen de type Stackelberg à deux joueurs ayant une variable chacune. Sa stratégie optimale est donc donnée par l'assertion 2.1 (page précédente). Il nous reste donc à définir la stratégie optimale du premier joueur pour l'instanciation de sa première variable.

En effectuant le même raisonnement que précédemment, nous trouvons donc l'assertion suivante :

Assertion 2.2. *Une fois la première variable du meneur instanciée, une stratégie optimale pour le joueur 2 est donnée dans l'assertion 2.1 (page précédente), ce joueur se retrouvant meneur du sous-jeu à deux joueurs et deux variables restant. Soit s_{1_1} l'instanciation de la première variable du joueur 1 dans la stratégie s_1 (puisque s_{1_1}, s_{1_2} et s_2 correspondent chacun à l'instanciation d'une seule variable, on peut écrire comme précédemment $\overline{s}_{1_1}$, $\overline{s}_{1_2}$ et $\overline{s}_2$).*

* **La stratégie s_2 est optimale pour le suiveur** *si et seulement si*

$$\exists s_{1_2} \; tq \; s_2 \models (\varphi_2)_{s_{1_1}} \vee (((\varphi_1)_{s_{1_1}} \leftrightarrow (\varphi_2)_{s_{1_1}}) \wedge (\varphi_2)_{s_{1_1}s_{1_2}}) \qquad \textit{(stratégie gagnante)}$$

ou

$$\exists s_{1_2} \; tq \; \overline{s}_2 \models \neg(\varphi_2)_{s_{1_1}} \vee (\neg((\varphi_1)_{s_{1_1}} \leftrightarrow (\varphi_2)_{s_{1_1}}) \wedge (\varphi_1)_{s_{1_1}s_{1_2}}) \qquad \textit{(stratégie non perdante)}$$

ou

$$\begin{pmatrix} s_2 \models (\varphi_1)_{s_{1_1}} \\ ou \\ s_2 \models \neg(\varphi_1)_{s_{1_1}} \end{pmatrix} \; et \; \begin{pmatrix} \overline{s}_2 \models (\varphi_1)_{s_{1_1}} \\ ou \\ \overline{s}_2 \models \neg(\varphi_1)_{s_{1_1}} \end{pmatrix} \qquad \textit{(peu importe)}$$

* *Le joueur 1 sait que 2 choisira une stratégie s_2 optimale suivant l'instanciation de sa première variable, c'est-à-dire s_{1_1}. Il peut donc choisir sa stratégie suivant s_2.* **La stratégie s_1 est optimale pour le meneur** *si et seulement si*

$$\forall s_2 \; optimale \; pour \; 2 \; : \begin{pmatrix} \exists s_{1_2} \; tq \; (s_{1_1}, s_{1_2}) \models (\varphi_1)_{s_2} & \textit{(stratégie gagnante)} \\ ou & \\ \overline{s}_{1_1} \models \neg(\varphi_1)_{s_2} & \textit{(stratégie non perdante)} \end{pmatrix}$$

ou

$$si \; \forall s_{1_2} \; optimale \; pour \; 1, \; 1 \; est \; indifférent \; : \begin{pmatrix} \exists s_{1_2} \; tq \; s_{2_1} \models \neg(\varphi_2)_{s_{1_2}} \\ et \\ \exists s_{1_2} \; tq \; \overline{s}_{2_1} \models \neg(\varphi_2)_{s_{1_2}} \end{pmatrix} \qquad \textit{(peu importe)}$$

Exemple 2.14. *Soit $G = (N, V, \Gamma, \pi, \Phi)$ un jeu booléen tel que $N = \{1,2\}$, $V = \{a,b,c\}$, $\gamma_1 = \gamma_2 = \top$, $\pi_1 = \{a,c\}$, $\pi_2 = \{b\}$, $\varphi_1 = (a \wedge b) \vee (c \leftrightarrow b)$ et $\varphi_2 = (\neg a \wedge \neg b) \vee c$.*

Le joueur 1 commence par jouer la variable a, et doit tout d'abord calculer les stratégies optimales du joueur 2 pour chacune des instanciations de a.

Soit $s_{1_1} = a$. On a alors $(\varphi_2)_{s_{1_1}} = c$, $(\varphi_1)_{s_{1_1}} = b \vee (c \leftrightarrow b)$.

On a $\neg b \models \neg((\varphi_1)_{s_{1_1}} \leftrightarrow (\varphi_2)_{s_{1_1}})$*, ce qui est une stratégie perdante pour* 2*, et* $b \models (\varphi_1)_{s_{1_1}}$ *ce qui est une stratégie non perdante pour* 2*, et donc sa stratégie optimale.*

Une fois que 1 *a fait ce calcul, il n'a plus besoin de calculer la stratégie optimale de* 2 *pour* $\bar{s}_{1_1} = \neg a$ *car il sait déjà qu'il a une stratégie gagnante s'il choisit a :* 2 *étant rationnel va choisir b, et alors* $ab \models \varphi_1$.

2.6.3 2 joueurs, 2 variables chacun

Nous étudions à présent des jeux booléens $G = (N, V, \Gamma, \pi, \Phi)$ tels que $N = \{1,2\}$, $V = \{a,b,c,d\}$, $\gamma_1 = \gamma_2 = \top$, $\pi_1 = \{a,c\}$, $\pi_2 = \{b,d\}$. Supposons que le joueur 1 commence par instancier la variable a, puis que 2 instancie b, et ainsi de suite.

Nous connaissons déjà la stratégie optimale de 2 une fois que 1 a instancié a, ainsi que la stratégie optimale de 1 pour la variable c une fois que a et b sont instanciées. Il nous manque donc à connaître l'instanciation optimale de a pour le premier joueur.

s_{1_1} est une stratégie optimale de 1 si et seulement si $\forall s_{2_1}$ optimales pour[15] 2 :

1. $\exists s_{1_2}$ une stratégie gagnante pour 1, donc telle que :

$$\exists s_{2_2}\ tq\ (s_{1_1}, s_{1_2}) \models (\varphi_1)_{s_{2_1}} \vee (((\varphi_1)_{s_{2_1}} \leftrightarrow (\varphi_2)_{s_{2_1}}) \wedge (\varphi_1)_{s_{2_1}s_{2_2}})$$

2. ou, $\bar{s}_{1_1}$ est une stratégie perdante :

$$\exists s_{2_2}\ tq\ \bar{s}_{1_1} \models \neg(\varphi_1)_{s_{2_1}} \vee (\neg((\varphi_1)_{s_{2_1}} \leftrightarrow (\varphi_2)_{s_{2_1}}) \wedge (\varphi_2)_{s_{2_1}s_{2_2}})$$

3. ou, 1 est indifférent :

$$\begin{pmatrix} (s_{1_1}, s_{1_2}) \models (\varphi_2)_{s_{2_1}} \\ ou \\ (s_{1_1}, s_{1_2}) \models \neg(\varphi_2)_{s_{2_1}} \end{pmatrix} \text{ et } \begin{pmatrix} (s_{1_1}, \bar{s}_{1_2}) \models (\varphi_2)_{s_{2_1}} \\ ou \\ (s_{1_1}, \bar{s}_{1_2}) \models \neg(\varphi_2)_{s_{2_1}} \end{pmatrix}$$

$$\text{et } \begin{pmatrix} (\bar{s}_{1_1}, s_{1_2}) \models (\varphi_2)_{s_{2_1}} \\ ou \\ (\bar{s}_{1_1}, s_{1_2}) \models \neg(\varphi_2)_{s_{2_1}} \end{pmatrix} \text{ et } \begin{pmatrix} (\bar{s}_{1_1}, \bar{s}_{1_2}) \models (\varphi_2)_{s_{2_1}} \\ ou \\ (\bar{s}_{1_1}, \bar{s}_{1_2}) \models \neg(\varphi_2)_{s_{2_1}} \end{pmatrix}$$

Assertion 2.3.

∗ **La stratégie s_{2_1} est optimale pour le suiveur** *si et seulement si*

$$\forall s_{1_2} \textit{ optimale pour } 1 : \begin{pmatrix} \exists s_{2_2}\ tq\ (s_{2_1}, s_{2_2}) \models (\varphi_2)_{s_{1_1}, s_{1_2}} & \textit{(stratégie gagnante)} \\ ou & \\ \bar{s}_{2_1} \models \neg(\varphi_2)_{s_{1_1}, s_{1_2}} & \textit{(stratégie non perdante)} \end{pmatrix}$$

ou

$$\textit{si } \forall s_{1_2} \textit{ optimale, } 1 \textit{ est indifférent} : \begin{pmatrix} \exists s_{1_2}\ tq\ s_{2_1} \models \neg(\varphi_2)_{s_{1_2}} \\ et \\ \exists s_{1_2}\ tq\ \bar{s}_{2_1} \models \neg(\varphi_2)_{s_{1_2}} \end{pmatrix} \quad \textit{(peu importe)}$$

[15] De nouveau, quels que soient $i \in \{1,2\}$ et $j \in \{1,2\}$, chaque s_{i_j} correspondant à l'instanciation d'une seule variable, il est possible d'écrire $\bar{s}_{i_j}$.

* **La stratégie s_{1_1} est optimale pour le meneur** *si et seulement si pour toute stratégie s_{2_1} optimale pour 2, soit il existe s_{1_2} optimale pour 1, soit $\bar{s}_{1_1}$ est une stratégie perdante, soit 1 est indifférent. Donc s_{1_1} est une stratégie optimale pour 1 si et seulement si $\forall s_{2_1}$ optimale pour 2 :*

$$\begin{array}{cl}
\exists s_{1_2}, s_{2_2} \textit{ tq } (s_{1_1}, s_{1_2}) \models ((\varphi_1)_{s_{2_1}} \vee ((\varphi_1)_{s_{2_1}} \leftrightarrow (\varphi_2)_{s_{2_1}}) \wedge (\varphi_1)_{s_{2_1}s_{2_2}}) & \textit{(stratégie gagnante)} \\
\textit{ou} & \\
\exists s_{2_2} \textit{ tq } \bar{s}_{1_1} \models \neg(\varphi_1)_{s_{2_1}} \vee (\neg((\varphi_1)_{s_{2_1}} \leftrightarrow (\varphi_2)_{s_{2_1}}) \wedge (\varphi_2)_{s_{2_1}s_{2_2}}) & \textit{(stratégie non perdante)} \\
\textit{ou} & \\
\left(\begin{array}{c} (s_{1_1}, s_{1_2}) \models (\varphi_2)_{s_{2_1}} \\ \textit{ou} \\ (s_{1_1}, s_{1_2}) \models \neg(\varphi_2)_{s_{2_1}} \end{array}\right) \textit{ et } \left(\begin{array}{c} (s_{1_1}, \bar{s}_{1_2}) \models (\varphi_2)_{s_{2_1}} \\ \textit{ou} \\ (s_{1_1}, \bar{s}_{1_2}) \models \neg(\varphi_2)_{s_{2_1}} \end{array}\right) & \\
\textit{et} & \textit{(peu importe)} \\
\left(\begin{array}{c} (\bar{s}_{1_1}, s_{1_2}) \models (\varphi_2)_{s_{2_1}} \\ \textit{ou} \\ (\bar{s}_{1_1}, s_{1_2}) \models \neg(\varphi_2)_{s_{2_1}} \end{array}\right) \textit{ et } \left(\begin{array}{c} (\bar{s}_{1_1}, \bar{s}_{1_2}) \models (\varphi_2)_{s_{2_1}} \\ \textit{ou} \\ (\bar{s}_{1_1}, \bar{s}_{1_2}) \models \neg(\varphi_2)_{s_{2_1}} \end{array}\right) &
\end{array}$$

2.6.4 2 joueurs, *n* variables chacun

Etudions à présent des jeux booléens $G = (N, V, \Gamma, \pi, \Phi)$ tels que $N = \{1, 2\}$, $V = \{v_1, \dots v_{2n}\}$, $\gamma_1 = \gamma_2 = \top$, $\pi_1 = \{v_1, \dots, v_n\}$, $\pi_2 = \{v_{n+1}, \dots v_{2n}\}$. Chaque joueur contrôle donc exactement n variables. Supposons que le joueur 1 est le meneur, et qu'il commence par jouer la variable v_1. Le joueur 2 joue ensuite la variable v_{n+1}, et ainsi de suite.

Propriété 2.22.

* **La stratégie s_{1_1} est optimale pour le meneur** *si et seulement si $\forall s_{2_1}, \dots, s_{2_{n-1}}$ optimales*[16] *pour 2 :*

$$\begin{array}{ll}
& \exists s_{1_2}, \dots, s_{1_n}, s_{2_n} \textit{ tq} \\
& (s_{1_1}, \dots, s_{1_n}) \models (\varphi_1)_{s_{2_1} \dots s_{2_{n-1}}} \vee (((\varphi_1)_{s_{2_1} \dots s_{2_{n-1}}} \leftrightarrow (\varphi_2)_{s_{2_1} \dots s_{2_{n-1}}}) \wedge (\varphi_1)_{s_{2_1} \dots s_{2_n}}) \qquad (1) \\
\textit{ou} & \\
& \exists s_{2_n} \textit{ tq } \bar{s}_{1_1} \models \neg(\varphi_1)_{s_{2_1} \dots s_{2_{n-1}}} \vee (\neg((\varphi_1)_{s_{2_1} \dots s_{2_{n-1}}} \leftrightarrow (\varphi_2)_{s_{2_1} \dots s_{2_{n-1}}}) \wedge (\varphi_2)_{s_{2_1} \dots s_{2_n}}) \qquad (2) \\
\textit{ou} & \\
& \forall s_1 \in S_1, \left(\begin{array}{c} s_1 \models (\varphi_2)_{s_{2_1} \dots s_{2_{n-1}}} \\ \textit{ou} \\ s_1 \models \neg(\varphi_2)_{s_{2_1} \dots s_{2_{n-1}}} \end{array}\right) \qquad (3)
\end{array}$$

* **La stratégie s_{2_1} est optimale pour le suiveur** *si et seulement si*

$$\begin{array}{l}
\forall s_{1_2}, \dots, s_{1_n} \textit{ optimales pour } 1 \ : \left(\begin{array}{cc} \exists s_{2_2}, \dots, s_{2_n} \textit{ tq } (s_{2_1}, \dots, s_{2_n}) \models (\varphi_2)_{s_{1_1}, \dots, s_{1_n}} & (1) \\ \textit{ou} & \\ \bar{s}_{2_1} \models \neg(\varphi_2)_{s_{1_1}, \dots, s_{1_n}} & (2) \end{array}\right) \\
\textit{ou} \\
\textit{si } \forall s_{1_2}, \dots, s_{1_n} \textit{ optimales, } 1 \textit{ est indifférent } \ : \left(\begin{array}{c} \exists s_1 \in S_1 \textit{ tq } s_{2_1} \models \neg(\varphi_2)_{s_1} \\ \textit{et} \\ \exists s_1 \in S_1 \textit{ tq } \bar{s}_{2_1} \models \neg(\varphi_2)_{s_1} \end{array}\right) \quad (3)
\end{array}$$

[16] De nouveau, quels que soient $i \in \{1, 2\}$ et $j \in \{1, \dots, n\}$, chaque s_{i_j} correspondant à l'instanciation d'une seule variable, il est possible d'écrire $\bar{s}_{i_j}$.

Preuve :

∗ Joueur 1.

⇒ L'optimalité n'est pas exclusive (s_{1_1} et $\overline{s}_{1_1}$ peuvent être optimales). Nous voulons donc montrer que si s_{1_1} ne respecte pas les équations (1), (2) et (3), alors $\overline{s}_{1_1}$ est optimale.

(1) $\forall s_{1_2},\ldots,s_{1_n},(s_{1_1},s_{1_2},\ldots,s_{1_n}) \not\models (\varphi_1)_{s_{2_1}\ldots s_{2_{n-1}}} \vee (((\varphi_1)_{s_{2_1}\ldots s_{2_{n-1}}} \leftrightarrow (\varphi_2)_{s_{2_1}\ldots s_{2_{n-1}}}) \wedge (\exists s_{2_n}\ tq\ (\varphi_1)_{s_{2_1}\ldots s_{2_n}}))$.
Les possibilités de satisfaire cette formule dépendent de la valeur de s_{2_n}, qui est la seule variable dont on ne connaît pas encore l'instanciation. On obtient les possibilités suivantes :$\forall s_{1_2},\ldots,s_{1_n}$ et $\forall s_{2_1},\ldots,s_{2_{n-1}}$ stratégies optimales pour 2, après un chemin contenant $s_{1_1},s_{1_2},\ldots,s_{1_n},s_{2_1},\ldots,s_{2_{n-1}}$:

2

s_{2_n}	$\overline{s}_{2_n}$	
$(1,\_)$	$(0,1)$	$[(1,1) \rightarrow$ 1 est indifférent ; $(1,0) \rightarrow$ 1 perd]
$(0,1)$	$(1,\_)$	$[(1,1) \rightarrow$ 1 est indifférent ; $(1,0) \rightarrow$ 1 perd]
$(0,\_)$	$(0,\_)$	[1 perd]

En effet, si (1) n'est pas satisfaite, on ne peut pas avoir, une fois $s_{1_1},s_{1_2},\ldots,s_{1_n},s_{2_1},\ldots,s_{2_{n-1}}$ instanciées, φ_1 satisfaite pour tout s_{2_n}. On ne peut pas avoir non plus pour tout s_{2_n}, $\varphi_1 \leftrightarrow \varphi_2$ et pour un s_{2_n}, φ_1 satisfaite. Les seules options disponibles sont donc celles décrites plus haut.

(2) $\overline{s}_{1_1} \not\models \neg(\varphi_1)_{s_{2_1}\ldots s_{2_{n-1}}} \vee (\neg((\varphi_1)_{s_{2_1}\ldots s_{2_{n-1}}} \leftrightarrow (\varphi_2)_{s_{2_1}\ldots s_{2_{n-1}}}) \wedge (\exists s_{2_n}\ tq\ (\varphi_2)_{s_{2_1}\ldots s_{2_n}}))$.
Les possibilités de satisfaire cette formule dépendent une fois encore de la valeur de s_{2_n}. Nous obtenons les possibilités suivantes : $\forall s_{2_1},\ldots,s_{2_{n-1}}$ stratégies optimales pour 2, $\exists s_{1_2},\ldots,s_{1_n}$ telles que le chemin contenant $\overline{s}_{1_1},s_{1_2},\ldots,s_{1_n},s_{2_1},\ldots,s_{2_{n-1}}$ satisfait les conditions suivantes pour s_{2_n} :

2

s_{2_n}	$\overline{s}_{2_n}$	
$(0,\_)$	$(1,1)$	$[(0,1) \rightarrow$ 1 est indifférent ; $(0,0) \rightarrow$ 1 gagne]
$(1,1)$	$(0,\_)$	$[(0,1) \rightarrow$ 1 est indifférent ; $(0,0) \rightarrow$ 1 gagne]
$(1,0)$	$(0,0)$	[1 est indifférent]
$(0,0)$	$(1,0)$	[1 est indifférent]
$(1,\_)$	$(1,\_)$	[1 gagne]

(3) $\exists s_1 \in S_1, \begin{pmatrix} s_1 \not\models (\varphi_2)_{s_{2_1}\ldots s_{2_{n-1}}} \\ et \\ s_1 \not\models \neg(\varphi_2)_{s_{2_1}\ldots s_{2_{n-1}}} \end{pmatrix}$

Une fois encore, les possibilités de satisfaire cette formule dépendent de la valeur de s_{2_n}. Nous obtenons les possibilités suivantes : $\forall s_{2_1}, \ldots, s_{2_{n-1}}$ stratégies optimales pour 2, $\exists s_{1_1}, s_{1_2}, \ldots, s_{1_n}$ telles que le chemin contenant $\bar{s}_{1_1}, s_{1_2}, \ldots, s_{1_n}, s_{2_1}, \ldots, s_{2_{n-1}}$ satisfait les conditions suivantes pour s_{2_n} :

$$\begin{array}{lll} & 2 & \\ s_{2_n} \swarrow & & \searrow \bar{s}_{2_n} \\ (\_,0) & & (\_,1)\ [(\_,0)(0,1) \rightarrow 1 \text{ perd} ; (\_,0)(1,1) \rightarrow 1 \text{ gagne}] \\ (\_,1) & & (\_,0)\ [(0,1)(\_,0) \rightarrow 1 \text{ perd} ; (1,1)(\_,0) \rightarrow 1 \text{ gagne}] \end{array}$$

Plusieurs cas sont à envisager :

* **À partir de (1)** : Si $\forall s_{1_2}, \ldots, s_{1_n}$ et $\forall s_{2_1}, \ldots, s_{2_{n-1}}$, le résultat obtenu par le joueur 1 est $[(0,1)$ et $(1,0)]$ ou $[(1,0)$ et $(0,1)]$ ou $[(0,\_)$ et $(0,\_)]$, alors s_{1_1} est une stratégie perdante, et $\bar{s}_{1_1}$ est une stratégie optimale.

  Si $\forall s_{1_2}, \ldots, s_{1_n}$ et $\forall s_{2_1}, \ldots, s_{2_{n-1}}$, le résultat obtenu par le joueur 1 est $[(1,1)$ et $(0,1)]$ ou $[(0,1)$ et $(1,1)]$, on sait d'après (2) que le joueur 1 peut obtenir les mêmes résultats en jouant $\bar{s}_{1_1}$, auquel cas il est indifférent entre ces deux stratégies (et $\bar{s}_{1_1}$ est une stratégie optimale) ; ou alors la stratégie $\bar{s}_{1_1}$ est une stratégie gagnante.
* **À partir de (2)** : Si $\exists s_{1_2}, \ldots, s_{1_n}$ et $\forall s_{2_1}, \ldots, s_{2_{n-1}}$, le résultat obtenu par le joueur 1 est $[(0,0)$ et $(1,1)]$ ou $[(1,1)$ et $(0,0)]$ ou $[(1,\_)$ et $(1,\_)]$, alors $\bar{s}_{1_1}$ est une stratégie gagnante, donc est une stratégie optimale.

  D'après (1), on sait que soit s_{1_1} est une stratégie perdante, soit le joueur 1 est indifférent. Donc si $\exists s_{1_2}, \ldots, s_{1_n}$ et $\forall s_{2_1}, \ldots, s_{2_{n-1}}$ tels que le résultat obtenu par le joueur 1 est $[(0,1)$ et $(1,1)]$ ou $[(1,1)$ et $(0,1)]$ ou $[(1,0)$ et $(0,0)]$, ou $[(0,0)$ et $(1,0)]$, $\bar{s}_{1_1}$ est une stratégie optimale (puisque soit s_{1_1} est une stratégie perdante, soit le joueur 1 est indifférent entre s_{1_1} et $\bar{s}_{1_1}$).
* **À partir de (1), (2) et (3)** : Si la stratégie s_1 obtenue par (3) entraîne un résultat pour le joueur 1 de la forme $[(\_,0)$ et $(1,1)]$ ou $[(1,1)$ et $(\_,0)]$, alors d'après (1) et (2), nous savons que s_1 contient $\bar{s}_{1_1}$, et que s_1 est une stratégie gagnante, donc $\bar{s}_{1_1}$ est optimale.
* **À partir de (1), (2) et (3)** : Si la stratégie s_1 obtenue par (3) entraîne un résultat pour le joueur 1 de la forme $[(\_,0)$ et $(0,1)]$ ou $[(0,1)$ et $(\_,0)]$, alors d'après (1) et (2), nous savons que s_1 contient s_{1_1}. Dans ce cas, soit s_{1_1} est une stratégie perdante pour tout s_{1_i}, $i \in [1,n]$ et $\bar{s}_{1_1}$ est optimale ; soit le joueur 1 est indifférent entre s_{1_1} et $\bar{s}_{1_1}$, auquel cas $\bar{s}_{1_1}$ est optimale.

$\Leftarrow$ Supposons à présent que s_{1_1} satisfait les équations (1), (2) et (3), et prouvons alors que s_{1_1} est optimale.

(1) $\forall s_{2_1},\ldots,s_{2_{n-1}}$ optimales pour 2, $\exists s_{1_2},\ldots,s_{1_n},s_{2_n}$ telles que

$$(s_{1_1},\ldots,s_{1_n}) \models \underbrace{(\varphi_1)_{s_{2_1}\ldots s_{2_{n-1}}}}_{\varphi_i \text{ est satisfaite } \forall s_{2_n}} \vee$$

$$\underbrace{(((\varphi_1)_{s_{2_1}\ldots s_{2_{n-1}}} \leftrightarrow (\varphi_2)_{s_{2_1}\ldots s_{2_{n-1}}}) \wedge (\varphi_1)_{s_{2_1}\ldots s_{2_n}})}_{(*)}$$

$(*)$: 2 jouant de façon rationnelle, ce joueur choisira s_{2_n} telle que $s_{2_n} \models (\varphi_2)_{s_{1_1},\ldots,s_{1_n},s_{2_1}\ldots s_{2_{n-1}}}$, et alors $s_{2_n} \models (\varphi_1)_{s_{1_1},\ldots,s_{1_n},s_{2_1}\ldots s_{2_{n-1}}}$.
s_{1_1} est donc optimale.

(2) $\forall s_{2_1},\ldots,s_{2_{n-1}}$ optimales pour 2, $\exists s_{2_n}$ telle que $\forall s_{1_2},\ldots,s_{1_n},s_{2_n}$,

$$\bar{s}_{1_1} \models \underbrace{\neg(\varphi_1)_{s_{2_1}\ldots s_{2_{n-1}}}}_{\varphi_i \text{ n'est pas satisfaite } \forall s_{2_n}} \vee$$

$$\underbrace{(\neg((\varphi_1)_{s_{2_1}\ldots s_{2_{n-1}}} \leftrightarrow (\varphi_2)_{s_{2_1}\ldots s_{2_{n-1}}}) \wedge (\varphi_2)_{s_{2_1}\ldots s_{2_n}})}_{(*)}$$

$(*)$: 2 jouant de façon rationnelle, ce joueur choisira s_{2_n} telle que $s_{2_n} \models (\varphi_2)_{s_{1_1},\ldots,s_{1_n},s_{2_1}\ldots s_{2_{n-1}}}$, et alors $s_{2_n} \models \neg(\varphi_1)_{s_{1_1},\ldots,s_{1_n},s_{2_1}\ldots s_{2_{n-1}}}$.
s_{1_1} est donc optimale.

(3) $\forall s_{2_1},\ldots,s_{2_{n-1}}$ optimales pour 2, $\forall s_1 \in S_1$,

$$\left(\begin{array}{c} s_1 \models (\varphi_2)_{s_{2_1}\ldots s_{2_{n-1}}} \\ ou \\ s_1 \models \neg(\varphi_2)_{s_{2_1}\ldots s_{2_{n-1}}} \end{array} \right)$$

$\forall s_1 \in S_1$, 1 ne connaît donc pas le choix de 2, et donc ne peut pas choisir. s_{1_1} et $\bar{s}_{1_1}$ sont donc optimales.

∗ Joueur 2.

⇒ L'optimalité n'est pas exclusive (s_{2_1} et $\bar{s}_{2_1}$ peuvent être optimales). Nous voulons donc montrer que si s_{2_1} ne respecte pas les équations (1), (2) et (3), alors $\bar{s}_{2_1}$ est optimale.

(1) $\exists s_{1_2},\ldots,s_{1_n}$ optimales pour 1 telles que $\forall s_{2_2},\ldots,s_{2_n}$ telles que $(s_{2_1},\ldots,s_{2_n}) \not\models (\varphi_2)_{s_{1_1},\ldots,s_{1_n}}$, donc $(s_{2_1},\ldots,s_{2_n}) \models \neg(\varphi_2)_{s_{1_1},\ldots,s_{1_n}}$, donc s_{2_1} est une stratégie perdante, et donc $\bar{s}_{2_1}$ est optimale.

(2) $\exists s_{1_2},\ldots,s_{1_n}$ optimales pour 1 telles que $\bar{s}_{2_1} \not\models \neg(\varphi_2)_{s_{1_1},\ldots,s_{1_n}}$. Donc, $\exists s_{2_2},\ldots,s_{2_n}$ telles que $\bar{s}_{2_1},\ldots,s_{2_n} \models (\varphi_2)_{s_{1_1},\ldots,s_{1_n}}$. Donc il existe une stratégie gagnante contenant $\bar{s}_{2_1}$, et donc cette stratégie est optimale.

(3) si $\forall s_{1_2},\ldots,s_{1_n}$ optimales pour 1, 1 est indifférent :

$$\begin{pmatrix} \forall s_1 \in S_1, s_{2_1} \not\models \neg(\varphi_2)_{s_1} \\ ou \\ \forall s_1 \in S_1, \bar{s}_{2_1} \not\models \neg(\varphi_2)_{s_1} \end{pmatrix}.$$

Donc, $\begin{pmatrix} \forall s_1 \in S_1, \exists s_{2_2}, \ldots, s_{2_n} \text{ telles que } s_{2_1} \models (\varphi_2)_{s_1} \quad (4) \\ or \\ \forall s_1 \in S_1, \exists s_{2_2}, \ldots, s_{2_n} \text{ telles que } \bar{s}_{2_1} \models (\varphi_2)_{s_1} \quad (5) \end{pmatrix}.$

Nous savons d'après (1) que (4) n'est pas possible. (5) est donc vrai, et $\bar{s}_{2_1}$ est optimale.

$\Leftarrow$ Supposons à présent que s_{2_1} satisfait les équations (1), (2) et (3), et prouvons que s_{2_1} est optimale.

(1) $\forall s_{1_2}, \ldots, s_{1_n}$ optimales pour 1 : $\exists s_{2_2}, \ldots, s_{2_n}$ telles que $(s_{2_1}, \ldots, s_{2_n}) \models (\varphi_2)_{s_{1_1}, \ldots, s_{1_n}}$. φ_2 est satisfaite, et donc s_{2_1} est optimale.

(2) $\forall s_{1_2}, \ldots, s_{1_n}$ optimales pour 1 : $\bar{s}_{2_1} \models \neg(\varphi_2)_{s_{1_1}, \ldots, s_{1_n}}$. φ_2 n'est pas satisfaite, et donc s_{2_1} est optimale.

(3) si $\forall s_{1_2}, \ldots, s_{1_n}$ optimales pour 1, 1 est indifférent : $\begin{pmatrix} \exists s_1 \in S_1 \ st \ s_{2_1} \models \neg(\varphi_2)_{s_1} \\ et \\ \exists s_1 \in S_1 \ st \ \bar{s}_{2_1} \models \neg(\varphi_2)_{s_1} \end{pmatrix}.$

Dans ce cas, 2 ne peut pas prévoir les choix de 1, et peut donc toujours avoir une stratégie perdante. Ce joueur est donc indifférent entre ses deux choix, et s_{1_1} est optimale.

■

3 Jeux booléens et préférences non dichotomiques

Ce choix d'utilités binaires (pour lequel les agents peuvent seulement exprimer leur totale satisfaction ou leur total mécontentement, sans niveaux intermédiaires) est une vraie perte de généralité. En effet, on ne peut par exemple pas exprimer dans un jeu booléen classique des préférences du type "Je préfèrerais boire un café, si je ne peux pas un thé, et si ce n'est pas possible non plus je prendrai un chocolat chaud", qui nécessite 3 niveaux d'utilité ; tout comme on ne peut pas exprimer le dilemme du prisonnier classique (présenté exemple 1.1 (page 10)), qui lui nécessite 4 niveaux d'utilité.

On aimerait donc incorporer des préférences non dichotomiques aux jeux booléens afin que chaque joueur puisse représenter ses préférences de manière plus souple. Nous voudrions pouvoir associer une relation de préférence quelconque à l'ensemble des profils de stratégies S.

Définition 3.1. *Une* **relation de préférence** $\succeq$ *est une relation binaire reflexive et transitive (non nécessairement totale*[1]*) sur S. La* **relation de préférence stricte** $\succ$ *associée à* $\succeq$ *est définie par $s_1 \succ s_2$ si et seulement si $s_1 \succeq s_2$ et non($s_2 \succeq s_1$).*

Ces préférences peuvent être de différents types : elles peuvent être numériques (on parlera alors de préférences *cardinales*), ou encore *ordinales*. Les préférences cardinales consistent à associer à chaque issue du jeu (ou profil de stratégies) une valeur numérique, et sont donc des relations complètes. Les préférences ordinales consistent à associer directement un pré-ordre sur l'ensemble des issues du jeu, et peuvent être partielles.

Le choix entre ces deux représentations dépend des notions que l'on veut étudier : certaines notions (comme par exemple les équilibres de Nash en stratégies mixtes) nécessitent des préférences cardinales, tandis que d'autres (comme les équilibres de Nash en stratégies pures ou les stratégies dominées) peuvent être définies dans un contexte purement ordinal. Les notions qui nous intéressent ici pouvant être définies avec des préférences ordinales, nous choisissons d'étudier tout particulièrement ces dernières.

Il nous faut donc choisir à présent une représentation permettant d'éviter la description explicite de la fonction d'utilité de chaque agent, qui est, comme nous l'avons déjà vu, de taille exponentielle en fonction du nombre d'agents. Pour les préférences dichotomiques, nous avons utilisé la logique propositionnelle, qui est le langage de représentation compacte le plus naturel pour ces préférences. Pour les préférences non dichotomiques, il nous faut donc trouver un langage permettant d'expri-

[1] Une relation de préférence totale est une relation transitive, reflexive et qui vérifie : $\forall x, y : x \succeq y$ ou $y \succeq x$

mer les préférences de chacun des joueurs de façon aussi concise, ou compacte, que possible. Ces langages sont appelés *langages de représentation compacte*. On définit formellement un langage de représentation de préférences [Lang, 2006] :

Définition 3.2. *Un* **langage de représentation de préférences** *est un couple* $\langle L, Ind \rangle$ *où*

- *L est un langage propositionnel formé sur V,*
- *Ind est une fonction de L dans $Pref$, $Pref$ étant l'ensemble des relations de préférence sur 2^V, qui associe à chaque élément de L la relation de préférence induite. Dans le cas des préférences cardinales, la relation de préférence est obtenue par l'intermédiaire d'une fonction d'utilité u.*

Exemple 3.1. *Soit $V = \{a, b, c, d\}$ un ensemble de variables propositionnelles. Nous pouvons définir le langage de représentation de préférences dichotomique de la façon suivante :*

- *L_{dicho} est le langage formé sur V,*
- *u_{dicho} est définie comme suit : pour toute formule $\varphi \in L_{dicho}$, et pour tout ensemble $w \subseteq V$,*
  - *$u^{\varphi}_{dicho}(w) = 1$ si $w \models \varphi$,*
  - *$u^{\varphi}_{dicho}(w) = 0$ sinon.*

Ce langage est celui que nous avons utilisé pour définir les jeux booléens tout au long du chapitre 2 (page 31).

Il existe de nombreux langages de représentation compacte de préférences. On peut citer par exemple :

- **Logique des pondérations** [Pinkas, 1991; Haddawy et Hanks, 1992; Dupin de Saint Cyr *et al.*, 1992; Benferhat *et al.*, 2001]. Le principe de cette famille de langage de représentation de préférences est d'associer des pondérations numériques à des formules propositionnelles. Il n'est pas nécessaire que les poids soient numériques : il est possible d'utiliser une échelle plus ou moins qualitative munie d'une fonction d'agrégation, comme par exemple la logique possibiliste.
- **Logique des priorités**. Cette famille de langage de représentations est la contrepartie ordinale des logiques à pondérations. Nous la présenterons plus longuement en section 3.3 (page 99).
- **Logique des distances** [Katsuno et Mendelzon, 1991, 1992; Lafage et Lang, 2000, 2001; Benferhat *et al.*, 2002]. L'idée est d'utiliser des distances entre interprétations propositionnelles : si un agent doit satisfaire un but G, plus l'interprétation est "loin" de G, moins elle est satisfaisante.
- **Logique des préférences du type "ceteris paribus"**. Cette famille de logiques des préférences est construite sur le principe de l'interprétation *ceteris paribus*, ou encore *toutes choses étant égales par ailleurs*, ou plus généralement *toutes choses non pertinentes étant égales*. Le principe consistant à interpréter des préférences ceteris paribus est dû à [von Wright, 1963], et a été revu par [Hansson, 1989, 2001]. Un langage de représentation graphique, les CP-nets, est fondé sur le critère de comparaison ceteris paribus. Nous présentons les CP-nets en section 3.2 (page 74).
- **Logique des conditionnels**. L'idée est ici d'utiliser la logique des conditionnels [Lewis, 1973] pour représenter des préférences, et permettre de réviser ces préférences [Boutilier, 1994; Lang, 1996; Lang *et al.*, 2002b].

Mis à part le langage de représentation dichotomique présenté au chapitre 2 (page 31), il existe d'autres travaux dans le cadre des jeux booléens : des préférences non dichotomiques ont déjà été introduites dans ces jeux. Il s'agit des jeux d'évaluation distribués, introduits par [Harrenstein, 2004a], qui sont des jeux booléens à n joueurs dont les préférences sont représentées par des ensembles de formules propositionnelles : plus le nombre de formules satisfaites pour un agent est grand, plus cet agent est satisfait. Cette relation de préférences simple est pourtant déjà assez sophistiquée : elle permet de représenter des jeux plus complexes avec des jeux booléens, comme par exemple le dilemme classique du prisonnier.

Nous allons donc continuer ici dans cette voie, et étudier deux nouveaux langages de représentation de préférences : les CP-nets et les logiques des priorités[2] qui permettront eux aussi de traiter des préférences non dichotomiques.

Nous pouvons à présent définir un jeu booléen ayant des préférences ordinales.

Définition 3.3. *Soit un langage propositionnel L pour une représentation compacte de préférences. Un L-***jeu booléen** *est défini par un 5-uple* $G = (N, V, \pi, \Gamma, \Phi)$*, avec*

- $N = \{1, \ldots, n\}$, V, π *et* Γ *étant définis comme précédemment, et*
- $\Phi = \langle \Phi_1, \ldots, \Phi_n \rangle$. *Pour chaque i,* Φ_i *est une représentation compacte, dans L, de la relation de préférences*[3] *de* $\succeq_i$ *de l'agent i sur*[4] *S. On note* $Pref_G = \langle \succeq_1, \ldots, \succeq_n \rangle$.

3.1 Préférences ordinales et théorie des jeux

Nous devons redéfinir ici les équilibres de Nash en utilisant des préférences ordinales. En effet, les équilibres de Nash sont classiquement définis pour des jeux avec des préférences totales, ce qui n'est plus nécessairement le cas ici, les préférences ordinales pouvant être partielles. L'introduction de ces préférences partielles peut être justifiée de façon épistémique : il se peut qu'un agent ne puisse pas donner ses préférences sur deux états, car il n'a pas toutes les informations nécessaires pour le faire. Par exemple, supposons que Mylen doive réserver un billet d'avion pour Florian. Un avion part à 10h, l'autre à 21h. Si Mylen ne connaît pas les disponibilités de Florian, elle est dans l'incapacité de comparer, et de choisir, entre ces deux vols. Ces deux états sont donc incomparables. Un autre type d'incomparabilité est l'incomparabilité intrinsèque : deux états peuvent être incomparables du fait de leur nature. Par exemple la majorité des gens sont incapables de répondre à la question “Préférez-vous avoir sur votre ordinateur un virus qui efface tous vos messages électroniques, ou un virus qui envoie tous ces messages à tout vos contacts ?”

Nous allons donc définir deux notions d'équilibre de Nash, une forte et une faible (elles sont équivalentes aux notions d'équilibre maximal et maximum dans [Harrenstein, 2004a]).

On rappelle qu'un équilibre de Nash en stratégies pures est un profil de stratégies tel que la stratégie de chaque joueur est une réponse optimale aux stratégies des autres joueurs.

Définition 3.4. *Soit* $G = (N, V, \Gamma, \pi, \Phi)$ *un jeu booléen, avec* $N = \{1, \ldots, n\}$ *l'ensemble des joueurs,* $\Phi = \langle \Phi_1, \ldots, \Phi_n \rangle$ *l'ensemble des buts des joueurs, et* $Pref_G = \langle \succeq_1, \ldots, \succeq_n \rangle$ *l'ensemble des préférences de chaque joueur. Deux définitions, une faible et une forte, peuvent ici caractériser les équilibres de Nash.*

[2] Les résultats donnés dans ce chapitre ont fait l'objet de plusieurs publications : [Bonzon *et al.*, 2006a, 2007a]

[3] La définition des buts Φ correspond à la définition de *Ind* dans la définition 3.2 (page ci-contre).

[4] On rappelle que dans un jeu booléen, $S_i = \{s_i \in 2^{\pi_i} | s_i \models \gamma_i\}$, et que $S = S_1 \times \ldots \times S_n$.

$s = (s_1, \ldots, s_n)$ *est un* **équilibre de Nash faible en stratégies pures** *si et seulement si :*

$$\forall i \in \{1, \ldots, n\}, \forall s'_i \in S_i, (s'_i, s_{-i}) \not\succ_i (s_i, s_{-i}) \tag{3.1}$$

$s = (s_1, \ldots, s_n)$ *est un* **équilibre de Nash fort en stratégies pures** *si et seulement si :*

$$\forall i \in \{1, \ldots, n\}, \forall s'_i \in S_i, (s'_i, s_{-i}) \preceq_i (s_i, s_{-i}) \tag{3.2}$$

L'ensemble des équilibres de Nash forts (resp. faibles) en stratégie pure sera noté NE_{fort} (resp. NE_{faible}).

Un équilibre de Nash fort en stratégies pures est donc un profil de stratégies dont on a la certitude qu'il est une réponse optimale aux stratégies des autres joueurs. Un équilibre de Nash faible en stratégies pures est lui un profil de stratégies qui *peut* être une réponse optimale aux stratégies des autres joueurs.

Il est clair que tout équilibre de Nash fort est un équilibre de Nash faible. On a donc $NE_{fort}(G) \subseteq NE_{faible}(G)$.

Exemple 3.2. *Soit le L-jeu booléen $G = (N, V, \Gamma, \pi, \Phi)$ suivant :*

- $N = \{1, 2\}$,
- $V = \{a, b\}$,
- $\gamma_1 = \gamma_2 = \top$,
- $\pi_1 = \{a\}$, $\pi_2 = \{b\}$,
- $\succeq_1$ *est défini par : $ab \succ_1 a\overline{b} \succ_1 \overline{a}b$ et $ab \succ_1 \overline{a}\overline{b} \succ_1 \overline{a}b$ (les profils de stratégies $a\overline{b}$ et $\overline{a}\overline{b}$ sont incomparables pour le joueur 1), et*
- $\succeq_2$ *est défini par : $\overline{a}\overline{b} \succ_2 ab \succ_2 a\overline{b} \succ_2 \overline{a}b$.*

Ce jeu a un équilibre de Nash fort : $NE_{fort}(G) = \{ab\}$ et deux équilibres de Nash faibles : $NE_{faible}(G) = \{ab, \overline{a}\overline{b}\}$.

Nous devons également raffiner la notion de stratégies dominées, définies initialement à partir des préférences totales (définitions 1.8 et 1.9 (page 22)). L'introduction de préférences partielles, et donc de la notion d'incomparabilité, nous permet d'introduire un nouveau cas de stratégie dominée. Cette nouvelle notion est très faible : toutes les stratégies peuvent être partiellement dominées.

Définition 3.5. *Soit i un joueur et $\succeq_i$ sa relation de préférence sur S. La stratégie s_i du joueur i est dite* **strictement dominée** *s'il existe une autre stratégie s'_i telle que, quelles que soient les stratégies des autres joueurs, s'_i est strictement préférée à s_i pour $\succeq_i$. Donc, $s_i \in S_i$ est* **strictement dominée** *si et seulement si :*

$$\exists s'_i \in S_i \; telle\ que\ \forall s_{-i} \in S_{-i}, (s_i, s_{-i}) \prec_i (s'_i, s_{-i})$$

Soit i un joueur et $\succeq_i$ sa relation de préférence sur S. La stratégie s_i du joueur i est dite **faiblement dominée** *s'il existe une autre stratégie s'_i telle que, quelles que soient les stratégies des autres joueurs, s'_i est préférée à s_i pour $\succeq_i$, et qu'il existe au moins une combinaison des stratégies des autres joueurs telle que s'_i soit strictement préférée à s_i pour $\succeq_i$. Donc, $s_i \in S_i$ est* **faiblement dominée** *si $\exists s'_i \in S_i$ telle que :*

$$\forall s_{-i} \in S_{-i}, (s_i, s_{-i}) \preceq_i (s'_i, s_{-i}) \text{ et que } \exists s_{-i} \in S_{-i} \; telle\ que\ (s_i, s_{-i}) \prec_i (s'_i, s_{-i})$$

Soit i un joueur et $\succeq_i$ sa relation de préférence sur S. La stratégie s_i du joueur i est dite **partiellement dominée** *s'il existe une autre stratégie s'_i telle que, quelles que soient les stratégies des autres joueurs, s_i n'est pas strictement préférée à s'_i pour $\succeq_i$, et qu'il existe au moins une combinaison des stratégies des autres joueurs telle que s'_i soit strictement préférée à s_i pour $\succeq_i$. Donc, $s_i \in S_i$ est* **partiellement dominée** *si $\exists s'_i \in S_i$ telle que :*

$$\forall s_{-i} \in S_{-i}, (s_i, s_{-i}) \not\succ_i (s'_i, s_{-i}) \textit{ et que } \exists s_{-i} \in S_{-i} \textit{ telle que } (s_i, s_{-i}) \prec_i (s'_i, s_{-i})$$

Toute stratégie strictement dominée est faiblement dominée ; et toute stratégie faiblement dominée est partiellement dominée. Lorsque la relation $\succeq_i$ est un pré-ordre total, il y a équivalence entre stratégies faiblement et partiellement dominées.

Exemple 3.3. *Soit le L-jeu booléen $G = (N, V, \Gamma, \pi, \Phi)$ suivant :*

- $N = \{1, 2\}$,
- $V = \{a, b\}$,
- $\gamma_1 = \gamma_2 = \top$,
- $\pi_1 = \{a\}$, $\pi_2 = \{b\}$,
- $\succeq_1$ *est défini par : $ab \succ_1 a\overline{b} \succ_1 \overline{a}b$ et $ab \succ_1 \overline{a}\overline{b} \succ_1 \overline{a}b$ (les profils de stratégies $a\overline{b}$ et $\overline{a}\overline{b}$ sont incomparables pour le joueur 1), et*
- $\succeq_2$ *est défini par : $ab \succeq_2 \overline{a}b \succeq_2 a\overline{b} \succ_2 \overline{a}\overline{b}$.*

Etudions les stratégies dominées de ce jeu :

- *Joueur 1 : La stratégie a du joueur 1 domine partiellement sa stratégie $\overline{a}$. En effet, $ab \succ_1 \overline{a}b$, et $\overline{a}\overline{b} \not\succ_1 a\overline{b}$.*
- *Joueur 2 : La stratégie b du joueur 2 domine faiblement (et partiellement) sa stratégie $\overline{b}$. En effet, par transitivité de $\succeq_2$, $ab \succeq_2 a\overline{b}$, et $\overline{a}b \succ_2 \overline{a}\overline{b}$.*

La propriété 2.14 (page 51), issue de la théorie des jeux, reste valable ici :

Propriété 3.1.

- *l'ordre d'élimination des stratégies strictement dominées* n'affecte pas *le résultat final.*
- *Par contre, l'ordre d'élimination des stratégies faiblement (et donc partiellement) dominées* affecte *le résultat final.*

Preuve :

- Montrons que l'ordre d'élimination des stratégies strictement dominées n'affecte pas le résultat final. Supposons que nous sommes en présence d'un jeu G ayant au moins deux stratégies strictement dominées. Ces deux stratégies peuvent être :
  - deux stratégies d'un joueur donné.

    Dans ce cas, on suppose que la stratégie s_{i_1} du joueur i est dominée strictement par la stratégie s'_i, et que s_{i_2} est dominée strictement par s''_i. On a alors 3 possibilités, 2 d'entre elles étant équivalentes :
    - $s_{i_2} = s'_i$ (ou $s_{i_1} = s''_i$) : la stratégie s_{i_1} est strictement dominée par s_{i_2}. Comme s''_i domine strictement s_{i_2}, alors s''_i domine strictement s_{i_1}. En effet si on a : $\forall s_{-i} \in S_{-i}, (s_{i_2}, s_{-i}) \succ_i s_{i_1}, s_{-i})$ et $(s''_i, s_{-i}) \succ_i (s_{i_2}, s_{-i})$ alors

on a, par transitivité de $\succ$: $\forall s_{-i} \in S_{-i}, s_i'', s_{-i} \succ_i s_{i_1}, s_{-i}$. Dans ce cas, si on élimine s_{i_1} en premier, s_{i_2} sera toujours strictement dominée par s_i'' : et si on élimine s_{i_2} en premier, s_{i_1} sera toujours strictement dominée par s_i''.

∗ $s_{i_2} \neq s_i'$ et $s_{i_1} \neq s_i''$: dans ce cas, l'élimination de l'une de ces stratégies n'influe pas sur l'élimination de l'autre.

∗ des stratégies de deux joueurs différents.

Nous supposons ici que la stratégie s_i du joueur i est dominée strictement par la stratégie s_i', et que la stratégie s_j du joueur j est dominée strictement par s_j'. On a alors :

$$\forall s_{-i} \in S_{-i}, (s_i', s_{-i}) \succ_i (s_i, s_{-i})$$

$$\forall s_{-j} \in S_{-j}, (s_j', s_{-j}) \succ_j (s_j, s_{-j})$$

Si on élimine s_i en premier, la stratégie s_j sera toujours strictement dominée par s_j'. En effet, on aura toujours $\forall s_{-j} \in S_{-j}, (s_j', s_{-j}) \succ_j (s_j, s_{-j})$ car l'ensemble des $s_{-j} \in S_{-j}$ diminue après la suppression de s_i.

On raisonne de même si on élimine s_j en premier.

∗ On montre sur l'exemple suivant que l'ordre d'élimination des stratégies faiblement dominées affecte le résultat final :

Soit un L-jeu booléen tel que $N = \{1,2\}$, $V = \{a,b\}$, $\gamma_1 = \gamma_2 = \top$, $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\succeq_1$ définie par : $ab \succeq_1 \overline{a}b \succeq_1 a\overline{b} \succ_1 \overline{a}\overline{b}$, et $\succeq_2$ définie par : $ab \succeq_2 a\overline{b} \succeq_2 \overline{a}b \succ_2 \overline{a}\overline{b}$.

∗ Pour le joueur 1, a domine faiblement $\overline{a}$. En effet, on a $ab \succeq_1 \overline{a}b$ et $a\overline{b} \succ_1 \overline{a}\overline{b}$. Eliminons cette stratégie. On obtient alors 2 états, ab et $a\overline{b}$, et 2 n'a aucune stratégie dominée.

∗ Pour le joueur 2, b domine faiblement $\overline{b}$. En effet, on a $ab \succeq_2 a\overline{b}$ et $\overline{a}b \succ_2 \overline{a}\overline{b}$. Eliminons cette sratégie. On obtient alors 2 états, ab et $\overline{a}b$, et 1 n'a aucune stratégie dominée.

On voit ici que si l'on élimine $\overline{a}$ ou $\overline{b}$ en premier, on obtient des résultats différents.

Comme toute stratégie faiblement dominée est partiellement dominée, ce contre-exemple est valable pour les stratégies partiellement dominées.

■

Nous pouvons à présent étudier des langages de représentation compacte de préférences et essayer de les adapter à notre modèle. Nous étudierons dans un premier temps un langage de représentation dit “graphique”, les CP-nets ; puis un langage fondé sur la représentation de préférences en logique propositionnelle, les buts à priorité.

3.2 Jeux booléens et CP-nets

Les CP-nets appartiennent à une famille de langages de représentation dit “graphiques”. Ce langage est fondé sur le critère de comparaison Ceteris Paribus que nous allons présenter dans la section suivante.

3.2.1 Ceteris Paribus

Quand un agent exprime en langage naturel une préférence telle que “une table ronde sera mieux dans le salon qu’une table carrée”, il ne veut sans doute pas dire que n’importe quelle table ronde sera préférée à n’importe quelle table carrée. Il veut exprimer le fait qu’il préfèrera une table ronde à une table carrée si ces deux tables ne diffèrent pas significativement dans leurs autres caractéristiques (la taille, la couleur, le bois utilisé, les finitions ou encore le prix). Le principe qui est à l’œuvre dans l’interprétation de telles préférences est que les alternatives doivent être comparées *toutes choses étant égales par ailleurs*, ou encore *Ceteris Paribus*. Les préférences *Ceteris Paribus* ont été tout d’abord étudiées par von Wright dans [von Wright, 1963], puis ont été considérablement revues par Hansson [Hansson, 1989, 2001], qui en a proposé une généralisation fondée sur les fonctions de représentation et qui en a étudié les propriétés logiques. Ces préférences ont été indépendamment redécouvertes dans la communauté IA par [Doyle et Wellman, 1991; Doyle *et al.*, 1991], et étudiées notamment par [Boutilier *et al.*, 1999].

Soit $V = \{X_1, \ldots, X_n\}$ un ensemble de variables, chaque variable X_i étant associée à un domaine $D(X_i)$. Si $\{X_1, \ldots, X_p\}$ est contenu dans V, alors $D(\{X_1 \ldots X_p\}) = D(X_1) \times \ldots \times D(X_p)$, produit cartésien des domaines de chaque variable.

Définition 3.6. *Un sous-ensemble de variables X est* **préférentiellement indépendant** *à son complément $Y = V \setminus X$ si et seulement si $\forall x_1, x_2 \in D(X)$ et $\forall y_1, y_2 \in D(Y)$ on a :*

$$x_1 y_1 \succeq x_2 y_1 \text{ si et seulement si } x_1 y_2 \succeq x_2 y_2$$

On dit que x_1 est **préférée** *à x_2* **ceteris paribus***.*

En d’autres mots, la relation de préférence sur les instanciations de X, quand toutes les autres variables sont fixées, est la même quelles que soient les valeurs de ces autres variables.

Définition 3.7. *Soit X, Y et Z trois ensembles non vides disjoints formant une partition de V. X et Y sont* **conditionnellement préférentiellement indépendants étant donné** *Z si et seulement si $\forall z \in D(Z)$, $\forall x_1, x_2 \in D(X)$ et $\forall y_1, y_2 \in D(Y)$ on a :*

$$x_1 y_1 z \succeq x_2 y_1 z \text{ si et seulement si } x_1 y_2 z \succeq x_2 y_2 z$$

Pour une valeur de Z fixée, la relation de préférence sur les instanciations de X est la même quelles que soient les valeurs des instanciations de Y.

3.2.2 CP-nets

3.2.2.1 Définitions générales

Les CP-nets ont été introduits dans [Boutilier *et al.*, 1999] comme un outil pour représenter compactement les relations de préférences qualitatives. Ce modèle graphique utilise l’indépendance préférentielle conditionnelle pour structurer les préférences d’un agent sous l’hypothèse *Ceteris Paribus*. Ils ont été principalement étudiés dans [Domshlak, 2002], [Boutilier *et al.*, 2004a] ou encore [Boutilier *et al.*, 2004b].

Les CP-nets peuvent être utilisés dans le cadre des jeux booléens de façon à représenter les préférences de chaque joueur. On ne considèrera plus que les buts des joueurs sont représentés par une formule logique, mais directement par un CP-net propositionnel (pour lequel les domaines des variables sont binaires). Dans ce contexte, $\forall x_i \in V, D(x_i) = \{x_i, \overline{x_i}\} = 2^{x_i}$ et $D(\{x_1 \dots x_p\}) = 2^{\{x_1,\dots,x_p\}}$. Ainsi, un élément de $D(x_i)$ correspond à une $\{x_i\}$-interprétation, et un élément de $D(\{x_1 \dots x_p\})$ est une $\{x_1 \dots x_p\}$-interprétation.

Définition 3.8. *Pour chaque variable $X \in V$, un ensemble de* **variables parents***, noté $Pa(X)$, est spécifié. Ces variables sont celles qui influent sur les préférences de l'agent entre les différentes valeurs de X. Formellement, X et $V \setminus (\{X\} \cup Pa(X))$ sont conditionnellement préférentiellement indépendantes étant donné $Pa(X)$.*

La **table de préférence conditionnelle** *(notée CPT) décrit les préférences de l'agent sur les valeurs de la variable X, étant données toutes les combinaisons des valeurs des variables parents.*

Pour chaque instanciation de $Pa(X)$, $CPT(X)$ spécifie un ordre total *sur $D(X)$ tel que $\forall x_i, x_j \in D(X)$ on ait soit $x_i \succ x_j$, soit $x_j \succ x_i$.*

Soit $V = \{X_1, \dots, X_n\}$ un ensemble de variables. $\mathcal{N} = \langle \mathcal{G}, \mathcal{T} \rangle$ est un **CP-net** *sur V, $\mathcal{G}$ étant un graphe orienté sur V, et $\mathcal{T}$ étant un ensemble de tables de préférences conditionnelles $CPT(X_i)$ pour chaque $X_i \in V$. Chaque table de préférence conditionnelle $CPT(X_i)$ est associée à un ordre total $\succ^i_p$, selon chaque instanciation $p \in D(Pa(X_i))$.*

Illustrons ces définitions avec un exemple simple :

Exemple 3.4. *Soit le CP-net présenté sur la figure 3.1 qui exprime mes préférences sur le menu du dîner. Ce CP-net est composé de deux variables, S et V qui correspondent respectivement à la soupe et au vin. Je préfère strictement manger une soupe de poisson (S_p) plutôt qu'une soupe de légumes (S_l), tandis que mes préférences sur le vin rouge (V_r) ou blanc (V_b) dépendent de la soupe que je mangerai. En effet je préfère du vin rouge avec une soupe de légumes, mais du blanc avec une soupe de poisson. On a donc $D(S) = \{S_p, S_l\}$ et $D(V) = \{V_r, V_b\}$.*

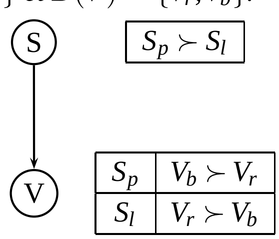


Figure 3.1 **—** CP-net "Mon dîner simple"

3.2.2.2 Sémantique des CP-nets

La sémantique des CP-nets a été principalement étudiée dans [Domshlak, 2002], [Boutilier *et al.*, 2004a] ou encore [Boutilier *et al.*, 2004b].

Informellement, un CP-net $\mathcal{N}$ est satisfait par $\succ$ si $\succ$ satisfait chacune des préférences conditionnelles exprimées dans les CPTs de $\mathcal{N}$ sous l'interprétation *ceteris paribus*.

Définition 3.9. *Soit $\mathcal{N}$ un CP-net sur les variables V. Soit $X \in V$ une variable, $U \subseteq V$ les parents de X dans $\mathcal{N}$, et $Y = V \setminus (U \cup \{X\})$. Soit $\succ_u$ une relation d'ordre sur $D(X)$ dictée par $CPT(X)$ pour l'instanciation $u \in D(U)$ des parents de X. Soit enfin $\succ$ une relation de préférence sur $D(V)$.*

1. *Une relation de préférence $\succ$* **satisfait** *$\succ_u$ si et seulement si $yux_i \succ yux_j$, pour tout $y \in D(Y)$, à chaque fois que $x_i \succ_u x_j$.*
2. *Une relation de préférence $\succ$* **satisfait** *$CPT(X)$ si et seulement si $\succ$ satisfait $\succ_u$ pour chaque $u \in D(U)$.*
3. *Une relation de préférence $\succ$* **satisfait** *le CP-net $\mathcal{N}$ si et seulement si elle satisfait $CPT(X)$ pour chaque variable X.*

Un CP-net $\mathcal{N}$ est **satisfiable** *si et seulement si il existe une relation $\succ$ qui le satisfait.*

Définition 3.10. *Soit $\mathcal{N}$ un CP-net sur les variables V, et $o, o' \in D(V)$ deux états quelconques. $\mathcal{N}$* **entraîne** *$o \succ o'$ (i.e. l'état o est préféré à o'), noté $\mathcal{N} \models o \succ o'$, si et seulement si $o \succ o'$ dans chaque relation de préférence qui satisfait $\mathcal{N}$. On dit alors que $o \succ o'$ est une* **conséquence** *de $\mathcal{N}$.*

L'ensemble des conséquences $o \succ o'$ d'un CP-net constitue un ordre partiel sur les états : o est préferré à o' dans cette relation d'ordre si et seulement si $\mathcal{N} \models o \succ o'$. Cet ordre partiel peut être représenté par un graphe orienté, que l'on appelera graphe de préférences induit, ou pré-ordre associé à $\mathcal{N}$:

1. Les nœuds du graphe de préférences correspondent à l'affectation de toutes les variables du CP-net,
2. il y a un arc du nœud o au nœud o' si et seulement si les affectations de o et o' ne diffèrent que pour la valeur d'une seule variable X, et si, étant donné $Pa(X)$, la valeur assignée par o à X est préférée à la valeur assignée par o' à X[5].

La fermeture transitive de ce graphe spécifie les ordres partiels sur les états induits par le CP-net. Plus précisément, nous avons $\mathcal{N} \models o \succ o'$ si et seulement si il existe un chemin de o à o' dans le graphe de préférences induit par $\mathcal{N}$.

Formellement, la relation de préférence induite par $\mathcal{N}$, représentée par le graphe de préférences induit, est définie comme suit :

Définition 3.11. *La* **relation de préférence** *sur les états induite par un CP-net $\mathcal{N}$ est dénotée par $\succ_{\mathcal{N}}$, et est définie par $\forall o, o' \in D(V)$, $o \succ_{\mathcal{N}} o'$ si et seulement si $\mathcal{N} \models o \succ o'$.*

Exemple 3.4 (page précédente), suite : *La figure 3.2 (page suivante) représente la relation de préférence induite par ce CP-net. L'élément du bas ($S_l \wedge V_b$) représente le pire état tandis que l'élément du haut ($S_p \wedge V_b$) est le meilleur état possible.*

Nous remarquons ici que nous avons une flèche entre les nœuds $(S_l \wedge V_b)$ et $(S_p \wedge V_b)$ car nous comparons les états 2 par 2, toutes choses étant égales par ailleurs.

Nous pouvons donc ordonner totalement les états possibles (du plus préféré au moins préféré) :

$$(S_p \wedge V_b) \succ (S_p \wedge V_r) \succ (S_l \wedge V_r) \succ (S_l \wedge V_b)$$

[5] La représentation graphique des préférences que nous utilisons ici dans le cadre des CP-nets est contraire à la représentation utilisée dans la littérature [Boutilier *et al.*, 2004a,b] : dans cette dernière les flèches vont de l'état le moins préféré vers le plus préféré. Afin d'homogénéiser cette section avec la section 3.3 (page 99) nous choisissons ici d'orienter les flèches de l'état le plus préféré vers l'état le moins préféré.

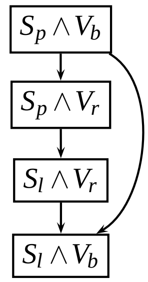


***Figure 3.2* —** Graphe de préférences induit par le CP-net “Mon dîner simple”

Cette relation $\succ$ est la seule relation de préférence qui satisfasse ce CP-net.

Théorème 3.1 ([Boutilier *et al.*, 2004a])**.** *Tout CP-net acyclique est satisfiable.*

Lemme 3.1 ([Boutilier *et al.*, 2004a])**.** *La conséquence préférentielle en ce qui concerne un CP-net est transitive. Donc, si $o \succ_{\mathcal{N}} o'$ et $o' \succ_{\mathcal{N}} o''$ alors $o \succ_{\mathcal{N}} o''$.*

Ce lemme nous permet de simplifier les futurs schémas. Par exemple, sur la figure 3.2 de l'exemple 3.4, nous pouvons supprimer la flèche entre les nœuds $(S_l \wedge V_b)$ et $(S_p \wedge V_b)$ car elle est déductible par transitivité.

Définition 3.12. *Soit $\mathcal{N}$ un CP-net sur les variables V. $D(V)$ dénote l'ensemble de toutes les instanciations de V.*

On appelle **ensemble des résultats optimaux de** *$\mathcal{N}$ l'ensemble d'états O tel que $\forall o \in O$, $\mathcal{N} \models o \succ o'$ pour tout $o' \in D(V) \setminus O$.*

Chercher le résultat optimal d'un CP-net $\mathcal{N}$ consiste intuitivement à parcourir le graphe de haut en bas (c'est-à-dire des parents aux descendants), en instanciant chaque variable à sa valeur préférée selon l'instanciation de ses parents. Cette procédure est appelée *recherche avant* (*forward sweep*).

Par exemple, sur la figure 3.2 de l'exemple 3.4, le résultat optimal sera $(S_p \wedge V_b)$.

Dans [Boutilier *et al.*, 1999], il est montré que si le CP-net $\mathcal{N}$ est acyclique, le résultat sera unique. Par contre, en cas d'un graphe cyclique, il peut y avoir plusieurs résultats optimaux, comme il peut n'y en avoir aucun.

Lemme 3.2. [Boutilier *et al.*, 1999] *Soit $\mathcal{N}$ un CP-net acyclique sur un ensemble de variables V. La procédure de recherche en avant (forward sweep procedure) construit le résultat optimal dans $D(V)$.*

Les CP-nets sont utilisés dans [Apt *et al.*, 2005], pour représenter des jeux : les CP-nets sont vus comme des jeux en forme normale et vice versa. Il est alors montré dans [Apt *et al.*, 2005] qu'un profil de stratégies est un équilibre de Nash du jeu G si et seulement si c'est un résultat optimal du CP-net $\mathcal{N}(G)$. De même, dans la traduction inverse, il est montré qu'un profil de stratégies est un résultat optimal d'un CP-net si et seulement si c'est un équilibre de Nash du jeu associé. Nous en parlerons plus longuement dans la section 4.2.1 (page 127).

3.2.2.3 Utilisation des CP-nets dans les jeux booléens

Les définitions classiques des CP-nets doivent être adaptées pour pouvoir être appliquées à des jeux booléens. En effet, nous sommes ici dans un contexte particulier où les variables que nous manipulons sont booléennes, et où nous avons plusieurs joueurs. Dans ce contexte, chaque joueur doit pouvoir choisir la stratégie qui lui permettra d'obtenir le meilleur résultat possible. Pour ce faire, nous pouvons utiliser les notions définies en section 3.1 (page 71) pour calculer les équilibres de Nash et les stratégies dominées, condition de disposer d'une relation de préférence pour chaque joueur sur l'ensemble des profils de stratégies. Les CP-nets interviennent puisqu'ils constituent un outil de représentation des préférences et fourniront ainsi les relations de préférence de chaque joueur.

Définition 3.13 La table de préférence conditionnelle du joueur i pour un jeu booléen (notée $CPT_i(X)$) décrit les préférences du joueur i sur les valeurs de la variable X, étant donné toutes les combinaisons des valeurs des variables parents.
Pour chaque instanciation p de $Pa_i(X)$, $CPT_i(X)$ spécifie un ordre total tel que l'on ait soit $x \succ_{i,p} \bar{x}$, soit $\bar{x} \succ_{i,p} x$.

Définition 3.14 Un CP-jeu booléen est un 5-tuple $G = (N, V, \Gamma, \pi, \Phi)$, où $N = \{1, \dots, n\}$ est un ensemble de joueurs, $V = \{x_1, \dots, x_n\}$ est un ensemble de variables et $\Phi = \langle \mathcal{N}_1, \dots, \mathcal{N}_n \rangle$, chaque $\mathcal{N}_i$ étant un CP-net sur V et pour tout $i \in N$, $\succ_i = \succ_{\mathcal{N}_i}$.

La grande différence entre les CP-nets classiques et ceux que nous introduisons dans les jeux booléens est l'introduction de contraintes. Les CP-nets avec contraintes ont été étudiés parallèlement dans [Boutilier et al., 2004b] et [Prestwich et al., 2004].

Dans [Prestwich et al., 2004], les préférences sont exprimées sur toutes les issues du jeu, les états ne respectant pas les contraintes étant considérés comme inaccessibles. S'il existe un état accessible et non dominé, ce dernier est le résultat optimal.

Dans [Boutilier et al., 2004b], les préférences exprimées dans les CPTs du CP-net doivent respecter les contraintes : si par exemple une contrainte est du type $a \to b$, et que $a \in P(b)$, les préférences sur b seront identiques à celles sur a (si a est indépendant et que $a \succ \bar{a}$, alors on aura $b \succ b'$). Un algorithme appelé Search-CP permet alors de trouver l'ensemble de résultats optimaux qui sont accessibles (c'est-à-dire qui respectent les contraintes) et qui ne sont pas dominés par un autre état accessible. Nous choisissons ici d'utiliser cette méthode.

Exemple 3.5 Soit le jeu $G = (N, V, \Gamma, \pi, \Phi)$ suivant :

- $N = \{1, 2, 3\}$
- $V = \{a, b, c\}$, avec $D(a) = \{a, \bar{a}\}$, $D(b) = \{b, \bar{b}\}$ et $D(c) = \{c, \bar{c}\}$.
- $\gamma_1 = \gamma_2 = \gamma_3 = \top$,
- $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\pi_3 = \{c\}$,
- Le but du joueur 1 est représenté par le CP-net et la relation de préférence induite figure 3.3 (page suivante),
- Le but du joueur 2 est représenté par le CP-net et la relation de préférence induite figure 3.4 (page suivante),
- Le but du joueur 3 est représenté par le CP-net et la relation de préférence induite figure 3.5 (page 81).

Pour calculer facilement les équilibres de Nash de ce jeu, nous mettons en exergue les relations qui nous intéressent pour chaque joueur. Ces relations apparaissent sous la forme de flèches en pointillé sur la figure 3.3 (page suivante).

En effet, pour le joueur 1, qui contrôle la variable a, nous comparons les états abc et $\bar{a}bc$, les états $ab\bar{c}$, $\bar{a}b\bar{c}$, les états $a\bar{b}c$ et $\bar{a}\bar{b}c$, et enfin les états $a\bar{b}\bar{c}$ et $\bar{a}\bar{b}\bar{c}$.