

École des Hautes Études Commerciales
Affiliée à l'Université de Montréal

Le problème de la satisfiabilité probabiliste
qualitative

par

Sylvain Perron

Sciences de la gestion

Travail dirigé présenté en vue de l'obtention du
grade de maître ès sciences

(M.Sc)

Option

Modélisation et décision

Décembre 1997

©Sylvain Perron, 1997

École des Hautes Études Commerciales
Affiliée à l'Université de Montréal

Ce travail dirigé intitulé :
Le problème de la satisfiabilité probabiliste
qualitative

présenté par :
Sylvain Perron

a été évalué par un jury composé des personnes
suivantes :
Fabien Chauny
et
Pierre Hansen

Travail dirigé accepté le 12 mars 1998

SOMMAIRE

Ce travail de recherche vise à étudier une extension du *problème de la satisfiabilité probabiliste* incorporant des probabilités qualitatives. Le cadre général du problème consiste à considérer un ensemble d'événements et leur probabilité d'être vrai. La probabilité de chaque événement est soit totalement inconnue ou comprise à l'intérieur d'un intervalle. On considère également un ensemble de relations d'inégalité stricte ou non stricte entre les probabilités de certaines paires d'événements. Le premier problème que l'on veut alors résoudre consiste à vérifier si les jugements de probabilité sont cohérents. Si les jugements sont cohérents, on pourrait aussi vouloir analyser la probabilité d'être vrai d'un événement supplémentaire. On tenterait alors de trouver les bornes inférieure et supérieure les meilleures possibles sur cette probabilité. On serait également intéressé à déterminer, avec le moins de calculs possibles, les relations qui existent entre la probabilité du nouvel événement et celle de chacun des autres événements.

Dans ce projet de recherche, on tente de développer des méthodes algorithmiques pour résoudre efficacement ces problèmes. Ces méthodes sont basées essentiellement sur la technique de génération de colonnes jointe à la programmation non-linéaire en variables 0–1 ainsi que l'analyse paramétrique de la programmation linéaire. Des résultats numériques démontrent qu'il est possible d'utiliser ces méthodes pour résoudre des problèmes d'assez grande taille en des temps raisonnables.

Table des matières

SOMMAIRE	iii
REMERCIEMENTS	viii
LISTE DES TABLEAUX	ix
LISTE DES FIGURES	x
CHAPITRE 1 INTRODUCTION	1
1.1 Introduction au problème	1
1.2 Définition du problème et exemple	3
1.3 Méthodologie	6
CHAPITRE 2 LE PROBLÈME DE LA SATISFIABILITÉ PROBA- BILISTE	7
2.1 Introduction	7

2.2	Formulation mathématique de <i>PSAT</i>	10
2.2.1	Formulation selon Hailperin	10
2.2.2	Formulation selon Coletti	13
2.3	Résolution de <i>PSAT</i>	16
2.3.1	Méthode de génération de colonnes	16
2.3.2	Expression du problème auxiliaire	17
2.3.3	Résolution du problème auxiliaire	18
2.3.4	Méthode heuristique	18
2.3.5	Méthode exacte	21
2.4	<i>PSAT</i> avec intervalles	23
2.5	<i>PSAT</i> avec probabilités conditionnelles	24
2.5.1	Formulation selon Hansen, Jaumard et Poggi de Aragão	24
2.5.2	Formulation selon Coletti	25
2.6	Expérience de calculs sur <i>PSAT</i>	28

CHAPITRE 3 LE PROBLÈME DE LA SATISFIABILITÉ PROBABILISTE QUALITATIVE 29

3.1	Introduction	29
-----	------------------------	----

3.2	Survol des travaux antérieurs sur <i>QPSAT</i>	31
3.3	Cohérence sans inégalité stricte	35
3.3.1	Définition du problème	35
3.3.2	Formulation mathématique	36
3.3.3	Méthode de résolution	37
3.4	Cohérence avec inégalités strictes	39
3.4.1	Définition du problème	39
3.4.2	Formulation mathématique	39
3.4.3	Méthode de résolution	42
3.5	Extension sans inégalité stricte	43
3.5.1	Bornes inférieure et supérieure sur π_{m+1}	43
3.5.2	Relation entre la probabilité π_{m+1} et chacune des m probabilités π_i	45
3.6	Extension avec inégalités strictes	53
3.6.1	Bornes inférieure et supérieure sur π_{m+1}	53
3.6.2	Relation entre la probabilité π_{m+1} et chacune des m probabilités π_i	56
3.7	Discussion sur <i>QPSAT</i> avec probabilités conditionnelles	58

CHAPITRE 4 RÉSULTATS NUMÉRIQUES	61
4.1 Introduction	61
4.2 Génération de problèmes tests	62
4.3 Résultats et analyse	64
CHAPITRE 5 CONCLUSION	73
BIBLIOGRAPHIE	75

REMERCIEMENTS

En premier lieu, je voudrais remercier sincèrement mes deux directeurs Fabien Chauny et Pierre Hansen pour leurs précieux conseils, pour leur grande disponibilité ainsi que pour leurs nombreux encouragements.

Je voudrais aussi exprimer mes remerciements au CRSNG qui a bien voulu subventionner généreusement ce projet de recherche. Parallèlement, merci à la Fondation des diplômés HEC pour la remise d'une bourse d'excellence.

Je suis aussi très reconnaissant envers le GERAD qui m'a permis d'utiliser ses locaux, ses ordinateurs et son équipement dans la réalisation de ce travail de recherche. Je tiens également à remercier les membres de ce centre de recherche qui étaient continuellement présents pour m'aider à résoudre certains problèmes d'ordre technique.

Enfin, je voudrais témoigner mon extrême gratitude envers les membres de ma famille et mes amis pour leur support moral. Sans leurs encouragements, la réalisation de ce travail aurait été beaucoup plus difficile. Un merci spécial à Mylène pour avoir bien voulu agir à titre de lectrice informelle de ce texte.

Liste des tableaux

4.1	Résultats d'exécution en déterminant l'ordre des comparaisons selon trois méthodes.	65
4.2	Méthode générale vs. méthode améliorée pour la détermination des relations.	66
4.3	Résultats d'exécution sur des <i>QPSAT</i> sans inégalité stricte en faisant varier le nombre de variables et le nombre d'événements.	70
4.4	Résultats d'exécution sur des <i>QPSAT</i> avec inégalités strictes en faisant varier le nombre de variables et le nombre d'événements.	71
4.5	Résultats d'exécution sur des <i>QPSAT</i> avec inégalités strictes en faisant varier le nombre de relations.	72

Table des figures

2.1	Schéma de l'algorithme de recherche avec tabous pour la maximisation d'une fonction non-linéaire en variables 0-1.	20
3.1	Graphe d'implication de l'exemple 1.	48
3.2	Schéma de l'analyse paramétrique.	55

CHAPITRE 1

INTRODUCTION

1.1 Introduction au problème

De nombreux systèmes experts ont à prendre en compte l'incertitude de faits ou de règles. L'utilisation de la théorie des probabilités conjointement avec la logique est alors d'un grand intérêt. Le *problème de la satisfiabilité probabiliste* (Boole [7, 8, 9, 10, 11], Hailperin [39, 40, 41], Nilsson [56], Jaumard, Hansen et Poggi de Aragão [48]) ou *PSAT*, connu en intelligence artificielle sous le nom de la logique probabiliste, est alors un problème de base, très étudié et appliqué récemment. En effet, *PSAT* permet d'obtenir des intervalles les meilleurs possibles sur la vérité d'un ou de plusieurs diagnostics. L'objectif du présent travail consiste à étudier une extension de *PSAT* incorporant des probabilités qualitatives, c'est-à-dire des relations entre les probabilités de certaines paires d'événements. Cette extension peut s'avérer fort pratique pour le traitement de problèmes réels dans lesquels la probabilité associée à la véracité de certains événements est difficile à évaluer. Cependant, il se peut que dans ces situations, un expert soit en mesure de nous transmettre des jugements du type "A est plus probable que B" ou "A est au moins aussi probable que B". Ainsi, au lieu de donner une approximation de la probabilité de chaque événement, l'expert présente une série de relations entre les probabilités de certaines paires d'événements.

Dans ce travail de recherche, nous tentons d'élaborer des méthodes algorithmiques pour résoudre des instances de grande taille du *problème de la satisfiabilité probabiliste qualitative* (noté dans la suite du texte par *QPSAT*). L'outil principal utilisé est la technique de génération de colonnes, jointe à la programmation non-linéaire en variables 0-1. Une discussion sur les méthodes de résolution envisageables

pour des *QPSAT* incluant des probabilités conditionnelles (événements conditionnants distincts ou non) est également présentée.

La suite de ce chapitre est consacrée à la définition du problème illustré à l'aide d'un exemple (section 1.2) et la description de la méthodologie employée dans ce projet de recherche (section 1.3).

1.2 Définition du problème et exemple

Le problème étudié dans ce projet de recherche est une extension du *problème de la satisfiabilité probabiliste*. Le cadre général du problème consiste à considérer un ensemble d'événements (appelés aussi propositions logiques ou phrases logiques) et leur probabilité d'être vrai. Contrairement au cas du *problème de la satisfiabilité probabiliste*, les probabilités associées à la véracité de chacun des événements ne sont pas connues, l'expert n'étant pas en mesure de nous donner une estimation de ces probabilités. Il peut, néanmoins, nous fournir une série d'intervalles plus ou moins précis sur leur valeur. Il peut également nous transmettre une série de relations d'inégalité stricte ou non stricte entre les probabilités de certaines paires d'événements. Il peut, par exemple, mentionner que la faillite d'une firme donnée est plus probable que celle d'une autre (relation d'inégalité stricte). Il peut également préciser que la baisse du prix d'un produit est au moins aussi probable que celle d'un autre (relation d'inégalité non stricte). Une première question que l'on peut se poser est la suivante : les jugements de probabilité donnés par l'expert sont-ils cohérents ? Cette première question conduit au premier problème que nous tenterons de résoudre dans ce travail de recherche, c'est-à-dire, le *problème de la cohérence des jugements de probabilité*. Ce problème découle du *problème des conditions d'expérience possible* étudié par Boole au milieu du 19^e siècle et décrit à la section 2.1. Par ailleurs, supposons que l'on ajoute un événement supplémentaire, on pourrait alors se poser la question suivante : quelle relation existe-t-il entre sa probabilité d'être vrai et la probabilité associée à la véracité de chacun des autres événements ? On pourrait également se demander quelles sont les bornes inférieure et supérieure les meilleures possibles sur la probabilité d'être vrai de cet événement supplémentaire tout en respectant les relations et les intervalles donnés par l'expert sur les autres probabilités. Ces deux questions conduisent au deuxième problème étudié dans le cadre de ce projet de recherche, soit le *problème de l'extension*.

Pour rendre l'exposé plus clair, prenons un petit exemple nous permettant d'illustrer le cadre général de *QPSAT*.

Exemple 1. Considérons une petite industrie où l'on retrouve uniquement deux firmes (la firme 1 est plus importante que la firme 2). Nous avons certaines informations concernant des événements pouvant se produire au cours de la prochaine année. La probabilité associée à la réalisation de chacun de ces événements est soit totalement inconnue ou comprise à l'intérieur d'un intervalle donné par un expert. L'expert nous transmet également une série de relations entre les probabilités de certaines paires d'événements. Formellement, ce problème s'énonce comme suit. On considère un ensemble d'événements élémentaires et on associe, à chacun, une variable booléenne prenant la valeur 1 si l'événement est vrai et 0 sinon.

Voici la liste des événements élémentaires :

$x_1 \equiv$ La firme 1 déclare faillite.

$x_2 \equiv$ La firme 2 déclare faillite.

$x_3 \equiv$ Arrivée d'un nouveau produit de l'extérieur.

$x_4 \equiv$ Récession.

À partir de ces événements élémentaires, on peut définir un ensemble de propositions logiques (ou événements) avec les opérateurs booléens suivants : \vee (somme logique), \wedge (produit logique) et \neg (négation). On peut également utiliser l'opérateur d'implication logique pour définir un événement du type : si A se produit alors B se produit (noté par $A \rightarrow B$). Ce type d'événement est équivalent à $\overline{A} \vee B$ puisque la seule possibilité qui n'est pas acceptée est que A se produise et que B ne se produise pas. Voici l'ensemble des propositions logiques :

$$S_1 \equiv x_1$$

$$S_2 \equiv x_2$$

$$S_3 \equiv x_3$$

$$S_4 \equiv x_4$$

$$S_5 \equiv x_4 \rightarrow x_1 \equiv \overline{x_4} \vee x_1$$

$$S_6 \equiv x_4 \rightarrow x_2 \equiv \overline{x_4} \vee x_2$$

$$S_7 \equiv x_1 \wedge x_3$$

$$S_8 \equiv x_2 \wedge x_3$$

On a également des relations, fournies par l'expert, entre certaines probabilités :

$$\text{prob}(S_1) < \text{prob}(S_2)$$

$$\text{prob}(S_3) \leq \text{prob}(S_4)$$

$$\text{prob}(S_5) \leq \text{prob}(S_6)$$

$$\text{prob}(S_7) < \text{prob}(S_8)$$

L'expert fournit également des intervalles sur la probabilité de certaines propositions.

Voici les intervalles associés à la probabilité des propositions S_1, S_2, S_3, S_4 :

$$0.1 \leq \text{prob}(S_1) \leq 0.2$$

$$0.1 \leq \text{prob}(S_2) \leq 0.3$$

$$0.2 \leq \text{prob}(S_3) \leq 0.5$$

$$0.1 \leq \text{prob}(S_4) \leq 0.4$$

Le premier problème consiste à déterminer si les jugements de probabilité donnés par l'expert sont cohérents. Supposons qu'on considère un événement supplémentaire (par exemple, $S_9 \equiv x_1 \wedge x_2$), on pourrait vouloir analyser la probabilité d'être vrai de S_9 , c'est-à-dire résoudre le *problème de l'extension*. La façon de modéliser et résoudre ces deux problèmes sera exposée au chapitre 3.

1.3 Méthodologie

Ce travail de recherche vise à étudier le problème *QPSAT* et d'implanter des algorithmes pour le résoudre efficacement. Ce projet de recherche se divise donc en quatre parties : 1) compréhension du *problème de la satisfiabilité probabiliste*, de ses extensions et des méthodes pour le résoudre ; 2) étude de différents problèmes découlant de *QPSAT*, élaboration et programmation d'algorithmes pour les résoudre ; 3) génération de problèmes pour étudier le comportement des algorithmes implantés ; 4) analyse des résultats.

Avant d'étudier *QPSAT*, il est primordial de bien comprendre le *problème de la satisfiabilité probabiliste* qui en est à l'origine. Le chapitre 2 sera donc consacré entièrement à l'étude de *PSAT* et de certaines extensions (probabilités conditionnelles et intervalles de probabilités). On y présente un bref historique de *PSAT* en citant les principaux auteurs ayant étudié ce problème depuis Boole [7, 8, 9, 10, 11]. En ce qui concerne l'étude de la formulation de ce problème ainsi que les méthodes algorithmiques pour le résoudre, l'accent sera mis sur les travaux de Hansen et Jaumard (voir [45] pour une revue récente de ces travaux) puisqu'ils sont à l'origine de la méthode utilisée, dans ce projet de recherche, pour résoudre *QPSAT*. Cette méthode est basée essentiellement sur la technique de génération de colonnes décrite au chapitre 2.

Le chapitre 3 vise à exposer en profondeur *QPSAT*. On présente notamment, pour chacun des problèmes à résoudre, sa formulation mathématique sous forme de programme linéaire et la méthode utilisée pour le résoudre. Les techniques de résolution sont basées sur la technique de génération de colonnes et font appel à des méthodes telles que l'analyse paramétrique de la programmation linéaire et la fermeture transitive d'un graphe. Le chapitre 4 présente des résultats numériques sur la performance des algorithmes implantés dans le cadre de ce travail. On y présente donc une brève description de la méthode de génération des problèmes tests, des tableaux illustrant les résultats numériques obtenus ainsi que les commentaires se rattachant à ces résultats. Finalement, le chapitre 5 conclut le présent travail.

CHAPITRE 2

LE PROBLÈME DE LA SATISFIABILITÉ PROBABILISTE

2.1 Introduction

Le *problème de la satisfiabilité probabiliste (PSAT)* doit son origine à Georges Boole [7, 8, 9, 10, 11] qui l'étudia au milieu du 19^e siècle. Le premier problème étudié par Boole, *le problème des conditions d'expérience possible*, peut s'énoncer comme suit : étant donné un ensemble de propositions logiques et leur probabilité d'être vraie, déterminer si ces probabilités sont cohérentes. Boole étudia également une extension de ce problème qui consiste à considérer une proposition logique supplémentaire et à déterminer les bornes inférieure et supérieure les meilleures possibles sur sa probabilité d'être vraie. Il appela ce problème, le *problème général de la théorie des probabilités*. Boole avait élaboré des méthodes algébriques pour résoudre ces deux problèmes (voir Hansen et Jaumard [45] pour une description de ce type de méthodes).

À l'exception des critiques formulées par Wilbraham [60] et, bien plus tard, par Keynes [50], les travaux de Boole ne reçurent pas beaucoup d'attention dans les pays anglo-saxons et ce, pendant plus d'un siècle. Ces travaux eurent néanmoins une influence notable sur de Finetti [25, 26, 27, 28], par l'intermédiaire de Medolaghi [55] dans le développement de sa théorie des probabilités subjectives. Hailperin, à partir du milieu du 20^e siècle [39, 40, 41], a consacré plusieurs livres et articles à *PSAT*. Cet auteur, en plus de proposer certaines extensions de la satisfiabilité probabiliste, a

montré comment exprimer *PSAT* sous forme de programme linéaire. Il a ainsi réussi à faire renaître un intérêt considérable pour les travaux de Boole dans les pays de langue anglaise.

L'expression de *PSAT* sous forme de programme linéaire a conduit à l'élaboration d'algorithmes permettant de résoudre des problèmes *SAT* de grande taille (voir Kavvadias et Papadimitriou [49], Jaumard, Hansen et Poggi de Aragão [48] et Douay-Nguetse [29]). L'outil principal utilisé est la technique de génération de colonnes, jointe à la programmation non-linéaire en variables 0–1. Les travaux réalisés dans le cadre de ce projet de recherche reposent essentiellement sur cette méthode de résolution. Elle sera donc expliquée en profondeur à la section 2.3.

À l'heure actuelle, *PSAT* fait l'objet de plusieurs recherches théoriques et appliquées. Cet engouement pour *PSAT* est dû, en grande partie, à Nilsson qui, dans un article de 1986 [56], souligna l'importance de *PSAT*, qu'il avait redécouvert indépendamment de la littérature déjà importante qui lui était consacrée, dans un traitement rigoureux de l'incertitude en intelligence artificielle.

Hansen et Jaumard [45] présentent une revue de littérature complète et récente sur *PSAT*. Des modèles et algorithmes pour la satisfiabilité probabiliste et ses extensions sont présentés. On y expose les méthodes de résolution analytique de *PSAT*. Elles comprennent l'approche algébrique de Boole et l'énumération des sommets et des rayons extrêmes du polyèdre dual du problème de la satisfiabilité probabiliste. Les méthodes de résolution numérique sont également étudiées dans ce texte. Ces méthodes sont basées sur la technique de génération de colonnes de la programmation linéaire et sur la programmation non-linéaire en variables 0–1 pour résoudre le problème auxiliaire du choix de la colonne entrante. Finalement, des applications et des considérations sur le potentiel de l'approche de la satisfiabilité probabiliste sont présentées.

La suite de ce chapitre est consacrée à l'étude approfondie du *problème de la satisfiabilité probabiliste*. La section 2.2 expose la façon de formuler *PSAT* en un programme linéaire. Par la suite, à la section 2.3, on discute des méthodes de résolution de *PSAT*. Les deux sections subséquentes sont consacrées à l'étude de

deux extensions importantes de *PSAT* : le cas avec intervalles sur les probabilités à la section 2.4 et le cas des probabilités conditionnelles à la section 2.5. Finalement, on conclut ce chapitre en discutant brièvement de résultats de calculs obtenus sur *PSAT*.

2.2 Formulation mathématique de *PSAT*

La formulation mathématique d'un problème peut varier d'un auteur à l'autre. Cette section est consacrée à l'étude de deux formulations : celle de Hailperin [39, 40, 41] et celle de Coletti [17, 19, 23, 24]. La première est présentée parce qu'elle est à la base de la formulation utilisée dans ce projet de recherche et la seconde car la plupart des travaux importants sur *QPSAT* est due à Coletti.

2.2.1 Formulation selon Hailperin

Formellement, *PSAT* peut s'énoncer ainsi. Soit un ensemble de m propositions logiques S_1, S_2, \dots, S_m définies sur n variables logiques x_1, x_2, \dots, x_n avec les opérateurs booléens usuels \vee (somme logique), \wedge (produit logique) et \neg (négation). On considère également les probabilités $\pi_1, \pi_2, \dots, \pi_m$ associées à la véracité des m propositions logiques ou événements. Sous forme de problème de décision, le problème consiste à vérifier si ces probabilités sont cohérentes ou non.

Comme il avait été montré par Hailperin [39, 40, 41], ce problème peut être formulé sous forme de programme linéaire. On note, tout d'abord, qu'il y a 2^n produits complets w_j de variables logiques x_1, x_2, \dots, x_n sous forme directe ou complétementée (x_i ou \bar{x}_i). Ces produits complets correspondent aux affectations des valeurs vraie ou fausse aux n variables logiques. On a 2^n produits complets w_j puisque chaque variable peut prendre la valeur vraie ou bien la valeur fausse et qu'on retrouve les n variables dans chaque produit complet. Pour la suite de ce rapport, on appellera ces produits complets de variables : les *mondes possibles*. Nilsson utilise une terminologie différente car il définit les mondes possibles à partir de l'affectation de valeurs vraie ou fausse sur les propositions logiques. Selon lui, un monde w_j est possible s'il existe une affectation sur l'ensemble des variables logiques permettant d'obtenir l'affectation de w_j sur les propositions et le monde est impossible sinon. Cette différence de définition sera expliquée plus amplement à la prochaine section où l'on décrit la formulation de Coletti. Le problème de décision de *PSAT* peut donc s'énoncer ainsi : existe-t-il une distribution de probabilités p_1, p_2, \dots, p_{2^n} sur l'ensemble des mondes possibles

telle que, pour $i = 1, 2, \dots, m$, la somme des probabilités des mondes possibles pour lesquels la proposition S_i est vraie soit égale à la probabilité π_i associée à la véracité de la proposition. La somme des probabilités sur l'ensemble des mondes possibles est égale à 1.

On peut maintenant formuler le problème sous forme de programme linéaire. Soit $A = (a_{ij})$, une matrice $m \times 2^n$ où

$$a_{ij} = \begin{cases} 1 & \text{si } S_i \text{ est vraie dans le monde possible } w_j \\ 0 & \text{sinon} \end{cases}$$

Sous forme de problème de décision, *PSAT* devient :

$$\begin{aligned} \mathbf{1}p &= 1 \\ Ap &= \pi \\ p &\geq 0 \end{aligned} \tag{2.1}$$

où p et π sont les vecteurs colonnes $(p_1, p_2, \dots, p_{2^n})^T$ et $(\pi_1, \pi_2, \dots, \pi_m)^T$ respectivement et $\mathbf{1}$ est un vecteur ligne de longueur 2^n dont chaque élément est égal à 1.

Les probabilités sont cohérentes si le système (2.1) admet une solution, c'est-à-dire s'il existe un vecteur p respectant (2.1). Il est important de souligner que les colonnes de A ne sont pas nécessairement toutes différentes.

Revenons à l'**exemple 1** pour montrer comment formuler *PSAT* sous forme de programme linéaire. En omettant les intervalles de probabilités et les relations entre les probabilités de certaines paires d'événements. La probabilité de chaque événement est fixée à π_i pour $i = 1, 2, \dots, m$.

On définit d'abord les 16 mondes possibles à partir des 4 variables logiques x_1, x_2, x_3, x_4 et on leur associe une variable p_j correspondant à leur probabilité d'être vrai :

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}	p_{14}	p_{15}	p_{16}
x_1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
x_2	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
x_3	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
x_4	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

Le problème revient à vérifier si le système suivant admet une solution :

$$\begin{aligned}
p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} + p_{13} + p_{14} + p_{15} + p_{16} &= 1 \\
p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 &= \pi_1 \\
p_1 + p_2 + p_3 + p_4 &+ p_9 + p_{10} + p_{11} + p_{12} &= \pi_2 \\
p_1 + p_2 &+ p_5 + p_6 &+ p_9 + p_{10} &+ p_{13} + p_{14} &= \pi_3 \\
p_1 &+ p_3 &+ p_5 &+ p_7 &+ p_9 &+ p_{11} &+ p_{13} &+ p_{15} &= \pi_4 \\
p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 &+ p_{10} &+ p_{12} &+ p_{14} &+ p_{16} &= \pi_5 \\
p_1 + p_2 + p_3 + p_4 &+ p_6 &+ p_8 + p_9 + p_{10} + p_{11} + p_{12} &+ p_{14} &+ p_{16} &= \pi_6 \\
p_1 + p_2 &+ p_5 + p_6 &= \pi_7 \\
p_1 + p_2 &+ p_9 + p_{10} &= \pi_8 \\
p_1, p_2, p_3, \dots, p_{16} &\geq 0
\end{aligned}$$

Le problème d'optimisation, nommé *le problème de l'extension* dans le cadre de ce rapport, consiste à déterminer les bornes inférieure et supérieure les meilleures possibles sur la probabilité d'être vraie d'une proposition logique S_{m+1} additionnelle tout en respectant les contraintes associées au système (2.1).

Le problème d'optimisation sous forme de programme linéaire devient :

$$\begin{aligned}
&\min / \max && A_{m+1}p \\
&\text{sujet à :} && \\
&&& \mathbb{1} p = 1 \\
&&& A p = \pi \\
&&& p \geq 0
\end{aligned} \tag{2.2}$$

2.2.2 Formulation selon Coletti

Coletti, seule ou avec d'autres auteurs, a publié plusieurs articles [17, 19, 23, 24] sur *PSAT* sans probabilités qualitatives. L'essentiel des propos qui seront tenus dans cette section et à la section 2.5.2 provient de Coletti et Scozzafava [24] dans lequel est présenté un survol des travaux de Coletti sur *PSAT* avec ou sans probabilités conditionnelles. Notons tout d'abord que la façon de traiter *PSAT*, par Coletti, est basée sur la théorie des probabilités proposée par de Finetti [25, 26, 27, 28]. Dans le cadre du problème de la cohérence, Coletti, à l'instar de Hailperin, considère une évaluation de probabilités $\pi_1, \pi_2, \dots, \pi_m$ sur un ensemble d'événements S_1, S_2, \dots, S_m . Cependant, au lieu de définir les mondes possibles, comme Hailperin, à partir des variables logiques définissant les événements ou propositions, Coletti définit les mondes possibles à partir des propositions logiques. Notons que dans les deux cas, les mondes possibles correspondent à des affectations de valeurs (vrai ou faux) aux phrases logiques ou aux variables logiques selon le cas. En général, sa façon de procéder est beaucoup moins efficace. Elle nécessite de résoudre 2^m problèmes SAT pour vérifier si certains des mondes définis sur les propositions logiques engendrent une contradiction (mondes impossibles selon Nilsson). Les mondes obtenus après le retrait des mondes impossibles sont appelés atomes par Coletti. Prenons un exemple pour illustrer les deux manières de définir les mondes.

Exemple 2. Soit,

$$\begin{aligned} S_1 &\equiv x_1 \\ S_2 &\equiv \bar{x}_1 \vee x_2 \\ S_3 &\equiv x_2 \end{aligned}$$

Mondes possibles sur les variables (selon Hailperin) :

	w_1	w_2	w_3	w_4
x_1	1	1	0	0
x_2	1	0	1	0

Mondes possibles (ou impossibles) sur les propositions logiques (selon Coletti) :

	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8
S_1	1	1	1	1	0	0	0	0
S_2	1	1	0	0	1	1	0	0
S_3	1	0	1	0	1	0	1	0

On voit, en observant les 8 mondes définis sur les propositions, que les mondes m_2, m_3, m_7, m_8 sont impossibles car ils engendrent des contradictions. En effet, si la proposition S_1 est fautive, alors S_2 est nécessairement vraie puisque si $x_1 = 0$ alors $\bar{x}_1 = 1$. Ainsi, les mondes m_7 et m_8 sont impossibles. Par ailleurs, si S_3 est vraie, alors S_2 est vraie car $S_3 \subseteq S_2$. Par conséquent, le monde m_3 est impossible car S_3 est vraie et S_2 est fautive dans ce monde. En ce qui concerne le monde m_2 , il est impossible car si S_1 est vraie et que S_3 est fautive, alors S_2 est nécessairement fautive ($S_2 \equiv \bar{S}_1 \vee S_3$). En somme, pour trouver les atomes (mondes possibles), il a fallu vérifier la cohérence de 8 (2^3) mondes.

Une fois les atomes générés, on considère la probabilité p'_j associée à la véracité de l'atome w'_j . Soit W' l'ensemble des atomes. Considérons une matrice A' de dimension $m \times |W'|$ où chaque élément est défini ainsi :

$$a'_{ij} = \begin{cases} 1 & \text{si l'événement } S_i \text{ est vrai dans l'atome } w'_j \\ 0 & \text{sinon} \end{cases}$$

Le problème de la cohérence consiste donc à vérifier si le système d'équations linéaires suivant admet une solution :

$$\begin{aligned} \mathbf{1}p' &= 1 \\ A'p' &= \pi \\ p' &\geq 0 \end{aligned} \tag{2.3}$$

En d'autres termes, tester la cohérence est équivalent à trouver une distribution de probabilités sur les atomes de manière à respecter la probabilité donnée à chaque événement. On voit que le problème (2.3) est identique au problème (2.1) à l'exception de la façon dont les mondes sont générés.

Coletti définit également le problème de la cohérence en termes de système de paris (appelé “betting scheme” par de Finetti). Le problème de la cohérence peut se concevoir de la manière suivante : si quelqu’un parie un montant π_i sur un événement S_i et reçoit un montant de 1 si S_i est vrai et 0 sinon, alors les probabilités π_i pour $i = 1, 2, \dots, m$ sont cohérentes s’il est impossible d’effectuer de tels paris de manière à perdre ou gagner assurément. Plus formellement, on définit le gain aléatoire

$$G = \sum_{i=1}^m \lambda_i (s_i - \pi_i)$$

où $s_i = 1$ si S_i est vrai et 0 sinon. G peut être interprété comme le gain correspondant à la combinaison de m paris de montants $\pi_1 \lambda_1, \pi_2 \lambda_2, \dots, \pi_m \lambda_m$ sur les événements S_1, S_2, \dots, S_m avec des sommes $\lambda_1, \lambda_2, \dots, \lambda_m$ arbitraires. Les probabilités $\pi_1, \pi_2, \dots, \pi_m$ sont cohérentes si pour tout choix de $\lambda_1, \lambda_2, \dots, \lambda_m \in \mathbb{R}$,

$$\sup_{w'_j} G \geq 0 \quad (\inf \leq 0).$$

Coletti étudie également le *problème de l’extension*. Dans le cadre de ce problème, il faut trouver les bornes inférieure et supérieure les meilleures possibles sur la probabilité d’être vrai d’un événement S_{m+1} en plus. Pour ce faire, on résout un programme linéaire obtenu en ajoutant au problème (2.3), l’objectif suivant : min/max π_{m+1} . Bref, on effectue le même genre d’ajout que lors du passage de (2.1) à (2.2) dans la formulation d’Hailperin. Notons que Coletti impose une restriction supplémentaire sur S_{m+1} . Cet événement se doit d’être défini comme l’union de certains des atomes définis sur les m événements déjà présents, c’est-à-dire que S_{m+1} est dépendant logiquement des événements S_1, S_2, \dots, S_m .

2.3 Résolution de *PSAT*

Pour résoudre *PSAT*, plusieurs méthodes sont envisageables. Tout d'abord, on peut utiliser des méthodes de type analytique comme la méthode algébrique développée par Boole [7, 8, 9, 10, 11] et l'énumération des rayons extrêmes du dual (Jaumard, Hansen et Poggi de Aragão [47]). Pour une description complète des méthodes analytiques, voir Hansen et Jaumard [45]. On peut également utiliser une méthode numérique basée sur la génération de colonnes (Kavvadias et Papadimitriou [49], Jaumard, Hansen et Poggi de Aragão [48] et Douanya Nguetse [29]). La suite de cette section décrit la méthode de génération de colonnes appliquée à *PSAT*.

2.3.1 Méthode de génération de colonnes

Les programmes linéaires (2.1) et (2.2), décrivant *PSAT* sous forme de décision et d'optimisation respectivement, ont un nombre exponentiel de colonnes qui augmente avec le minimum entre le nombre n de variables logiques et le nombre m de propositions. Par conséquent, des *PSAT* de grande taille ne peuvent pas être résolus, ni même introduits en mémoire, si on énumère explicitement l'ensemble des colonnes associées aux variables p_j . Une méthode efficace consiste à utiliser la technique de génération de colonnes. Cette méthode exacte est basée sur le prolongement de la méthode révisée du simplexe dans laquelle un petit nombre de colonnes sont présentes de manière explicite et où l'on détermine la colonne entrante par la résolution d'un problème auxiliaire (voir Gilmore et Gomory [36] et Chvátal [14] pour une description plus détaillée de cette méthode).

L'utilisation de cet algorithme pour des applications de *PSAT* à la fiabilité des réseaux a été proposée par Zemel [61]. Plus tard, Georgakopoulos *et al* [35] ont montré comment se servir de la génération de colonnes pour le cas général de *PSAT*. Ces travaux ont été prolongés par Jaumard, Hansen et Poggi de Aragão [48].

2.3.2 Expression du problème auxiliaire

Dans le cas qui nous intéresse, le problème auxiliaire associé au choix de la colonne entrante de coût réduit minimal (maximal) consiste à minimiser (maximiser) l'équation suivante :

$$c_j - uA^j = a_{m+1} - u_0 - \sum_{i=1}^m u_i a_{ij} \quad (2.4)$$

où u_0 et u_i sont les variables duales associées à la tautologie S_0 et aux propositions logiques S_1, \dots, S_m et u est le vecteur ligne (u_0, u_1, \dots, u_m) . Si pour chaque proposition logique, on attribue les valeurs 1 et 0 selon qu'elle soit vraie ou fausse, on peut réécrire (2.4) ainsi :

$$S_{m+1} - u_0 - \sum_{i=1}^m u_i S_i \quad (2.5)$$

Cette équation peut être reformulée en une expression arithmétique faisant intervenir les variables logiques x_1, \dots, x_n apparaissant dans les propositions logiques S_i pour $i = 1, 2, \dots, m$. On associe également les valeurs 1 et 0 aux variables logiques selon que la variable est vraie ou fausse. La reformulation est réalisée à l'aide des opérateurs usuels de l'algèbre booléenne (\vee , \wedge et $\bar{}$) en utilisant les formules suivantes :

$$\begin{aligned} x_i \vee x_j &\equiv x_i + x_j - x_i \times x_j \\ x_i \wedge x_j &\equiv x_i \times x_j \\ \bar{x}_i &\equiv 1 - x_i. \end{aligned} \quad (2.6)$$

Suite à ces modifications, le problème auxiliaire devient un problème de minimisation (maximisation) d'une fonction non-linéaire en variables 0–1 ou fonction pseudo-booléenne (Hammer et Rudeanu [42]).

Revenons, encore une fois, à l'**exemple 1** et voyons comment on exprime le problème auxiliaire en un problème de minimisation ou maximisation d'une fonction non-linéaire en variables 0–1. Le problème auxiliaire du modèle (2.1) selon la formulation (2.5) s'énonce comme suit :

$$\max \quad u_0 + u_1 S_1 + u_2 S_2 + \dots + u_8 S_8$$

On peut reformuler ce problème en une expression arithmétique faisant intervenir les variables logiques tel que décrit ci-dessus. Le problème devient :

$$\begin{aligned} \max \quad & u_0 + u_1x_1 + u_2x_2 + u_3x_3 + u_4x_4 + u_5(\bar{x}_4 + x_1 - \bar{x}_4x_1) + \\ & u_6(\bar{x}_4 + x_2 - \bar{x}_4x_2) + u_7x_1x_3 + u_8x_2x_3 \\ \text{où } & x_1, x_2, x_3, x_4 \in \{0, 1\} \text{ et } \bar{x}_4 = 1 - x_4. \end{aligned}$$

En remplaçant \bar{x}_4 par $(1 - x_4)$, on obtient le problème suivant :

$$\begin{aligned} \max \quad & u_0 + u_1x_1 + u_2x_2 + u_3x_3 + u_4x_4 + u_5 - u_5x_4 + u_5x_4x_1 + u_6 - u_6x_4 + \\ & u_6x_4x_2 + u_7x_1x_3 + u_8x_2x_3 \\ \text{où } & x_1, x_2, x_3, x_4 \in \{0, 1\}. \end{aligned}$$

On a donc, en définitive, un problème de maximisation d'une fonction non-linéaire en variables 0-1.

2.3.3 Résolution du problème auxiliaire

Puisque le problème (2.5) doit être résolu à chaque itération de l'algorithme et qu'il s'agit d'un problème NP-difficile, le résoudre exactement peut être très coûteux en termes de temps de calcul. En fait, il n'est pas nécessaire de le résoudre exactement à chaque fois. On peut se contenter d'une méthode heuristique (voir section suivante) tant qu'une colonne de coût réduit de bon signe est trouvée. On utilise une méthode exacte (voir section 2.3.5) pour vérifier si une colonne de bon signe existe encore lorsque la méthode heuristique n'en a pas trouvée.

2.3.4 Méthode heuristique

Pour résoudre le problème auxiliaire, on peut utiliser des méthodes heuristiques du type recuit simulé (Dowland [30], Hansen et Jaumard [43]), recherche avec tabous (Glover [37, 38], Hansen et Jaumard [43]) ou recherche de profondeur variable

(Kavvadias et Papadimitriou [49], Lin et Kernighan [53]). La méthode retenue pour ce projet est la recherche avec tabous. Nous allons donc, dans la suite de cette section, exposer les rudiments de cet algorithme heuristique (pour une description complète des autres méthodes, voir Hansen et Jaumard [45]).

La recherche avec tabous est due à Glover [37, 38]. Certaines de ses idées de base furent proposées indépendamment par Hansen et Jaumard [43] sous le nom de *Steepest Ascent Mildest Descent (SAMD)*. Cette méthode consiste à exploiter les informations du gradient tout en procurant une façon de sortir des minima locaux. Une version simple de cette procédure, pour un problème de maximisation, consiste à suivre la direction de plus grande montée jusqu'à atteindre un optimum local. Par la suite, on emprunte la direction de plus petite descente pour sortir de l'optimum local (d'où le nom *Steepest Ascent Mildest Descent*). On empêche l'algorithme de cycler par l'entremise d'une interdiction de marche arrière pour un nombre d'itérations donné. En ce qui nous concerne, la méthode consiste à trouver une solution de départ pour le problème de maximisation (minimisation) d'une fonction non-linéaire en variables 0–1. On choisit au hasard un vecteur initial X_0 d'affectations de valeurs aux variables x_j , $j = 1, \dots, n$. Ensuite, pour un nombre donné d'itérations sans changement, on trouve, parmi les variables sans interdiction, celle qui procure la meilleure augmentation (diminution) de la valeur de la fonction ou la moins grande diminution (augmentation) si on ne peut pas monter (descendre). On met à jour les interdictions sur les variables et on vérifie si la solution obtenue par la modification de la valeur de la variable choisie est meilleure que la meilleure solution trouvée jusqu'à présent. Si oui, on modifie la valeur de la meilleure solution trouvée et on fixe le nombre d'itérations sans changement à 0. Si la solution trouvée est moins bonne que la précédente, on interdit de modifier la variable (imposition d'un tabou) pour un nombre d'itérations donné pour empêcher l'algorithme de cycler. À la fin de la résolution, on conserve la meilleure solution obtenue durant tout le déroulement de l'algorithme. On espère évidemment que cette solution donne une colonne de coût réduit de bon signe. Sinon, on se doit de résoudre le problème avec une méthode exacte afin de savoir si une colonne de coût réduit de bon signe existe encore. La figure 1 donne une description de la méthode de recherche avec tabous pour le problème de maximisation d'une fonction non-linéaire en variables 0–1 telle qu'implantée dans le cadre de ce travail de recherche.

Trouver une solution initiale X_0 ; (assignation de valeurs aux variables x_1, x_2, \dots, x_n de manière aléatoire).

$$f_{opt} = f(X_0);$$

$$X_{opt} = X_0;$$

$t_j = 0$ pour $j = 1, \dots, n$; (aucune interdiction sur les variables).

$$\text{Nombre-iterations} = 0;$$

Tant que $\text{Nombre-iterations} < \text{limite}$

Choisir une variable x_i non tabouée telle que sa dérivée soit maximale;

$X_k =$ nouveau vecteur solution; (modification de la valeur de x_i).

$$\delta_k = f(X_k) - f(X_0);$$

Si $f(X_k) > f_{opt}$ alors

$$f_{opt} = f(X_k);$$

$$X_{opt} = X_k;$$

$$\text{Nombre-iterations} = 0;$$

Sinon

$$\text{Nombre-iterations} = \text{Nombre-iterations} + 1;$$

Si $\delta_k < 0$ alors $t_k = \ell$;

$t_j = t_j - 1$ pour $t_j > 0, j = 1, 2, \dots, n$;

$$X_0 = X_k;$$

Fin tant que.

À la fin de l'algorithme, X_{opt} est la meilleure solution trouvée durant tout le déroulement de l'algorithme et f_{opt} est la valeur de cette solution. f_{opt} constitue donc une approximation de la valeur de la solution optimale de la fonction.

X_0, X_k, X_{opt} correspondent à des vecteurs d'assignations de valeurs aux variables x_1, x_2, \dots, x_n . Les paramètres *limite* et ℓ correspondent, respectivement, à la limite du nombre d'itérations sans changement et le nombre d'itérations pour lequel une variable est fixée (tabou).

FIG. 2.1 – Schéma de l'algorithme de recherche avec tabous pour la maximisation d'une fonction non-linéaire en variables 0-1.

2.3.5 Méthode exacte

Plusieurs méthodes sont envisageables pour résoudre de manière exacte le problème auxiliaire. Parmi celles-ci, on note les méthodes de linéarisation, les méthodes algébriques, les méthodes d'évaluation et séparation progressive et les méthodes de coupes (voir Hansen, Jaumard et Mathon [44] pour une revue de la littérature récente sur le sujet). La suite de cette section est consacrée à la description de la méthode retenue dans le cadre de ce travail de recherche, c'est-à-dire la linéarisation suggérée par Fortet [32, 33]. La méthode consiste à remplacer chaque produit de variables par une nouvelle variable et à ajouter des contraintes pour faire en sorte que la nouvelle variable prenne la valeur correspondant au produit de variables.

Considérons un terme :

$$c \prod_{j \in J} x_j \quad (2.7)$$

où $c \in \mathbb{R}$ et $x_j \in \{0, 1\}$ pour $j \in J$. En remplaçant le produit de variables par la nouvelle variable y et en ajoutant les contraintes forçant y à prendre la valeur requise, le terme (2.7) devient :

$$\begin{aligned} & cy \\ \text{sujet à :} \\ & y \geq \sum_{j \in J} x_j - |J| + 1, \\ & y \leq x_j \quad j \in J, \\ & y \geq 0 \end{aligned} \quad (2.8)$$

La première contrainte force la nouvelle variable y à prendre la valeur 1 si toutes les variables x_j pour $j \in J$ sont égales à 1. Les deux autres contraintes forcent y à prendre la valeur 0 dans le cas où au moins une variable x_j pour $j \in J$ est égale à 0. Notons qu'il n'est pas nécessaire d'indiquer explicitement que y est une variable 0–1 puisque les variables x_j sont des variables 0–1. De plus, pour un problème de maximisation (minimisation), on peut omettre la première contrainte si $c > 0$ ($c < 0$) car, à l'optimum, y prendra la valeur la plus grande possible. Si $c < 0$ ($c > 0$), le deuxième type de contraintes est inutile car, à l'optimum, y prendra la valeur la plus petite possible. Ainsi, dans les deux cas, y prendra la valeur désirée à l'optimum.

Exemple 3. Regardons un petit exemple de l'utilisation de la méthode de linéarisation pour transformer le problème de minimisation d'une fonction non-linéaire en variables 0–1 en un programme linéaire, 0–1.

Soit le problème suivant :

$$\begin{aligned} \max \quad & 2x_1x_2 - 3x_1x_3 \\ \text{où} \quad & x_1, x_2, x_3 \in \{0, 1\} \end{aligned}$$

En posant $y_1 = x_1x_2$ et $y_2 = x_1x_3$, on obtient le programme linéaire suivant :

$$\begin{aligned} \max \quad & 2y_1 - 3y_2 \\ \text{sujet à :} \quad & \\ & y_1 \leq x_1 \\ & y_1 \leq x_2 \\ & y_2 \geq x_1 + x_3 - 1 \\ & y_1, y_2 \geq 0 \\ & x_1, x_2, x_3 \in \{0, 1\} \end{aligned}$$

Notons que le problème peut être assez gros si on a de nombreux produits de variables et de nombreuses variables x_j . Puisqu'il est en nombres entiers, il peut être relativement long à résoudre. Voilà pourquoi il est justifié de ne pas le résoudre exactement à chaque itération.

2.4 *PSAT* avec intervalles

L'utilisation d'intervalles au lieu d'estimations ponctuelles pour les probabilités est souvent plus réaliste et plus générale que le problème classique développé par Boole. Il se peut, en effet, que l'expert soit plus facilement en mesure de nous transmettre un intervalle sur la valeur d'une probabilité donnée plutôt qu'une estimation ponctuelle de celle-ci. Avec des intervalles sur les probabilités π_i , le problème (2.2) devient :

$$\begin{aligned}
 & \min / \max && A_{m+1}p \\
 & \text{sujet à :} && \\
 & && \mathbb{1}p = 1 \\
 & && \underline{\pi} \leq Ap \leq \bar{\pi} \\
 & && p \geq 0.
 \end{aligned} \tag{2.9}$$

On peut utiliser des variables bornées pour identifier les probabilités des événements. Soit $\pi = (\pi_1, \pi_2, \dots, \pi_m)^T$, le vecteur colonne des variables représentant les probabilités des événements. Le problème devient :

$$\begin{aligned}
 & \min / \max && A_{m+1}p \\
 & \text{sujet à :} && \\
 & && \mathbb{1}p = 1 \\
 & && Ap - \pi = 0 \\
 & && p \geq 0 \\
 & && \underline{\pi} \leq \pi \leq \bar{\pi}
 \end{aligned} \tag{2.10}$$

Cette extension de *PSAT* fut étudiée par Hailperin [39] et, plus tard, par Lad, Dickey et Rahman [52], Jaumard, Hansen et Poggi de Aragão [48], Andersen et Hooker [2].

2.5 *PSAT* avec probabilités conditionnelles

Il arrive parfois que les experts ne soient en mesure de donner une estimation seulement lorsqu'une condition est satisfaite (par exemple, un diagnostic en présence d'un symptôme). D'où l'intérêt d'étendre le modèle *PSAT* pour tenir compte des probabilités conditionnelles. Ces dernières peuvent être présentes dans les contraintes du problème ou bien dans l'objectif. Plusieurs auteurs ont étudié la façon de tenir compte des probabilités conditionnelles dans *PSAT* (*CONDSAT*). Nous exposerons, à la section 2.5.1, la manière proposée par Hansen, Jaumard et Poggi de Aragão [48] et, à la section 2.5.2, celle de Coletti [17, 19, 23, 24].

2.5.1 Formulation selon Hansen, Jaumard et Poggi de Aragão

Supposons, qu'en plus du cadre général de *PSAT*, nous avons une série de probabilités du type $\text{prob}(S_k|S_\ell)$. En introduisant une variable π_ℓ pour la probabilité $\text{prob}(S_\ell)$ inconnue, on obtient les deux contraintes suivantes :

$$\begin{aligned} A_{k \wedge \ell} p - \pi_{k|\ell} \pi_\ell &= 0 \\ A_\ell p - \pi_\ell &= 0 \end{aligned} \tag{2.11}$$

où $A_{k \wedge \ell} = (a_{k \wedge \ell, j})$ avec $a_{k \wedge \ell, j} = 1$ si les propositions S_ℓ et S_k sont toutes deux vraies dans le monde possible w_j et 0 sinon. Ces deux contraintes permettent de modéliser la formule suivante : $\text{prob}(S_k|S_\ell) = \frac{\text{prob}(S_k \wedge S_\ell)}{\text{prob}(S_\ell)} = \pi_{k|\ell}$.

Si les $\pi_{k|\ell}$ sont connues et si l'objectif n'inclut pas de probabilités conditionnelles, alors *CONDSAT* demeure linéaire après l'ajout de ce type de contraintes pour chaque $\pi_{k|\ell}$. Cela est vrai autant dans le cas où les autres probabilités π_i sont connues (modèle (2.2)) que dans le cas où elles sont comprises à l'intérieur d'un intervalle donné (modèle (2.10)). Si certaines probabilités conditionnelles $\pi_{k|\ell}$ sont comprises à l'intérieur d'un intervalle, *CONDSAT* reste linéaire si pour ces probabilités, les

contraintes ajoutées sont les suivantes :

$$\begin{aligned} A_{k \wedge \ell} p - \underline{\pi}_{k|\ell} A_{\ell} p &\geq 0 \\ A_{k \wedge \ell} p - \overline{\pi}_{k|\ell} A_{\ell} p &\leq 0 \end{aligned} \quad (2.12)$$

Si l'objectif est une probabilité conditionnelle et le dénominateur π_j est inconnu, alors *CONDSAT* devient un programme hyperbolique, c'est-à-dire l'optimisation d'un rapport de deux fonctions linéaires sous contraintes linéaires :

$$\begin{aligned} \min / \max & \quad \frac{A_{k \wedge \ell} p}{A_{\ell} p} \\ \text{sujet à :} & \quad \mathbb{1} p = 1 \\ & \quad A p = \pi \\ & \quad p \geq 0 \end{aligned} \quad (2.13)$$

Hailperin [41] et Chesnokov [13] ont montré qu'il était possible d'utiliser un résultat de Charnes et Cooper [12] pour transformer le problème (2.13) en un programme linéaire. Jaumard, Hansen et Poggi de Aragão [48] ont utilisé cette méthode conjointement avec la méthode de génération de colonnes pour résoudre le problème (2.13).

2.5.2 Formulation selon Coletti

En ce qui concerne la façon de traiter le cas conditionnel pour Coletti [17, 19, 23, 24], notons tout d'abord qu'elle considère séparément les événements conditionnants des événements dont on ne connaît la probabilité qu'en présence d'un autre événement. On a donc une évaluation de probabilités π_i , pour $i = 1, 2, \dots, m$, qui correspond à la probabilité qu'un événement E_i soit vrai sachant que H_i est vrai (prob $(E_i|H_i)$). On considère les $2m$ événements suivants : $E_1, E_2, \dots, E_m, H_1, H_2, \dots, H_m$. Soit W' l'ensemble des atomes w'_j , cet ensemble est obtenu en déterminant l'ensemble des 2^{2m} mondes définis selon les $2m$ événements en éliminant les mondes impossibles selon Nilsson ainsi que les mondes où tous les H_i sont faux. Puisqu'on génère les mondes à la fois sur les E_i et sur les H_i , on obtient encore plus de mondes que dans le cas non conditionnel (section 2.2.2). Il faut donc résoudre davantage de problèmes *SAT* pour déterminer les mondes engendrant des contradictions. Par conséquent,

l'inefficacité quant à la façon de définir les mondes par Coletti est d'autant plus importante dans le cas conditionnel.

Comme pour le cas non conditionnel, la cohérence des probabilités π_i peut être exprimée en termes de paris. Supposons que pour parier sur la réalisation de $E_i|H_i$ quelqu'un paie un montant π_i . Si H_i est vrai, il reçoit un montant de 1 si E_i est vrai et 0 sinon, par contre si H_i est faux alors il reçoit un montant de π_i (le pari est annulé). Les jugements de probabilité sont cohérents s'il est impossible de perdre ou gagner assurément en effectuant des paris de la sorte. Plus formellement, on définit le gain aléatoire

$$G = \sum_{i=1}^m \lambda_i h_i (e_i - \pi_i)$$

où h_i et e_i prennent les valeurs 1 ou 0 si l'événement correspondant est vrai ou faux respectivement. G peut être interprété comme le gain correspondant à la combinaison de m paris de montants $\pi_1 \lambda_1, \pi_2 \lambda_2 \dots, \pi_m \lambda_m$ sur les événements $E_1|H_1, E_2|H_2, \dots, E_m|H_m$ avec des sommes $\lambda_1, \lambda_2, \dots, \lambda_m$ arbitraires. Les probabilités $\pi_1, \pi_2, \dots, \pi_m$ sont cohérentes si pour tout choix de $\lambda_1, \lambda_2, \dots, \lambda_m \in \mathbb{R}$,

$$\sup_{w'_j} G \geq 0 \quad (\text{inf} \leq 0)$$

Sous forme de programme linéaire, soit p' le vecteur des probabilités associées à la véracité de chacun des atomes, le test de la cohérence est le suivant :

$$\begin{aligned} A'_{E_i \wedge H_i} p' - \pi_i A'_{H_i} p' &= 0 & \text{pour } i = 1, 2 \dots, m \\ \mathbf{1} p' &= 1 \\ p' &\geq 0 \end{aligned} \tag{2.14}$$

où $A'_{E_i \wedge H_i} = (a'_{E_i \wedge H_i, j})$ avec $a'_{E_i \wedge H_i, j} = 1$ si les événements E_i et H_i sont tous deux vrais dans le monde possible (atome) w'_j et 0 sinon. De même, $A'_{H_i} = (a'_{H_i, j})$ avec $a'_{H_i, j} = 1$ si H_i est vrai dans le monde possible (atome) w'_j et 0 sinon. Le premier type de contrainte permet de modéliser la définition d'une probabilité conditionnelle : $\text{prob}(E_i|H_i) = \frac{\text{prob}(E_i \wedge H_i)}{\text{prob}(H_i)}$. Cependant la vérification que (2.14) admet une solution ne garantit pas que le problème soit cohérent selon Coletti. En effet, il se peut que la probabilité de certains événements conditionnants (H_i) soit nulle. Regardons un exemple où le programme linéaire (2.14) admet une solution avec la probabilité d'un événement conditionnant égale à 0.

Exemple 4. Soit,

$$\text{prob}(E|H) = \text{prob}(\overline{E}|H) = 1/3.$$

Ce problème n'est pas cohérent selon Coletti car la seule solution possible consiste à fixer $\text{prob}(H)$ à 0 et $\text{prob}(E|H) + \text{prob}(\overline{E}|H) = 2/3 \neq 1$, ce que Coletti exclut, même dans ce cas.

Contrairement à nous, les chercheurs de l'école de de Finetti soutiennent que dans un cas semblable, le problème est incohérent. Ainsi, pour que le problème soit cohérent, il faudrait soit ajouter la contrainte suivante :

$$A'_{H_i} p' > 0 \quad \text{pour } i = 1, 2, \dots, m \quad (2.15)$$

soit vérifier que la cohérence existe dans le monde restreint où un ou plusieurs H_i ont une probabilité nulle. Coletti et Scozzafava [24] présentent un algorithme pour résoudre (2.14) tout en respectant (2.15). Il s'agit d'une méthode itérative qui consiste à tenter, en premier lieu, de résoudre (2.14). Si le système n'admet pas de solution, alors les jugements de probabilité sont incohérents et on arrête l'algorithme. Sinon, on vérifie (2.15). Si cette condition est respectée, alors les jugements de probabilité sont cohérents et on arrête le déroulement de l'algorithme. Sinon, on refait les étapes précédentes en prenant un sous-ensemble des atomes jusqu'à ce que (2.15) soit respectée ou que (2.14) n'admette pas de solution. Le sous-ensemble d'atomes considéré à chaque itération correspond aux atomes à l'intérieur desquels au moins un H_i , de probabilité nulle dans la solution, est vrai. Coletti et Scozzafava [24] présentent également l'application de cet algorithme pour la résolution de très petits exemples. Néanmoins, aucune expérience de calculs sur la performance de cet algorithme pour de grands problèmes n'est présentée.

2.6 Expérience de calculs sur *PSAT*

En appliquant l'algorithme de génération de colonnes et la recherche avec tabous, Jaumard, Hansen et Poggi de Aragão [48] ont réussi à résoudre des *PSAT* contenant jusqu'à 140 variables et 300 propositions logiques pour le cas de décision et d'optimisation. La partie programmation linéaire est réalisée par le code MPSX de Marsten [54]. Des problèmes de taille similaire avec probabilités conditionnelles ont été résolus. Plus récemment, en utilisant le code de CPLEX pour la programmation linéaire et la linéarisation pour le problème auxiliaire, Douanya Nguetse, Hansen et Jaumard [29] ont résolu des *PSAT* ayant jusqu'à 500 propositions.

En ce qui concerne les travaux de Coletti, ces derniers ne présentent pas de résultats de tests numériques. Il s'agit évidemment d'une grande lacune puisqu'il est difficile de juger de l'efficacité de sa façon de traiter le problème. Mais, d'après les commentaires des sections 2.2.2 et 2.5.2, tout nous porte à croire que sa manière de traiter le problème est moins efficace que celle préconisée dans les divers travaux de Hansen et Jaumard sur *PSAT*. De plus, elle ne présente aucun algorithme pour résoudre de grandes instances de *PSAT* et *CONDSAT*. Dans ces cas, bien que le problème soit linéaire, il n'en demeure pas moins que lorsque le nombre d'événements devient assez gros, la taille du programme linéaire devient dramatiquement énorme. Il s'avère donc indispensable d'avoir recours à des méthodes très poussées de la recherche opérationnelle, telle que la génération de colonnes, pour résoudre efficacement ces problèmes.

CHAPITRE 3

LE PROBLÈME DE LA SATISFIABILITÉ PROBABILISTE QUALITATIVE

3.1 Introduction

Le problème étudié dans ce projet de recherche et auquel est consacré ce troisième chapitre est le *problème de la satisfiabilité probabiliste qualitative* (QPSAT). Ce problème est une extension de *PSAT* étudié au chapitre précédent. Le cadre général de *QPSAT* consiste à considérer, comme dans *PSAT*, un ensemble d'événements ou de propositions logiques définis sur un ensemble d'événements élémentaires représentés par des variables logiques. La probabilité associée à la véracité d'un événement est soit totalement inconnue ou comprise à l'intérieur d'un intervalle donné par l'expert (voir section 2.4). De plus, on considère des relations d'inégalité stricte ou non stricte entre les probabilités de certaines paires d'événements (probabilités qualitatives). Ces relations sont du type : "A est au plus aussi probable que B" ($\text{prob}(A) \leq \text{prob}(B)$) ou "A est moins probable que B" ($\text{prob}(A) < \text{prob}(B)$). Le premier problème consiste à savoir si les jugements de probabilité donnés par l'expert sont cohérents. Ce problème sera traité aux sections 3.3 et 3.4 pour les cas sans ou avec relations d'inégalité stricte. En second lieu, on peut être intéressé, étant donné un événement supplémentaire, à déterminer les bornes inférieure et supérieure les meilleures possibles sur sa probabilité d'être vrai. De plus, on peut vouloir trouver, avec le moins de calculs possibles, la relation qui existe entre la probabilité de l'événement supplémentaire et celle de chacun des autres événements. La formulation de ce problème ainsi que la manière

de le résoudre sont présentées aux sections 3.5 et 3.6. Avant d'étudier notre façon de définir et résoudre $QPSAT$, nous allons présenter à la section 3.2, un bref survol des principaux travaux réalisés jusqu'à maintenant sur $QPSAT$. Les sections 3.3 à 3.6 sont consacrées à l'étude de $QPSAT$ dans le cas non conditionnel. Finalement, à la section 3.7, on discute brièvement de la formulation retenue pour le problème $QPSAT$ avec probabilités conditionnelles. Bien que ce problème ne soit pas résolu explicitement dans le cadre de ce projet de recherche, on discute d'une approche envisageable pour le résoudre.

3.2 Survol des travaux antérieurs sur *QPSAT*

Cette section a pour but de présenter un bref survol des principaux travaux antérieurs dont l’objet d’étude est *QPSAT*.

Au début des années 1970, T.L. Fine [31] consacrait le second chapitre de son volume, *Theories of Probability*, à l’étude axiomatique des probabilités comparatives (qualitatives). Cependant, contrairement à ce qui est fait dans le présent projet de recherche, T.L. Fine inclut la comparabilité parmi les hypothèses de son modèle. Cette hypothèse implique que la probabilité de tout évènement est comparable directement avec celle de tout autre. On a donc une relation entre chaque paire de probabilités. Il s’agit d’une hypothèse très forte rendant impossible l’utilisation directe des résultats obtenus par cet auteur pour le présent projet.

Depuis 1988, Giulianella Coletti s’intéresse à l’étude de *QPSAT*. Elle a publié, seule ou avec d’autres auteurs, plusieurs articles ([15] à [22] et [58]) discutant des probabilités qualitatives selon l’approche de de Finetti (voir aussi Coletti [19] pour un survol de ses travaux sur *PSAT* avec ou sans probabilités qualitatives). Contrairement à T.L. Fine, elle n’inclut pas la comparabilité parmi les hypothèses de ses modèles. Dans ses articles, Coletti mentionne les définitions, les théorèmes et les modèles de *QPSAT*. Cependant, elle ne discute pas vraiment d’une méthode de résolution envisageable lorsque la taille du problème augmente. Ce dernier point est à l’étude dans le cadre du présent travail de recherche.

Une étude de *QPSAT*, dans le cas non conditionnel, est présentée dans Coletti [16]. On y introduit les différents concepts de la cohérence pour le cas de probabilités qualitatives ainsi que les définitions et les théorèmes s’y rapportant. Cette étude de la cohérence est essentiellement basée sur le concept de système de paris tel que présenté aux sections 2.2.2 et 2.5.2. Dans le cadre de *QPSAT*, Coletti considère un ensemble d’évènements S_1, S_2, \dots, S_m et un ensemble de relations (probabilités qualitatives) entre les probabilités de certaines paires d’évènements. Une probabilité qualitative, telle que définie par Coletti, est une relation binaire du type $S_1 \preceq S_2$ exprimant l’idée que “ S_1 est au plus aussi probable que S_2 ”. On peut avoir

également une relation du type $S_1 \sim S_2$ qui indique que “ S_1 et S_2 sont équivalents en termes de probabilités” ou bien une relation du type $S_1 \prec S_2$ exprimant l’idée que “ S_1 est moins probable que S_2 ”. On génère tout d’abord l’ensemble W' des atomes w'_j sur les m événements en utilisant la technique décrite à la section 2.2.2. Soit $Q = \{(i, j) \mid S_i \preceq S_j\}$, en termes de paris, les probabilités qualitatives sont cohérentes si pour tout choix de $\lambda_{ij} \geq 0 \quad \forall (i, j) \in Q$,

$$\sup_{w'_j} \sum_{(i,j) \in Q} \lambda_{ij} (s_i - s_j) \geq 0$$

où $s_i = 1$ si S_i est vrai et 0 sinon. Cette condition peut s’interpréter de la façon suivante : un pari consiste en un échange entre un bookmaker qui mise sur un événement S_i et un joueur qui mise sur S_j . Ce pari procure un même montant λ_{ij} au bookmaker si S_i est vrai et au joueur si S_j est vrai. Les probabilités qualitatives sont cohérentes s’il est impossible que le joueur perde assurément pour une combinaison de paris tels que, pour chaque paire d’événements dans Q , l’événement sur lequel mise le joueur est au moins aussi probable que celui sur lequel mise le bookmaker. Une condition similaire est obtenue si on considère des relations d’inégalité stricte entre un ensemble de paires d’événements. Dans ce cas, il faut que le supremum prenne une valeur strictement positive.

En ce qui concerne la formulation mathématique de *QPSAT* sans probabilité conditionnelle, Coletti [21] indique qu’il s’agit d’un programme linéaire sans toutefois divulguer son modèle. De plus, sa façon de traiter les inégalités strictes n’est pas décrite explicitement. On peut cependant croire qu’elle utilise une approche similaire à la nôtre en observant son modèle pour le cas conditionnel (voir Coletti [19]). En conclusion, notons que son modèle correspond très certainement au prolongement du modèle (2.3) permettant de résoudre le problème de la cohérence sans probabilité qualitative. Comme il a été mentionné à la section 2.2.2, sa façon de définir les mondes est moins efficace que celle préconisée dans ce projet de recherche. Voilà pourquoi on utilisera une approche différente pour résoudre *QPSAT*.

Le cas avec probabilités qualitatives conditionnelles, étudié par Coletti [15, 17, 18, 19] ainsi que Coletti, Gilio et Scozzafava [20, 21, 22], donne des modèles généralement plus complexes. On considère l’ensemble W' des atomes w'_j générés sur l’ensemble des événements (voir section 2.5.2 pour une description de cette méthode).

Soit p' le vecteur des probabilités associées à la véracité de chacun des atomes, les contraintes permettant de modéliser les relations entre certaines probabilités conditionnelles, selon Coletti [21], sont les suivantes :

$$\begin{aligned}
[A'_{E_i \wedge H_i} p'] [A'_{H_j} p'] - [A'_{E_j \wedge H_j} p'] [A'_{H_i} p'] &\geq 0 && \text{si } E_i | H_i \succeq E_j | H_j \\
[A'_{E_i \wedge H_i} p'] [A'_{H_j} p'] - [A'_{E_j \wedge H_j} p'] [A'_{H_i} p'] &> 0 && \text{si } E_i | H_i \succ E_j | H_j \\
p' &\geq 0 && \\
A'_{H_i} p' &> 0 && \\
A'_{H_j} p' &> 0 &&
\end{aligned} \tag{3.1}$$

où $A'_{E_i \wedge H_i} = (a'_{E_i \wedge H_i, j})$ avec $a'_{E_i \wedge H_i, j} = 1$ si les événements E_i et H_i sont tous deux vrais dans le monde possible (atome) w'_j et 0 sinon. De même, $A'_{H_i} = (a'_{H_i, j})$ avec $a'_{H_i, j} = 1$ si H_i est vrai dans le monde possible (atome) w'_j et 0 sinon. Notons que dans le cas où les événements conditionnants sont les mêmes (par exemple $E_i | H_i \succeq E_j | H_i$), les modèles demeurent linéaires puisque les termes $[A'_{H_j} p']$ et $[A'_{H_i} p']$ sont égaux et peuvent donc être enlevés de (3.1). Cependant, lorsque les événements conditionnants sont distincts, on obtient des programmes avec des termes quadratiques. Il est important de souligner, qu'en utilisant de telles contraintes, on se retrouve avec une quantité énorme de termes quadratiques. Il s'agit de problèmes très difficiles à résoudre, voir même impossible en pratique sans l'utilisation de techniques telles que la génération de colonnes. Il est possible de reformuler (3.1) de manière à obtenir un programme non-linéaire avec beaucoup moins de termes quadratiques. Cette formulation est décrite à la section (3.7). Dans cette section, on illustre également la différence entre la formulation de Coletti et celle retenue dans le cadre de ce projet à l'aide d'un exemple.

La cohérence des probabilités qualitatives conditionnelles peut être analysée également selon le concept de système de paris. Soit $Q = \{(i, j) \mid E_i | H_i \succeq E_j | H_j\}$, en termes de paris, les probabilités qualitatives sont cohérentes si, pour tout $(i, j) \in Q$, $\lambda_{ij}, \beta_{ij} \geq 0$ et $\lambda_{ij} + \beta_{ij} > 0$:

$$\sup_{w'_j} \sum_{(i, j) \in Q} (\lambda_{ij}(e_i h_i - e_j h_j) + \beta_{ij}(h_j - h_i)) \geq 0$$

où $e_i = 1$ si E_i est vrai et 0 sinon (e_j, h_i et h_j sont définis de la même façon). Cette condition peut s'interpréter de la manière suivante : un pari consiste en un échange entre un bookmaker qui mise sur $E_i | H_i$ et un joueur qui mise sur $E_j | H_j$. Ce pari

procure le même montant λ_{ij} au bookmaker si E_i et H_i se réalisent et au joueur si E_j et H_j se réalisent. Ce pari procure également un montant β_{ij} (éventuellement différent de λ_{ij}) au bookmaker si H_j se produit et au joueur si H_i se produit. Les probabilités qualitatives conditionnelles sont cohérentes s'il est impossible que le joueur perde assurément pour une combinaison de paris telle que, pour chaque paire d'événements dans Q , l'événement conditionnel sur lequel le joueur mise est au moins aussi probable que celui sur lequel le bookmaker mise. En ce qui concerne les inégalités strictes, il faut que le supremum prenne une valeur strictement positive, c'est-à-dire qu'il soit impossible que le joueur obtienne assurément un gain négatif ou nul.

3.3 Cohérence sans inégalité stricte

3.3.1 Définition du problème

Le premier problème que nous allons étudier est celui de la cohérence des jugements de probabilité pour le cas qualitatif sans relation d'inégalité stricte. Formellement, dans le cadre de ce problème, on considère, m événements ou propositions logiques S_1, S_2, \dots, S_m définis sur un ensemble de n variables logiques x_1, x_2, \dots, x_n avec les opérateurs booléens usuels \vee (somme logique), \wedge (produit logique) et \neg (négation). On considère également les probabilités $\pi_1, \pi_2, \dots, \pi_m$ associées à la véracité des m propositions logiques ou événements. Ces probabilités ne sont pas fixées, c'est-à-dire qu'elles peuvent être totalement inconnues ou se situer à l'intérieur d'un intervalle donné par l'expert. On a, en plus, r_1 relations d'inégalité non stricte entre les probabilités de certaines paires d'événements (par exemple, $\pi_1 \leq \pi_2$).

Le problème consiste à savoir si les jugements de probabilité donnés par l'expert sont cohérents. Formellement, il s'agit d'un problème de décision qui consiste à répondre à la question suivante : existe-t-il une distribution de probabilités sur tous les mondes possibles telle que les r_1 relations soient respectées ? Rappelons qu'un monde possible, selon la définition retenue dans le cadre de ce projet de recherche, correspond à un produit complet de variables logiques, c'est-à-dire un produit où chaque variable apparaît sous forme directe ou complémentée. Les événements peuvent alors s'écrire comme une somme logique de mondes possibles. Puisque les mondes possibles sont deux à deux disjoints, la probabilité associée à la véracité d'une proposition peut s'écrire comme la somme des probabilités des mondes possibles où cette proposition est vraie. Notons également que lorsque nous sommes en présence d'intervalles sur les probabilités des événements, ces derniers se doivent d'être respectés par la solution trouvée.

3.3.2 Formulation mathématique

Soit Q_1 , l'ensemble des paires d'indices des événements telles qu'il existe une relation d'inégalité non stricte entre la probabilité des deux événements du type “ S_k est au plus aussi probable que S_l ”. Plus formellement, on définit cet ensemble de la manière suivante : $Q_1 = \{(k, l) \mid \pi_k \leq \pi_l\}$. La cardinalité de l'ensemble Q_1 est égale à r_1 . Le problème peut alors être formulé sous forme de programme linéaire.

Soit $A = (a_{ij})$, une matrice $m \times 2^n$ où

$$a_{ij} = \begin{cases} 1 & \text{si } S_i \text{ est vrai dans le monde possible } w_j \\ 0 & \text{sinon} \end{cases}$$

Le problème de la cohérence des jugements de probabilité sous forme de programme linéaire devient :

$$\begin{aligned} \mathbf{1}p &= 1 \\ Ap - \pi &= 0 \\ \pi_k - \pi_l &\leq 0 \quad \forall (k, l) \in Q_1 \\ p &\geq 0 \\ \underline{\pi} &\leq \pi \leq \bar{\pi} \end{aligned} \tag{3.2}$$

où p et π sont les vecteurs colonnes $(p_1, p_2, \dots, p_{2^n})^T$ et $(\pi_1, \pi_2, \dots, \pi_m)^T$ respectivement et $\mathbf{1}$ est un vecteur ligne de longueur 2^n dont chaque élément est égal à 1. Les vecteurs p et π contiennent les variables du modèle. Les jugements de probabilité sont cohérents si le système (3.2) admet une solution.

Revenons à l'**exemple 1** pour montrer comment formuler le problème de la cohérence sous forme de programme linéaire. On a le même problème que dans l'application de l'exemple 1 à la section 2.2.1 sauf qu'on considère en plus les 2 relations d'inégalité non stricte ainsi que les intervalles sur les probabilités des événements S_1, S_2, S_3, S_4 tels que présentés à la section 1.2. Notons tout d'abord qu'on a les 16 mondes possibles tels que définis à la section 2.2.1. Les probabilités π_i ne sont pas fixées et donc, les π_i , pour $i = 1, 2, \dots, m$, représentent des variables dans le modèle. De plus, l'ensemble Q_1 est défini ainsi : $Q_1 = \{(3, 4), (5, 6)\}$. Il suffit donc d'ajouter les contraintes suivantes au modèle obtenu lors de la formulation du modèle (2.1)

pour l'exemple 1 (section 2.2.1) :

$$\begin{aligned}\pi_3 &\leq \pi_4 \\ \pi_5 &\leq \pi_6\end{aligned}$$

On ajoute également les bornes suivantes sur certaines variables π_i pour faire en sorte que la solution respecte les intervalles donnés par l'expert :

$$\begin{aligned}0.1 &\leq \pi_1 \leq 0.2 \\ 0.1 &\leq \pi_2 \leq 0.3 \\ 0.2 &\leq \pi_3 \leq 0.5 \\ 0.1 &\leq \pi_4 \leq 0.4\end{aligned}$$

3.3.3 Méthode de résolution

Pour vérifier que le système (3.2) admet une solution, il suffit d'appliquer une phase 1 de l'algorithme révisé du simplexe en variables bornées pour trouver une solution de base admissible au programme linéaire. Cette solution correspond, en fait, à un sommet extrême du polytope représentant l'ensemble des solutions admissibles. Cependant, étant donné que le nombre de variables p_j croît exponentiellement avec la taille du problème (2^n mondes possibles), il est impensable de résoudre le problème en considérant l'ensemble des colonnes lorsque la taille du problème est grande. Voilà pourquoi, une approche de résolution par la méthode de génération de colonnes est utilisée (voir la section 2.3 pour une description de cette méthode pour résoudre des *PSAT*). Dans le cas qui nous intéresse, les colonnes explicites sont les colonnes associées aux variables d'écart, aux variables artificielles ainsi qu'aux variables π_i . Les colonnes implicites, c'est-à-dire les colonnes que l'on génère au fur et à mesure du déroulement de l'algorithme, sont les colonnes associées aux variables p_j . Pour formuler le problème (3.2) avec les variables d'écart et artificielles, on ajoute, au membre de gauche de l'équation, une variable non-négative pour chaque contrainte d'égalité (variable artificielle) et une variable non-négative pour chaque contrainte d'inégalité (variable d'écart). L'objectif, lors de la phase 1 de l'algorithme, consiste à minimiser la somme des variables artificielles. Si à l'optimum, on trouve un objectif

égal à 0, alors le problème est cohérent et on a trouvé une solution de base réalisable au problème. Dans le cas où la valeur de l'objectif, à l'optimum, est positive, alors le problème n'est pas cohérent. En effet, on ne peut trouver dans ce cas une solution respectant les contraintes de (3.2). On se retrouve, en fait, avec au moins une variable artificielle de valeur positive à l'optimum. Cette situation reflète l'impossibilité de respecter l'ensemble des contraintes d'égalité.

En ce qui concerne la formulation du problème auxiliaire, il faut noter que, puisque les p_j n'apparaissent pas dans l'objectif lors de la phase 1, leur coût est nul ($c_j = 0$) et ainsi, le problème auxiliaire consiste à maximiser la deuxième partie de l'équation (2.4). Si on attribue, à chaque événement S_i , la valeur 1 ou 0 advenant le cas où il est vrai ou faux, le problème auxiliaire est le suivant :

$$\max u_0 + \sum_{i=1}^m u_i S_i \quad (3.3)$$

où u_0 et u_i sont les variables duales associées à la tautologie S_0 et aux propositions logiques S_1, \dots, S_m .

Comme il a été mentionné à la section 2.3.2, on peut reformuler cette fonction en termes des variables logiques x_1, x_2, \dots, x_n en utilisant les formules (2.6). On obtient alors un problème de maximisation d'une fonction non-linéaire en variables 0–1 (voir l'application de l'exemple 1 à la section 2.3.2). Tel que mentionné à la section 2.3.3, il n'est pas nécessaire de résoudre exactement ce problème à chaque itération, on peut se contenter d'une méthode heuristique tant qu'une colonne de coût réduit négatif ($c_j - uA^j < 0$) est trouvée. Dans le cadre de ce projet de recherche, l'algorithme heuristique utilisé est la recherche avec tabous (voir section 2.3.4 pour une description de cet algorithme). On utilise ensuite une méthode exacte pour vérifier si une colonne de coût réduit négatif existe encore lorsque la méthode heuristique n'en a pas trouvée. L'algorithme exact utilisé dans ce projet de recherche consiste à linéariser la fonction non-linéaire en variables 0–1 en utilisant la linéarisation de Fortet [32, 33] (voir la section 2.3.5 pour une description de cette technique de linéarisation). Par la suite, on résout le programme linéaire mixte 0–1 obtenu en utilisant le module de CPLEX pour la résolution de programmes linéaires mixtes.

3.4 Cohérence avec inégalités strictes

3.4.1 Définition du problème

Le problème étudié dans cette section est sensiblement le même que celui de la section 3.3 sauf qu'on considère en plus r_2 relations d'inégalité stricte entre les probabilités de certaines paires d'événements. On a donc, $r = r_1 + r_2$ relations d'inégalité non stricte (par exemple, $\pi_1 \leq \pi_2$) ou stricte (par exemple, $\pi_1 < \pi_2$).

Le problème consiste à répondre à la question suivante : existe-t-il une distribution de probabilités sur tous les mondes possibles telle que les r relations d'inégalité stricte ou non stricte soient respectées ? Dans le cas où nous détenons des intervalles sur les probabilités des propositions, ces derniers se doivent d'être respectés par la distribution de probabilité sur les mondes possibles trouvée.

3.4.2 Formulation mathématique

En observant le modèle (3.2), une façon de modéliser ce problème nous vient naturellement à l'esprit. Il suffit de prendre le modèle (3.2) et d'y ajouter les contraintes relatives aux relations d'inégalité stricte. Par la suite, on vérifie si le système résultant admet une solution respectant l'ensemble des équations. Soit Q_2 , l'ensemble des paires d'indices des propositions telles qu'il existe une relation d'inégalité stricte entre la probabilité des deux propositions du type " S_k est moins probable que S_l ". Plus formellement, on définit cet ensemble de la manière suivante : $Q_2 = \{(k, l) \mid \pi_k < \pi_l\}$. On remarque que la somme des cardinalités des ensembles Q_1 (défini à la section 3.3.2) et Q_2 est égal à r , le nombre de relations.

Le modèle obtenu est le suivant :

$$\begin{aligned}
\mathbb{1}p &= 1 \\
Ap - \pi &= 0 \\
\pi_k - \pi_l &\leq 0 \quad \forall (k, l) \in Q_1 \\
\pi_k - \pi_l &< 0 \quad \forall (k, l) \in Q_2 \\
p &\geq 0 \\
\underline{\pi} &\leq \pi \leq \bar{\pi}
\end{aligned} \tag{3.4}$$

Pour tester la cohérence des jugements de probabilité, il suffit de vérifier si (3.4) admet une solution. Cependant, la présence d'inégalités strictes nous empêche d'utiliser directement l'algorithme du simplexe pour résoudre ce modèle. Il faut donc reformuler ce système afin d'éliminer les inégalités strictes. Pour ce faire, on ajoute une variable t dont la positivité stricte implique que les inégalités sont strictes. Par exemple, la relation $\pi_1 < \pi_2$ est équivalente à :

$$\begin{aligned}
\pi_1 + t &\leq \pi_2 \\
t &> 0
\end{aligned} \tag{3.5}$$

Notons qu'il n'est pas nécessaire d'ajouter une variable pour chaque relation d'inégalité stricte, une seule suffit pour l'ensemble des relations de ce type. Il suffit maintenant de savoir comment respecter le fait que t doit être strictement positive. Pour ce faire, il suffit de maximiser t . Si, à l'optimum, $t = 0$, alors le problème est incohérent car il est impossible de respecter les inégalités strictes. Ainsi, t représente, pour toutes les relations entre les paires d'événements de Q_2 , l'écart minimal entre les probabilités dans une inégalité stricte :

$$t = \min_{(k,l) \in Q_2} \{ \pi_l - \pi_k \} \tag{3.6}$$

Le problème de la cohérence des jugements de probabilité devient :

$$\begin{aligned}
&\max && t \\
&\text{sujet à :} && \\
&\mathbb{1}p &= & 1 \\
&Ap - \pi &= & 0 \\
&\pi_k - \pi_l &\leq & 0 \quad \forall (k, l) \in Q_1 \\
&\pi_k - \pi_l + t &\leq & 0 \quad \forall (k, l) \in Q_2 \\
&p &\geq & 0 \\
&\underline{\pi} &\leq & \pi \leq \bar{\pi}
\end{aligned} \tag{3.7}$$

Il n'existe pas de distribution sur l'ensemble des mondes possibles respectant les r relations si le programme linéaire n'a pas de solution réalisable (la phase 1 se termine avec une variable artificielle de valeur strictement positive à l'optimum) ou si la valeur optimale de t (obtenue par la phase 2) est nulle. Le dernier cas exprime l'impossibilité de respecter l'ensemble des relations d'inégalité stricte.

On peut maintenant illustrer la formulation de (3.7) sur l'**exemple 1**. Pour ce faire, en plus des ajouts réalisés à la section 2.3.2, il faut ajouter la variable t ainsi que les contraintes relatives aux relations d'inégalité stricte. On doit également incorporer un objectif consistant à maximiser t . Notons que l'on a, encore une fois, les 16 mondes possibles tels que définis à la section 2.2.1 et que l'ensemble Q_2 est défini ainsi : $Q_2 = \{(1, 2), (7, 8)\}$. Ainsi, le problème de la cohérence devient :

$$\begin{array}{ll}
\max & t \\
\text{sujet à :} & \\
& p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11} + p_{12} + p_{13} + p_{14} + p_{15} + p_{16} = 1 \\
& p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 = \pi_1 \\
& p_1 + p_2 + p_3 + p_4 + p_9 + p_{10} + p_{11} + p_{12} = \pi_2 \\
& p_1 + p_2 + p_5 + p_6 + p_9 + p_{10} + p_{13} + p_{14} = \pi_3 \\
& p_1 + p_3 + p_5 + p_7 + p_9 + p_{11} + p_{13} + p_{15} = \pi_4 \\
& p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_{10} + p_{12} + p_{14} + p_{16} = \pi_5 \\
& p_1 + p_2 + p_3 + p_4 + p_6 + p_8 + p_9 + p_{10} + p_{11} + p_{12} + p_{14} + p_{16} = \pi_6 \\
& p_1 + p_2 + p_5 + p_6 = \pi_7 \\
& p_1 + p_2 + p_9 + p_{10} = \pi_8 \\
& \pi_3 \leq \pi_4 \\
& \pi_5 \leq \pi_6 \\
& \pi_1 + t \leq \pi_2 \\
& \pi_7 + t \leq \pi_8 \\
& 0.1 \leq \pi_1 \leq 0.2 \\
& 0.1 \leq \pi_2 \leq 0.3 \\
& 0.2 \leq \pi_3 \leq 0.5 \\
& 0.1 \leq \pi_4 \leq 0.4 \\
& p_1, p_2, p_3, \dots, p_{16} \geq 0
\end{array}$$

3.4.3 Méthode de résolution

La manière de résoudre ce problème consiste à appliquer la méthode en deux phases de l'algorithme révisé du simplexe en variables bornées. La première phase consiste à voir si l'ensemble des solutions respectant les contraintes de (3.7) est non vide et dans l'affirmative, trouver une solution de base admissible. La seconde phase permet de trouver la valeur maximale de t tout en respectant les contraintes de (3.7). Pour résoudre efficacement ce problème, on applique donc l'algorithme de génération de colonnes tel que décrit à la section 2.3. Les colonnes associées aux variables d'écart, aux variables artificielles, aux variables π_i et t sont explicites tandis que les colonnes associées aux variables p_j sont implicites. En d'autres termes, seules les colonnes des variables p_j sont générées au fur à mesure du déroulement de l'algorithme et ce, par l'entremise de la résolution du problème auxiliaire. La première phase est donc identique au cas sans inégalité stricte (section 3.3.3) en ajoutant la colonne de t dans le problème maître. La deuxième phase consiste uniquement à modifier l'objectif du problème. Puisque t apparaît seul dans l'objectif et qu'il s'agit d'une colonne explicite, il suffit de modifier le sens de l'objectif pour le problème auxiliaire.

3.5 Extension sans inégalité stricte

Dans le cadre de ce problème, on a un événement S_{m+1} en plus dont il faut analyser la probabilité π_{m+1} d'être vrai. Deux cas sont alors possibles : 1) on peut vouloir déterminer les bornes inférieure et supérieure les meilleures possibles sur π_{m+1} (section 3.5.1) ; 2) on peut être intéressé à trouver, avec le moins de calculs possibles, la relation qui existe entre la probabilité π_{m+1} et chacune des autres probabilités π_i pour $i = 1, 2, \dots, m$ (section 3.5.2). Cet événement supplémentaire s'écrit à partir des événements élémentaires (représentés par les variables logiques) définissant les m autres événements. Rappelons qu'il n'y a pas de relations d'inégalité stricte ; le cas avec inégalités strictes sera traité à la section 3.6.

3.5.1 Bornes inférieure et supérieure sur π_{m+1}

Définition du problème

On a un événement S_{m+1} supplémentaire, on essaie de déterminer les bornes inférieure et supérieure les meilleures possibles sur sa probabilité π_{m+1} d'être vrai. Évidemment, ces bornes doivent respecter les relations entre les probabilités de certaines paires d'événements (probabilités qualitatives) ainsi que les intervalles sur les probabilités de certains événements.

Modèle mathématique

Étant donné qu'il faut déterminer les bornes inférieure et supérieure les meilleures possibles sur la probabilité π_{m+1} , il suffit d'ajouter l'objectif de minimisation et maximisation de π_{m+1} au modèle mathématique (3.2) associé au *problème de la cohérence des jugements de probabilité* sans inégalité stricte.

Sous forme de programme linéaire, le problème devient :

$$\begin{aligned}
 & \min / \max && A_{m+1} & p \\
 & \text{sujet à :} && & \\
 & && \mathbb{1}p & = 1 \\
 & Ap - \pi & = & 0 & \quad (3.8) \\
 & \pi_k - \pi_l & \leq & 0 & \quad \forall (k, l) \in Q_1 \\
 & && p & \geq 0 \\
 & \underline{\pi} \leq \pi \leq \bar{\pi} & & &
 \end{aligned}$$

Reprenons l'**exemple 1** en rappelant que l'événement supplémentaire, dont on veut analyser la probabilité, est le suivant : $S_9 \equiv x_1 \wedge x_2$. Il suffit alors d'ajouter l'objectif de minimisation et maximisation de π_9 . Puisque $S_9 \equiv x_1 \wedge x_2$, alors π_9 est égale à la somme des probabilités des mondes possibles où les événements élémentaires x_1 et x_2 sont vrais. Ainsi, il suffit de joindre, au modèle obtenu suite à l'ajout des contraintes et des bornes (voir section 3.3.2), l'objectif suivant : $\min/\max p_1 + p_2 + p_3 + p_4$.

Méthode de résolution

Pour résoudre ce problème, il suffit d'appliquer la méthode en deux phases de l'algorithme révisé du simplexe en variables bornées. La première phase consiste à résoudre le *problème de la cohérence des jugements de probabilité* afin de trouver une solution admissible respectant les contraintes de (3.8). La deuxième phase consiste à trouver la solution admissible permettant de minimiser et de maximiser la valeur de π_{m+1} . On résout ce problème également à l'aide de l'algorithme de génération de colonnes. Le problème auxiliaire pour la phase 1 est identique à celui du *problème de la cohérence des jugements de probabilité* (voir section 3.3.3). En ce qui concerne la phase 2, lors de la résolution du problème auxiliaire, on détermine la colonne entrante en minimisant (borne inférieure) ou maximisant (borne supérieure) la fonction (2.5). Cette fonction est transformée en fonction non-linéaire en variables 0–1 par la transformation décrite à la section 2.3.3. Lorsque la méthode heuristique et la méthode

exacte ne trouvent pas de colonne de coût réduit négatif (minimisation) ou positif (maximisation), alors on a trouvé la borne inférieure sur π_{m+1} (minimisation) ou la borne supérieure sur π_{m+1} (maximisation).

3.5.2 Relation entre la probabilité π_{m+1} et chacune des m probabilités π_i

Définition du problème

On a un événement S_{m+1} en plus, il faut trouver, pour chaque événement S_i , pour $i = 1, 2, \dots, m$, si ce dernier est :

1. plus probable que S_{m+1} ($\pi_i > \pi_{m+1}$),
2. au moins aussi probable que S_{m+1} ($\pi_i \geq \pi_{m+1}$),
3. moins probable que S_{m+1} ($\pi_i < \pi_{m+1}$),
4. au plus aussi probable que S_{m+1} ($\pi_i \leq \pi_{m+1}$),
5. incomparable, en termes de probabilités, avec S_{m+1} .

Méthode générale de résolution

La méthode générale de résolution permettant de résoudre ce problème comporte deux étapes :

1. Résoudre le *problème de la cohérence des jugements de probabilité*. On tente de trouver une solution réalisable au problème en résolvant le problème (3.2). Si le système admet une solution, on passe à l'étape 2. Sinon, on arrête.
2. Pour chaque événement S_i (pour $i = 1, 2, \dots, m$) :
 - (a) Comparer S_i avec l'objectif pour voir s'il y a inclusion. Par exemple, si $S_{m+1} \equiv x_1 \wedge x_2 \wedge x_3$ et $S_i \equiv x_1 \wedge x_2$, alors $S_{m+1} \subseteq S_i$ et donc $\pi_{m+1} \leq \pi_i$. S'il n'y a pas d'inclusion, on passe à (b).

- (b) Vérifier s'il existe une relation indirecte entre π_{m+1} et π_i . On résout, tout d'abord, le programme linéaire dont l'objectif consiste à trouver le minimum de $\pi_{m+1} - \pi_i$ sous les contraintes définies par (3.2). Le programme linéaire obtenu est le suivant :

$$\begin{aligned}
 \min \quad & A_{m+1}p - A_i p \\
 \text{sujet à :} \quad & \\
 & \mathbb{1}p = 1 \\
 & Ap - \pi = 0 \\
 & \pi_k - \pi_l \leq 0 \quad \forall (k, l) \in Q_1 \\
 & p \geq 0 \\
 & \underline{\pi} \leq \pi \leq \bar{\pi}
 \end{aligned} \tag{3.9}$$

La solution optimale de ce problème nous donne la valeur minimale de la différence entre π_{m+1} et π_i . Trois cas sont alors possibles :

$$\min(\pi_{m+1} - \pi_i) \begin{cases} < 0, & \text{si la relation } \pi_{m+1} \geq \pi_i \text{ n'est pas toujours vraie.} \\ = 0, & \text{si } \pi_{m+1} \geq \pi_i. \\ > 0, & \text{si } \pi_{m+1} > \pi_i. \end{cases}$$

Si la valeur minimale de $\pi_{m+1} - \pi_i$ est non-positive, alors on tente de résoudre le problème (3.9) en inversant le sens de l'objectif (maximisation au lieu de minimisation). Encore une fois, 3 cas sont possibles :

$$\max(\pi_{m+1} - \pi_i) \begin{cases} > 0, & \text{si la relation } \pi_{m+1} \leq \pi_i \text{ n'est pas toujours vraie.} \\ = 0, & \text{si } \pi_{m+1} \leq \pi_i. \\ < 0, & \text{si } \pi_{m+1} < \pi_i. \end{cases}$$

Si on obtient la première alternative dans les deux cas ($\min(\pi_{m+1} - \pi_i) < 0$ et $\max(\pi_{m+1} - \pi_i) > 0$), alors S_{m+1} et S_i sont incomparables en termes de probabilités. Si on obtient la deuxième alternative dans les deux cas ($\min(\pi_{m+1} - \pi_i) = \max(\pi_{m+1} - \pi_i) = 0$), alors $\pi_{m+1} = \pi_i$.

La réalisation de l'ensemble des comparaisons est coûteuse en termes de temps de calculs. En effet, il se peut que l'on soit obligé d'effectuer jusqu'à $2m$ appels de la phase 2 de l'algorithme du simplexe en génération de colonnes. En utilisant judicieusement l'information que nous détenons sur les relations entre les π_i ainsi que la conclusion trouvée suite à une comparaison avec l'objectif, il est possible de résoudre le problème beaucoup plus efficacement.

Amélioration de la méthode

Pour améliorer la méthode, il suffit de tenir compte des relations, impliquées ou fournies par l'expert, entre les probabilités de certaines paires d'événements. Les relations impliquées sont représentées par un graphe $G = (N, A)$ d'implication. Les sommets du graphe correspondent aux $m + 1$ événements. On trace un arc (i, j) si $S_i \subseteq S_j$ et donc $\pi_i \leq \pi_j$. On ajoute les arcs définis par les r_1 relations d'inégalité non stricte données par l'expert. Les composantes de G se définissent donc ainsi : $N = \{ 1, 2, \dots, m + 1 \}$ et $A = \{ (i, j) \mid \pi_i \leq \pi_j \}$.

Revenons, encore une fois, à l'**exemple 1** pour illustrer la manière de tracer le graphe d'implication. On a un sommet par événement, les numéros des sommets correspondent aux indices des événements. La figure 3.1 représente le graphe d'implication de l'exemple 1. Notons que les relations d'inégalité stricte y sont présentes. On a donc, tel qu'indiqué dans la légende de la figure 3.1, trois types d'arcs : 1) les arcs (i, j) représentant une relation du type $\pi_i \leq \pi_j$ qui est due à une inclusion, 2) les arcs (i, j) indiquant une relation du type $\pi_i \leq \pi_j$ donnée par l'expert et 3) les arcs (i, j) indiquant une relation du type $\pi_i < \pi_j$ fournie par l'expert. Le dernier type d'arcs permet de modéliser le graphe d'implication pour le problème de la section 3.6.2. Analysons maintenant brièvement le graphe de la figure 3.1 afin de comprendre comment les arcs ont été tracés. Les deuxième et troisième types d'arcs sont évidents puisqu'ils modélisent les relations fournies par l'expert telles que mentionnées à la section 1.2. En ce qui concerne le premier type d'arc, il faut analyser les événements deux à deux pour voir s'il y a inclusion. Par exemple, on a un arc $(8, 6)$ puisque $S_8 \subseteq S_6$ et ainsi, $\pi_8 \leq \pi_6$. En effet, en analysant ces deux événements, on constate que si S_8 est vrai, alors S_6 est vrai car x_2 est alors vrai. Une analyse similaire peut être réalisée pour expliquer la présence de tous les arcs de ce type dans le graphe. Par ailleurs, en analysant le graphe de la figure 3.1, on constate facilement la présence de relations indirectes entre certaines paires de probabilités. Par exemple, puisque $\pi_9 \leq \pi_1$ et $\pi_1 \leq \pi_5$, alors, par transitivité, $\pi_9 \leq \pi_5$. Cette analyse nous amène à la prochaine étape de la méthode, c'est-à-dire la détermination des relations induites par transitivité.

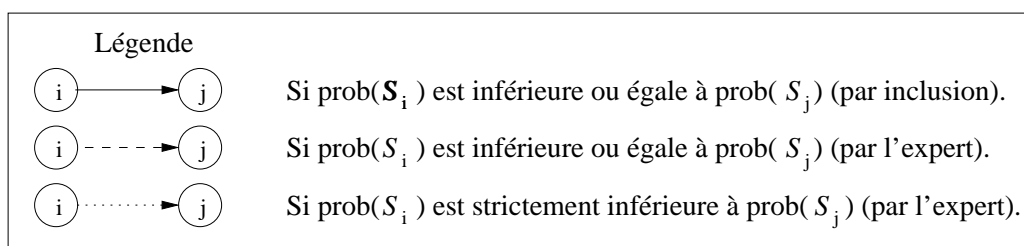
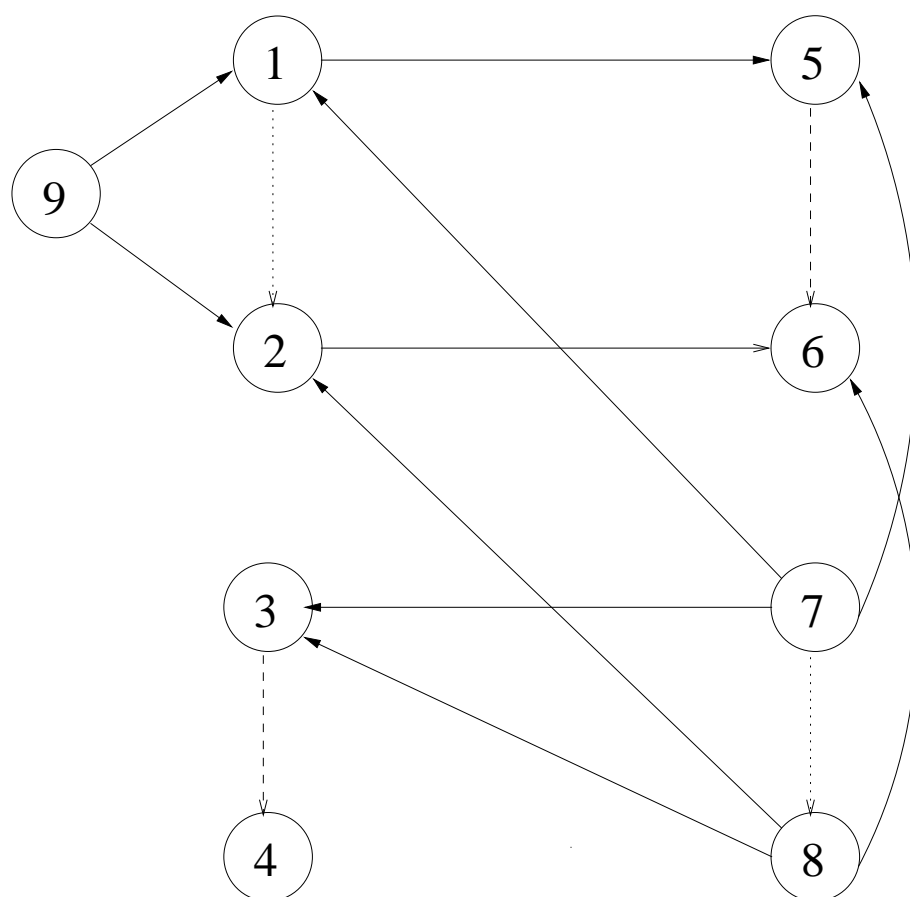


FIG. 3.1 – Graphe d'implication de l'exemple 1.

Formellement, si G contient l'arc (i, j) et l'arc (j, k) , alors, par transitivité, on peut ajouter l'arc (i, k) . En effet, si $\pi_i \leq \pi_j$ et $\pi_j \leq \pi_k$, alors, par transitivité, on a que $\pi_i \leq \pi_k$. Ainsi, un chemin orienté P de G reliant deux sommets i et j implique $\pi_i \leq \pi_j$ (en excluant les arcs représentant des relations d'inégalité stricte ; ce dernier cas sera traité à la section 3.6.2). Si on veut ajouter tous les arcs indiquant les relations indirectes, c'est-à-dire ajouter un arc (i, j) pour chaque paire de sommets telle qu'il existe un chemin orienté reliant i et j dans G , il suffit de déterminer la fermeture transitive du graphe. L'algorithme retenu dans le cadre de ce projet pour réaliser cette opération est une version de l'algorithme développé par Roy [57] (et redécouvert indépendamment par Warshall [59]). Soit G' , le graphe obtenu à la fin du déroulement de l'algorithme. Pour savoir s'il existe une relation directe ou indirecte entre les probabilités de deux événements S_i et S_j , il suffit de vérifier s'il existe un arc (i, j) ou (j, i) dans G' . Sans la réalisation de la fermeture transitive du graphe, nous aurions été dans l'obligation de vérifier si G contenait un chemin orienté de i vers j ou de j vers i .

Lorsqu'on effectue les comparaisons entre π_{m+1} et π_i , pour $i = 1, 2, \dots, m$, on vérifie d'abord si on connaît la relation entre π_{m+1} et π_i en examinant les arcs entre les sommets $m + 1$ et i dans G' . Si on ne connaît pas totalement la relation entre les deux probabilités, on calcule le minimum et/ou le maximum de la différence entre ces probabilités. Par la suite, une fois la relation trouvée par l'analyse du minimum ou du maximum, on peut déterminer la relation entre π_{m+1} et la probabilité des événements S_j pour lesquels, il existe un arc (i, j) ou (j, i) selon le cas. Formellement, la méthode améliorée s'énonce ainsi :

1. Résoudre le *problème de la cohérence des jugements de probabilité*. On tente de trouver une solution réalisable au problème en résolvant le problème (3.2). Si le système admet une solution, on passe à l'étape 2. Sinon, on arrête.
2. Tracer le graphe d'implication G' .
3. Effectuer la fermeture transitive du graphe G pour obtenir le graphe G' .

4. Pour chaque événement S_i (pour $i = 1, 2, \dots, m$) :

- (a) Déterminer, par l'analyse de G' , si on connaît déjà la relation entre π_{m+1} et π_i . Si on ne connaît pas la relation entre π_{m+1} et π_i , on passe à (b).
- (b) Vérifier s'il existe une relation indirecte entre π_{m+1} et π_i non déterminée par G' . On résout, tout d'abord, le programme linéaire (3.9). La solution optimale de ce problème nous donne la valeur minimale de la différence entre π_{m+1} et π_i . Cette solution nous donne non seulement la relation entre π_{m+1} et π_i mais également celle entre π_{m+1} et la probabilité des successeurs ou prédécesseurs de i dans G' . On met à jour G' avec ces nouvelles relations trouvées. Formellement, trois cas sont possibles :

$$\min(\pi_{m+1} - \pi_i) \left\{ \begin{array}{l} < 0, & \text{si la relation } \pi_{m+1} \geq \pi_i \text{ n'est pas toujours vraie.} \\ & \text{Ainsi, } \pi_{m+1} \text{ n'est pas toujours supérieure ou} \\ & \text{égale à } \pi_j \text{ pour tout } (i, j) \in A'. \\ = 0, & \text{si } \pi_{m+1} \geq \pi_i. \\ & \text{Ainsi, } \pi_{m+1} \geq \pi_j \text{ pour tout } j \text{ tel que } (j, i) \in A'. \\ > 0, & \text{si } \pi_{m+1} > \pi_i. \\ & \text{Ainsi, } \pi_{m+1} > \pi_j \text{ pour tout } j \text{ tel que } (j, i) \in A'. \end{array} \right.$$

Pour les cas où $\pi_{m+1} - \pi_i \geq 0$ à l'optimum, alors on tente de résoudre le problème (3.9) en inversant le sens de l'objectif (maximisation au lieu de minimisation). Encore une fois, 3 cas sont possibles :

$$\max(\pi_{m+1} - \pi_i) \left\{ \begin{array}{l} > 0, & \text{si la relation } \pi_{m+1} \leq \pi_i \text{ n'est pas toujours vraie.} \\ & \text{Ainsi, } \pi_{m+1} \text{ n'est pas toujours inférieure ou} \\ & \text{égale à } \pi_j \text{ pour tout } (j, i) \in A'. \\ = 0, & \text{si } \pi_{m+1} \leq \pi_i. \\ & \text{Ainsi, } \pi_{m+1} \leq \pi_j \text{ pour tout } j \text{ tel que } (i, j) \in A'. \\ < 0, & \text{si } \pi_{m+1} < \pi_i. \\ & \text{Ainsi, } \pi_{m+1} < \pi_j \text{ pour tout } j \text{ tel que } (i, j) \in A'. \end{array} \right.$$

Si on obtient la première alternative dans les deux cas ($\min(\pi_{m+1} - \pi_i) < 0$ et $\max(\pi_{m+1} - \pi_i) > 0$), alors S_{m+1} et S_i sont incomparables en termes de probabilités. Si on obtient la deuxième alternative dans les deux cas ($\min(\pi_{m+1} - \pi_i) = \max(\pi_{m+1} - \pi_i) = 0$), alors $\pi_{m+1} = \pi_i$.

Cette amélioration permet de réduire sensiblement le nombre de comparaisons à effectuer. Pour s'en convaincre, analysons le graphe d'implication de l'**exemple 1** (voir figure 3.1). Par exemple, si on trouve en résolvant le modèle 3.9 que la valeur minimale de $\pi_9 - \pi_3$ est égale à 0, alors on peut affirmer que $\pi_9 \geq \pi_3$ et, par transitivité, que π_9 est supérieure ou égale à π_8 et π_7 . Par la suite, il sera inutile de vérifier si S_7 et S_8 sont au plus aussi probables que S_9 car on détient déjà l'information. Par conséquent, on effectuera moins de comparaisons puisque, sans l'utilisation du graphe d'implication, nous aurions été dans l'obligation d'effectuer ces vérifications.

On peut alors se poser une question : l'ordre dans lequel les comparaisons sont effectuées peut-il influencer le nombre de comparaisons à réaliser et ce faisant, le temps d'exécution de l'algorithme ? La réponse à cette question n'est pas triviale. Pour y répondre, nous avons décidé de comparer trois façons de choisir l'ordre dans lequel on compare la probabilité de chaque événement avec celle de S_{m+1} :

1. ordre croissant des indices des événements,
2. parmi les sommets de G' non comparés avec S_{m+1} et dont un des demi-degrés est nul, choisir, à chaque fois, le sommet avec le degré maximal,
3. parmi tous les sommets de G' non comparés avec S_{m+1} , choisir, à chaque fois, le sommet de degré maximal.

Rappelons que dans un graphe orienté, chaque sommet a deux demi-degrés : le demi-degré intérieur qui correspond au nombre d'arcs entrants dans le sommet et le demi-degré extérieur qui correspond au nombre d'arcs sortants du sommet. Le degré d'un sommet est égal à la somme des deux demi-degrés, c'est-à-dire le nombre total d'arcs incidents au sommet. Lorsqu'on compare un sommet avec l'objectif, on met à jour les degrés des sommets qui précèdent et suivent ce sommet dans G' en réduisant d'une unité leur demi-degré extérieur et leur demi-degré intérieur respectivement. Notons que le premier critère de sélection ne tient pas compte de la structure du graphe d'implication tandis que les deux autres comparent, en premier lieu, les événements qui sont éventuellement susceptibles de nous procurer plus d'informations. En effet, si on choisit un événement dont le sommet du graphe est connecté

à plusieurs sommets, il est probable que la connaissance de la relation entre la probabilité de cet événement et celle de S_{m+1} nous procure beaucoup d'information sur les sommets connectés à ce sommet. Au chapitre 4, on tente de voir si une de ces méthodes a un comportement empirique supérieur aux deux autres. Pour ce faire, on analyse les résultats numériques obtenus par chacune de ces méthodes lors de la réalisation d'une série de tests.

3.6 Extension avec inégalités strictes

Il s'agit du même problème qu'à la section précédente sauf qu'on considère des relations d'inégalités strictes. Pour le modèle, cela se traduit par l'ajout d'un paramètre t . Ce dernier permet d'utiliser l'algorithme révisé du simplexe en variables bornées pour résoudre le problème en transformant les inégalités strictes en inégalités non strictes. Encore une fois, on tentera de déterminer les bornes inférieure et supérieure les meilleures possibles sur π_{m+1} (section 3.6.1) ainsi que les relations entre π_{m+1} et π_i , pour $i = 1, 2 \dots, m$ (section 3.6.2).

3.6.1 Bornes inférieure et supérieure sur π_{m+1}

Définition du problème

On a une proposition S_{m+1} supplémentaire, on essaie de déterminer les bornes inférieure et supérieure les meilleures possibles sur π_{m+1} en fonction de la valeur du paramètre t . Ainsi, on tente de voir, en effectuant une analyse paramétrique sur t , l'influence de ce paramètre sur la valeur optimale de π_{m+1} . La méthode de résolution consiste donc à appliquer les concepts de l'analyse paramétrique de la programmation linéaire pour notre algorithme de génération de colonnes (voir Chvátal [14] pour une description de l'analyse paramétrique de la programmation linéaire dans un cadre général).

Méthode de résolution

Formellement, on applique la méthode en deux étapes qui suit :

1. Résolution du problème (3.7) de la cohérence des jugements de probabilité avec relations d'inégalité stricte (section 3.4). Si le problème est cohérent, on passe à 2. Sinon, on arrête le déroulement de l'algorithme.

2. On change l'objectif du modèle (3.7) et on y ajoute la contrainte $t \geq b$. L'objectif est le suivant : $\min/\max \pi_{m+1} = A_{m+1}p$. Après ces modifications, le modèle obtenu est le suivant :

$$\begin{aligned}
 & \min/\max \quad A_{m+1}p \\
 & \text{ sujet à :} \\
 & \quad \mathbb{1}p \quad = \quad 1 \\
 & \quad Ap - \pi \quad = \quad 0 \\
 & \quad \pi_k - \pi_l \quad \leq \quad 0 \quad \forall (k, l) \in Q_1 \\
 & \quad \pi_k - \pi_l + t \leq 0 \quad \forall (k, l) \in Q_2 \\
 & \quad \quad \quad \quad t \geq b \\
 & \quad \quad \quad \quad p \geq 0 \\
 & \quad \underline{\pi} \leq \pi \leq \bar{\pi}
 \end{aligned} \tag{3.10}$$

On examine tout d'abord si la valeur optimale de l'objectif de 3.10 change avec la variation de t . Pour ce faire, on résout avec b égal à 0 et b égal à la valeur maximale de t trouvée en 1. Si les valeurs optimales obtenues sont identiques, alors on arrête puisqu'on sait que t n'a pas d'influence sur la valeur de π_{m+1} . Sinon, on résout le problème en faisant varier b pour déterminer les bornes inférieure et supérieure les meilleures possibles sur π_{m+1} en fonction de t . On commence avec b égal à 0 et on l'augmente jusqu'à ce qu'il atteigne la valeur maximale de t . Puisque la valeur optimale de l'objectif en fonction de t est une fonction convexe ou concave, et linéaire par morceaux, tout au long du déroulement de l'analyse paramétrique, on vérifie si la pente de l'objectif en fonction de t (pour la valeur de b courante) est la même que celle pour b égal à la valeur maximale de t . Si c'est le cas, on peut arrêter l'analyse paramétrique puisqu'on sait que la fonction est linéaire jusqu'à la valeur maximale de t . Formellement, on effectue l'analyse paramétrique telle que décrite dans la figure 3.2.

La méthode retenue dans ce projet de recherche pour retrouver une solution de base admissible au problème fait appel à l'algorithme primal du simplexe. Il aurait été possible d'utiliser l'algorithme dual du simplexe pour réaliser cette tâche (voir Chvátal [14]). Cependant, étant donné que nous utilisons l'algorithme de génération de colonnes pour résoudre (3.10), les colonnes associées aux variables p_j ne sont pas toutes connues. Ainsi, nous serions obligés de résoudre un programme hyperbolique pour retrouver une solution admissible par l'algorithme dual. La résolution d'un

$b = 0$;
 Tant que $b \leq \text{Max-}t$ et $\text{Pente-courante} \neq \text{Pente-max-}t$, faire

- Résoudre (3.10) ;
- Trouver par analyse de sensibilité sur b , l'intervalle de valeur pour lequel la solution de base courante reste admissible et optimale ;
- $u =$ borne sup. trouvée ;
- $b = u + \epsilon$;
- Retrouver une solution de base admissible de (3.10) ;

 Fin tant que.

$\text{Max-}t$ correspond à la valeur maximale de t trouvée lors du *problème de la cohérence des jugements de probabilité* (section 3.4). Pente-courante et $\text{Pente-max-}t$ représentent la valeur de la pente de la valeur optimale de l'objectif en fonction de t pour la valeur de b courante et pour $b = \text{Max-}t$ respectivement.

FIG. 3.2 – Schéma de l'analyse paramétrique.

tel programme est une tâche très difficile. Cette approche n'a donc pas été retenue pour le présent projet. La méthode primale, pour restaurer l'admissibilité du programme (3.10), consiste à ajouter une variable artificielle à chaque contrainte dont la variable de base atteint une valeur hors de ses bornes suite à la modification de b . Ces variables permettent de retrouver une solution de base rapidement. Par la suite, on applique une phase 1 de l'algorithme révisé du simplexe pour éliminer ces variables artificielles et trouver une nouvelle solution de base admissible au problème. Une fois l'analyse paramétrique terminée, l'expert peut vérifier l'impact de t sur l'objectif et utiliser cette information pour prendre une décision éclairée et rationnelle quant à la valeur minimale qu'il désire donner à t . En d'autres termes, il peut déterminer une valeur pour l'écart minimal entre les probabilités dans les relations d'inégalité stricte.

3.6.2 Relation entre la probabilité π_{m+1} et chacune des m probabilités π_i

Définition du problème

Comme dans le cas sans inégalité stricte (section 3.5.2), on a un événement S_{m+1} supplémentaire et il faut trouver, pour chaque événement S_i , pour $i = 1, 2, \dots, m$, si ce dernier est :

1. plus probable que S_{m+1} ($\pi_i > \pi_{m+1}$),
2. au moins aussi probable que S_{m+1} ($\pi_i \geq \pi_{m+1}$),
3. moins probable que S_{m+1} ($\pi_i < \pi_{m+1}$),
4. au plus aussi probable que S_{m+1} ($\pi_i \leq \pi_{m+1}$),
5. incomparable, en termes de probabilités, avec S_{m+1} .

Méthode de résolution

La méthode de résolution est très semblable à celle utilisée dans le cas sans inégalité stricte. Les contraintes du modèle sont les mêmes que celle de (3.10) avec $b = 0$. Cependant, lors des comparaisons, on vérifie, lorsque la valeur minimale (maximale) de $\pi_{m+1} - \pi_i$ est strictement négative (strictement positive), si une variation de t peut renverser la conclusion. En d'autres termes, si une variation de t peut impliquer un changement de signe de la valeur optimale obtenue. Cette opération est réalisée par l'entremise de l'analyse paramétrique. Notons qu'il n'est pas toujours nécessaire de réaliser cette analyse. On vérifie donc, au départ si, lorsque b correspond à la valeur maximale de t trouvée dans le problème de la cohérence (section 3.4), la valeur de l'objectif change de signe. Si oui, on effectue l'analyse paramétrique pour trouver la valeur minimale de t pour laquelle la conclusion est renversée. Sinon, il est inutile d'effectuer l'analyse paramétrique puisque une variation de t ne permet pas de modifier le signe de la valeur optimale de $\pi_{m+1} - \pi_i$. Notons qu'on utilise, encore une fois, la valeur de la pente de la valeur optimale de l'objectif en fonction de t pour réduire le temps passé à effectuer l'analyse paramétrique. Ainsi, tout au

long du déroulement de l'analyse paramétrique, on vérifie si la pente de la valeur optimale de l'objectif en fonction de t (pour la valeur de b courante) est la même que celle pour b égal à la valeur maximale de t . Si c'est le cas, on peut arrêter l'analyse paramétrique. On peut, par la suite, retrouver la valeur minimale de t pour laquelle la conclusion est renversée par interpolation linéaire.

De plus, on doit inclure, dans le graphe d'implication, les arcs représentant les relations d'inégalité stricte. Soit $G = (N, A)$, le graphe d'implication incluant l'ensemble des r relations. Pour le calcul de la fermeture transitive du graphe G , il faut tenir compte du fait qu'une inégalité stricte implique une relation plus forte qu'une inégalité non stricte. Par exemple, si $\pi_i \leq \pi_j$ et $\pi_j < \pi_k$, alors, par transitivité, on a que $\pi_i < \pi_k$. Ainsi, si dans un chemin P de G reliant deux sommet i et j , il existe au moins un arc (k, l) représentant une relation d'inégalité stricte, alors on peut ajouter un arc (i, j) indiquant que $\pi_i < \pi_j$. Cette affirmation est facile à prouver. Supposons que l'arc (k, l) soit le seul arc de P représentant une relation d'inégalité stricte. Le chemin de i vers k indique que $\pi_i \leq \pi_k$. De même, le chemin de l vers j indique que $\pi_l \leq \pi_j$. Puisque $\pi_k < \pi_l$, par transitivité, $\pi_i < \pi_l$ et donc $\pi_i < \pi_j$. Par exemple, dans le graphe d'implication de l'**exemple 1** (figure 3.1), on remarque que $\pi_9 \leq \pi_1$ et que $\pi_1 < \pi_2$, alors $\pi_9 < \pi_2$. Ainsi, on peut remplacer l'arc $(9, 2)$ représentant une relation d'inégalité non stricte par un autre arc $(9, 2)$ représentant une relation d'inégalité stricte.

3.7 Discussion sur *QPSAT* avec probabilités conditionnelles

Dans le cadre de ce problème, on considère également m événements S_1, S_2, \dots, S_m définis sur un ensemble de n variables logiques x_1, x_2, \dots, x_n . De plus, parmi les r relations d'inégalité stricte ou non stricte, on détient des relations entre des probabilités conditionnelles (par exemple, $\text{prob}(S_1|S_2) \leq \text{prob}(S_3|S_4)$). Tel que mentionné dans Coletti [21], on remarque que si les événements conditionnants sont les mêmes, alors le problème demeure linéaire. L'exemple suivant nous permet de constater l'exactitude de cette affirmation.

Exemple 5. Soit, 3 événements S_1, S_2, S_3 et la relation suivante :

$$\text{prob}(S_1|S_2) \leq \text{prob}(S_3|S_2)$$

En utilisant la définition d'une probabilité conditionnelle, on obtient la relation équivalente suivante :

$$\frac{\text{prob}(S_1 \wedge S_2)}{\text{prob}(S_2)} \leq \frac{\text{prob}(S_3 \wedge S_2)}{\text{prob}(S_2)} \equiv$$

En simplifiant, on a :

$$\text{prob}(S_1 \wedge S_2) \leq \text{prob}(S_3 \wedge S_2)$$

La dernière relation peut être facilement incorporée aux modèles obtenus aux sections précédentes sans toutefois perdre la linéarité de ces modèles.

Par ailleurs, lorsque les événements conditionnants sont distincts, on se retrouve avec des programmes contenant des termes quadratiques. Néanmoins, il est possible de formuler judicieusement les contraintes permettant de modéliser les relations entre deux probabilités conditionnelles de manière à obtenir beaucoup moins de termes quadratiques que dans la formulation (3.1) de Coletti. Pour ce faire, on insère les contraintes permettant de définir le numérateur et le dénominateur de la définition

d'une probabilité conditionnelle séparément de la relation entre les deux probabilités conditionnelles. Voici un exemple de contraintes modélisant la relation suivante :

$$\text{prob}(S_q|S_v) \leq \text{prob}(S_k|S_l).$$

$$\begin{aligned} A_{q \wedge v} p &= \pi_{q \wedge v} \\ A_{k \wedge l} p &= \pi_{k \wedge l} \\ A_v p &= \pi_v \\ A_l p &= \pi_l \\ \pi_{q \wedge v} \pi_l &\leq \pi_{k \wedge l} \pi_v \end{aligned} \tag{3.11}$$

où $A_{q \wedge v} = (a_{q \wedge v, j})$ avec $a_{q \wedge v, j} = 1$ si les événements S_q et S_v sont tous deux vrais dans le monde possible w_j et 0 sinon ; $A_v = (a_{v, j})$ avec $a_{v, j} = 1$ si S_v est vrai dans le monde possible w_j et 0 sinon.

L'ajout de ce type de contraintes, pour chaque relation entre deux probabilités conditionnelles, conduit à un modèle avec peu de termes quadratiques. En fait, cette formulation est beaucoup plus efficace que la formulation (3.1) de Coletti. Pour s'en convaincre, regardons la différence entre ces deux formulations sur le petit exemple suivant :

Exemple 6. Soit 4 événements S_1, S_2, S_3, S_4 définis ainsi :

$$\begin{aligned} S_1 &\equiv x_1 \\ S_2 &\equiv x_2 \\ S_3 &\equiv x_3 \\ S_4 &\equiv x_4 \end{aligned}$$

En définissant les mondes possibles sur les variables, on obtient les 16 mêmes mondes que pour l'exemple 1 (section 2.2.1). Puisque chaque proposition correspond à une variable, en définissant les mondes possibles sur les propositions, on obtient les 16 mêmes mondes (voir section 2.2.2 pour la description des deux façons de définir les mondes).

Supposons que l'on ait la relation suivante :

$$\text{prob}(S_1|S_2) \leq \text{prob}(S_3|S_4)$$

Si on modélise cette relation à l'aide de la formulation 3.11, on obtient les contraintes suivantes :

$$\begin{array}{rcl}
p_1+p_2+p_3+p_4 & & =\pi_{1\wedge 2} \\
p_1+p_2+p_3+p_4 & +p_9+p_{10}+p_{11}+p_{12} & =\pi_2 \\
p_1 & +p_5 & +p_9 & +p_{13} & =\pi_{3\wedge 4} \\
p_1 & +p_3 & +p_5 & +p_7+p_9 & +p_{11} & +p_{13}+p_{15} & =\pi_4 \\
& & & & & & \pi_{1\wedge 2}\times\pi_4\leq\pi_{3\wedge 4}\times\pi_2 \\
p_1,p_2, p_3, \dots,p_{16} & \geq 0 & & & & &
\end{array}$$

Si on utilise la formulation (3.1) de Coletti pour modéliser cette relation, en excluant les inégalités indiquant que la probabilité d'un événement conditionnant est non nul, les 5 premières contraintes sont alors remplacées par la contrainte suivante :

$$\begin{aligned}
& (p_1 + p_2 + p_3 + p_4) \times (p_1 + p_3 + p_5 + p_7 + p_9 + p_{11} + p_{13} + p_{15}) \leq \\
& (p_1 + p_2 + p_3 + p_4 + p_9 + p_{10} + p_{11} + p_{12}) \times (p_1 + p_5 + p_9 + p_{13})
\end{aligned}$$

En effectuant le produit des sommes des variables p_j dans cette équation, on constate que la formulation de Coletti génère beaucoup plus de termes quadratiques.

Une méthode envisageable pour résoudre des *QPSAT* avec probabilités conditionnelles consiste à adapter les techniques développées par Audet, Hansen, Jaumard et Savard [3] pour résoudre des programmes avec des termes quadratiques. Cette méthode consiste à appliquer un algorithme de séparation et coupes pour trouver, en temps fini, une solution approchée au problème de programmation quadratique à contraintes quadratiques. L'idée générale de cet algorithme consiste à estimer chaque terme quadratique par des linéarisations successives. L'adaptation de cet algorithme pour la résolution de *QPSAT* en génération de colonnes sera traité dans le cadre de mon projet de doctorat.

CHAPITRE 4

RÉSULTATS NUMÉRIQUES

4.1 Introduction

Ce chapitre a pour but de présenter les résultats numériques obtenus par nos algorithmes pour la résolution de *QPSAT* générés aléatoirement. L'information procurée par ces résultats permet d'analyser judicieusement la performance des méthodes retenues dans le cadre de ce projet. Ainsi, il est possible d'évaluer empiriquement l'impact de la modification de la taille du problème (nombre d'événements, nombre de variables ou nombre de relations) sur les temps de calcul.

Dans la section 4.2, on décrit de façon succincte la méthode utilisée pour générer les problèmes. La section 4.3 est consacrée à l'étude des résultats obtenus. On y présente donc, sous forme de tableaux, les résultats de diverses expériences numériques ainsi qu'une brève analyse de ces résultats. Notons que tous les algorithmes ont été programmés en langage C et exécutés sur une station SUN ULTRA-2 avec 300 Mhz et 384M de ram. De plus, la partie programmation linéaire a été résolue à l'aide du logiciel CPLEX 5.0.

4.2 Génération de problèmes tests

Les problèmes visant à étudier la performance de nos algorithmes ont été conçus à l'aide d'un générateur de problèmes. Le générateur que nous avons implanté génère des problèmes où chaque proposition ou événement correspond à une clause, c'est-à-dire à un produit logique de variables booléennes (par exemple $x_1 \wedge x_2$). Le nombre maximal de variables pouvant se retrouver sous forme directe ou complémentée à l'intérieur d'une clause est fixé à 5. De plus, puisque l'expert n'est pas toujours en mesure de fournir une estimation de la probabilité de chaque événement, nous avons décidé de fixer à 15% la proportion d'événements pour lesquels on détient un intervalle sur sa probabilité d'être vraie. Ainsi, pour chaque événement, on génère un nombre aléatoire compris dans l'intervalle $[0,1]$. Si ce nombre est plus grand que 85%, alors on calcule un intervalle sur la probabilité de cet événement. Les intervalles sont obtenus en générant aléatoirement un ensemble de vecteurs d'assignations de valeurs vraie ou fausse aux n variables logiques (le nombre de vecteurs générés = $\min\{2^n, 1024\}$, où n = nombre de variables). On calcule ensuite, pour chaque événement dont on veut trouver un intervalle sur sa probabilité, la proportion des vecteurs d'assignations de valeurs aux variables logiques induisant une assignation de valeur vraie à l'événement en question. En d'autres termes, on détermine le pourcentage de vecteurs pour lesquels toutes les variables qui définissent l'événement sont vraies. Ensuite, on établit l'intervalle de manière à ce que la valeur trouvée (\bar{p}) soit comprise dans cet intervalle. Plus formellement, on génère aléatoirement deux nombres u_1 et u_2 compris dans l'intervalle $[0,1]$ et on calcule l'intervalle ainsi : $[\bar{p} \times u_1, \bar{p} + (1 - \bar{p}) \times u_2]$. Ainsi, on trouve un intervalle de probabilités qui est cohérent dans le cas où on ne tient pas compte de probabilités qualitatives. Cette façon de procéder nous permet de calculer simplement des probabilités pour chacun des événements dont on veut déterminer un intervalle de probabilités.

En ce qui concerne les relations entre certaines paires de probabilités (probabilités qualitatives), on génère un ensemble de relations qui ne sont pas déjà impliquées par une inclusion. De plus, on exclut les relations qui engendrent directement une incohérence (par exemple, $\pi_1 < \pi_2$ et $\pi_1 > \pi_2$) ou une équivalence entre deux probabilités (par exemple, $\pi_1 \leq \pi_2$ et $\pi_1 \geq \pi_2$). Par ailleurs, pour les problèmes contenant

des relations d'inégalité stricte, la proportion du nombre d'inégalités strictes par rapport au nombre total de relations est établie à 15%. Ainsi, pour chaque relation, on génère un nombre aléatoire compris dans l'intervalle $[0,1]$. Si ce nombre est plus grand que 85%, alors on indique qu'il s'agit d'une inégalité stricte. Lorsque le nombre de relations n'est pas mentionné, il est fixé à 50% du nombre m d'événements. Ainsi, dans les 4 premiers tableaux de la prochaines sections, le nombre de relations est toujours égal à 50% de m . Finalement, notons que pour chaque taille de problème, 10 problèmes différents ont été générés.

4.3 Résultats et analyse

Dans cette section, on présente 5 tableaux visant à faire une analyse des méthodes algorithmiques retenues dans le cadre de ce projet de recherche. Tout d'abord, on essaie d'identifier laquelle, parmi les trois façons de déterminer l'ordre des comparaisons (voir section 3.5.2), performe le mieux. En second lieu, on compare la performance des deux méthodes (méthode générale et méthode améliorée) permettant de déterminer les relations entre les probabilités des m événements et celle de S_{m+1} . Finalement, on tente d'analyser l'impact d'une modification du nombre n de variables, du nombre m d'événements et du nombre r de relations sur les temps de calculs ainsi que le nombre d'appels des différentes étapes de la résolution d'un *QP-SAT*. Avant d'analyser chacun de ces tableaux, regardons les définitions essentielles à la compréhension de ceux-ci.

Légende des tableaux :

- n : le nombre de variables logiques ;
- m : le nombre d'événements ;
- r : le nombre de relations ;
- App. μ : le nombre moyen d'appels de l'algorithme (heuristique taboue ou algorithme exact) ;
- App. σ : l'écart-type du nombre d'appels de l'algorithme (heuristique taboue ou algorithme exact) ;
- Temps μ : le temps cpu moyen ;
- Temps σ : l'écart-type du temps cpu ;
- Incoh. : l'ensemble des problèmes incohérents ;
- Coh. : l'ensemble des problèmes cohérents ;
- Fréq. : la fréquence (en %) du nombre de problèmes cohérents sur les 10 problèmes testés ;
- Nb. comp. μ : le nombre moyen de comparaisons ;
- Nb. comp. σ : l'écart-type du nombre de comparaisons.

En guise de première analyse, nous avons décidé de comparer la performance des trois méthodes de détermination de l'ordre des comparaisons (voir section 3.5.2 pour la description de ces méthodes). Le tableau 4.1 présente donc, pour chacune des 4 tailles différentes de problèmes et chaque méthode, la moyenne et l'écart-type du temps cpu passé à déterminer les relations pour les problèmes, parmi les 10 générés, qui sont cohérents. Cette analyse sommaire nous permet de constater que la méthode consistant à comparer les événements selon l'ordre décroissant des degrés des sommets qui leur sont associés dans le graphe d'implication, semble donner des temps moyens de calculs légèrement plus faibles. De plus, les valeurs d'écart-type du temps de calculs pour cette méthode sont comparables ou inférieures à celles des 2 autres méthodes. Bien qu'il s'agit d'une analyse très sommaire et que les temps de calculs ne semblent pas être très différents d'une méthode à l'autre, nous avons décidé d'utiliser la méthode des degrés maximaux dans la suite de nos expériences numériques.

Problèmes	n	25	25	50	50
	m	25	50	50	100
Ordre aléatoire	Temps μ	0,63	5,02	4,77	42,29
	Temps σ	0,17	1,69	2,02	17,73
Degré max parmi deg. nuls	Temps μ	0,66	5,19	4,91	42,09
	Temps σ	0,17	1,42	2,43	19,05
Degré max	Temps μ	0,55	4,98	4,68	38,82
	Temps σ	0,16	1,84	2,41	16,18

TAB. 4.1 – Résultats d'exécution en déterminant l'ordre des comparaisons selon trois méthodes.

La prochaine analyse consiste à vérifier la pertinence des améliorations apportées à la méthode générale de détermination des relations entre la probabilité de S_{m+1} et celle de chacun des m événements (voir section 3.5.2). Le tableau 4.2 présente donc, pour chacune des 4 tailles différentes de problèmes et chaque méthode, la moyenne et l'écart-type du temps cpu et du nombre de comparaisons nécessaires à la détermination des relations pour les problèmes, parmi les 10 générés, qui sont cohérents. Notons que seules les comparaisons nécessitant la résolution d'un programme linéaire (voir section 3.5.2) sont comptabilisées. En observant ce tableau, on constate que les améliorations permettent de réduire sensiblement le nombre de comparaisons effectuées et ce faisant, les temps de calculs. En effet, la méthode améliorée

diminue le temps moyen de calculs pour déterminer les relations ainsi que le nombre moyen de comparaisons d'environ 40%.

Problèmes	n	25	25	50	50
	m	25	50	50	100
Méthode générale	Temps μ	0,93	8,11	6,46	69,07
	Temps σ	0,26	2,47	2,74	29,00
	Nb. comp. μ	48,67	99,14	99,78	198,57
	Nb. comp. σ	2,60	0,69	0,44	2,15
Méthode améliorée	Temps μ	0,55	4,98	4,68	38,82
	Temps σ	0,16	1,84	2,41	16,18
	Nb. comp. μ	29,56	61,71	62,67	112,71
	Nb. comp. σ	5,05	8,94	5,24	7,72

TAB. 4.2 – Méthode générale vs. méthode améliorée pour la détermination des relations.

Les trois prochains tableaux permettent de vérifier l'impact d'une augmentation du nombre n de variables, du nombre m d'événements et du nombre r de relations sur le temps de calculs et le nombre d'appels des différentes phases de l'algorithme de résolution de *QPSAT*. Chacun de ces 3 tableaux est séparé en 4 sections. Les trois premières sections présentent les statistiques sur le temps total passé dans chaque phase de résolution ainsi que le temps et le nombre d'appels des algorithmes heuristique et exact. Dans la première section, on affiche les résultats pour le test de la cohérence. Les statistiques (moyenne et écart-type) sont alors calculées sur les 10 problèmes générés. Les deux autres sections présentent les statistiques, pour le calcul des bornes et la détermination des relations, calculées sur l'ensemble des problèmes cohérents. Finalement, la dernière section est divisée en deux sous-sections. La première présente la moyenne et l'écart-type du temps total de calculs pour les problèmes incohérents. Évidemment, le temps de calculs comprend alors uniquement le temps passé à vérifier si le problème est cohérent. La seconde sous-section nous donne la fréquence (en %) du nombre de problèmes cohérents sur les 10 problèmes testés ainsi que la moyenne et l'écart-type du temps total de calculs pour les problèmes cohérents. De plus, les tableaux 4.3 et 4.4 présentent les résultats d'expériences de calculs lorsqu'on fait varier n et m pour des problèmes sans ou avec inégalités strictes. Finalement, le tableau 4.5 est consacré à une analyse du comportement des algorithmes implantés lorsque r varie (n et m sont constants et fixés à

50). La suite de ce chapitre est consacrée à une brève analyse de l'information la plus pertinente pouvant être extraite de ces tableaux.

Tout d'abord, en analysant les tableaux 4.3 et 4.4, on remarque que les statistiques sur le temps cpu et le nombre d'appels pour toutes les étapes de résolution d'un problème sont plus faibles pour les problèmes ne contenant pas d'inégalité stricte. Ce résultat était prévisible puisque la présence d'inégalités strictes rend plus difficiles le test de la cohérence, le calcul des bornes ainsi que la détermination des relations étant donné la présence de la variable t . Cependant, l'écart entre les deux types de problèmes est beaucoup moins grand pour le test de la cohérence des problèmes incohérents que pour les autres phases de l'algorithme de résolution. Par ailleurs, l'écart est généralement plus grand pour les étapes du calcul des bornes et de la détermination des relations que pour le test de la cohérence. Ceci est dû à la présence de la variable t qui oblige la résolution de plusieurs analyses paramétriques (résolution de plusieurs programmes linéaires à l'optimalité). De plus, lorsque la taille du problème augmente, l'écart en proportion entre les temps de résolution des problèmes sans et avec inégalités strictes s'accroît. Par ailleurs, on constate que la fréquence de cohérence des *QPSAT* sans inégalité stricte est inférieure ou égale à celle des problèmes avec inégalités strictes pour chacune des différentes tailles de problèmes. Cette constatation était aussi à prévoir car une relation d'inégalité stricte est une contrainte plus difficile à respecter qu'une relation non stricte.

En ce qui a trait à la vérification de la cohérence, on constate tout d'abord que les problèmes incohérents sont généralement plus longs à résoudre que les problèmes cohérents. En effet, on voit dans les tableaux que la moyenne des temps de calculs de cohérence pour l'ensemble des problèmes (première ligne du tableau) est généralement moins élevée que celle pour les problèmes incohérents (lorsqu'il y a des problèmes incohérents). Ce résultat est attribuable au fait qu'il est habituellement plus difficile de prouver qu'un problème n'admet pas de solution que d'en trouver une s'il en admet. En ce qui concerne la fréquence de cohérence, on réalise en observant les tableaux 4.3 et 4.4 qu'elle diminue normalement lorsqu'on augmente le nombre m d'événements tout en gardant le nombre n de variables constant. Ce résultat peut s'expliquer par le fait que plus m est grand pour un n constant, alors plus il peut y avoir de relations dues à des inclusions et ainsi créer de l'incohérence si on ajoute les

r relations. Par ailleurs, en examinant le tableau 4.5, on réalise, comme on pouvait l'imaginer, que la fréquence de cohérence diminue lorsque le nombre r de relations augmente dans le cas où n et m sont constants.

L'analyse des tableaux nous permet également de tirer des conclusions sur la pertinence d'utiliser l'algorithme de recherche avec tabous pour la résolution du problème auxiliaire. En fait, cet algorithme semble permettre de réduire sensiblement les temps de calculs par rapport à la résolution de manière exacte du problème auxiliaire à chaque étape. En effet, cet algorithme est appelé souvent et son temps moyen pour chaque appel est beaucoup plus faible (temps moyen de l'algorithme divisé par le nombre moyen d'appels). Ainsi, si on utilisait l'algorithme exact à chaque fois, le temps de calculs s'accroîtrait. Cependant, on remarque que, contrairement aux étapes du test de la cohérence et du calcul des bornes, la détermination des relations exige une utilisation plus importante de l'algorithme exact. En examinant les tableaux 4.3 et 4.4, on constate en effet que le temps passé dans l'algorithme exact pour l'étape de détermination des relations n'est pas toujours inférieur au temps passé dans la méthode heuristique taboue. Il est, en fait, supérieur lorsque m est grand par rapport à n . Ceci peut s'expliquer par le fait qu'il faut résoudre beaucoup de comparaisons et ainsi résoudre beaucoup de programmes linéaires à l'optimalité. Ce qui exige la résolution de manière exacte du problème auxiliaire afin de prouver l'optimalité d'une solution. Ce résultat est d'autant plus vrai pour les problèmes contenant des inégalités strictes puisqu'il faut effectuer plusieurs analyses paramétriques et ainsi vérifier l'optimalité de plusieurs programmes linéaires.

Analysons maintenant l'impact de la modification des paramètres n , m et r sur les résultats. On constate, en examinant les tableaux 4.3 et 4.4, que lorsqu'on double le nombre n de variables pour une valeur de m constante, les temps moyens de calculs vont bien souvent moins que doubler. À l'inverse, les temps moyens de calculs vont habituellement plus que doubler si on double le nombre d'événements pour un nombre constant de variables. Par conséquent, les temps de résolution sont très sensibles au nombre m d'événements et beaucoup moins au nombre n de variables. Ce résultat était prévisible puisque, tel que mentionné dans Chvátal [14], le nombre d'itérations de l'algorithme du simplexe est très sensible au nombre de contraintes et beaucoup moins au nombre de variables. Ce résultat est également attribuable

au fait que la matrice B de l'algorithme révisé du simplexe est plus volumineuse et donc plus difficile à inverser. De plus, en ce qui concerne la détermination des relations, le nombre de comparaisons à effectuer et ce faisant les temps de calculs, dépendent énormément du nombre d'événements. Voilà pourquoi l'augmentation du temps moyen de détermination des relations est généralement plus importante que celle des autres phases (cohérence et calcul des bornes). Enfin, lorsque r double, le temps de calculs pour la cohérence augmente généralement de plus que du double et le temps de détermination des relations de moins que du double. Le premier résultat s'explique par le fait que, tel que mentionné ci-dessus, une augmentation du nombre de contraintes implique une augmentation du nombre d'itérations de l'algorithme du simplexe ainsi qu'une plus grande difficulté lors de l'inversion de la matrice B . Le second résultat s'explique par le fait que l'augmentation du nombre de relations peut diminuer le nombre de comparaisons à effectuer car on se retrouve avec un graphe d'implication contenant plus d'arcs et donc plus de relations indirectes entre les événements. Ainsi, cela peut compenser pour l'augmentation du nombre de contraintes.

En conclusion, bien que la méthode améliorée diminue significativement les temps de calculs, la détermination des relations demeure l'étape cruciale dans la résolution d'un *QPSAT*. En effet, plus m augmente et plus le temps pour effectuer cette opération est grand. Cela est d'autant plus vrai pour les problèmes contenant des inégalités strictes car on doit alors effectuer plusieurs analyses paramétriques exigeant la résolution d'un très grand nombre de modèles linéaires.

Problèmes		n	25	25	25	50	50	50	100	100	100	
		m	25	50	100	50	100	150	100	150	200	
Cohé- rence	Total	Temps μ	0,02	0,04	0,22	0,04	0,19	0,32	0,21	0,62	0,85	
		Temps σ	0,01	0,03	0,23	0,01	0,13	0,23	0,14	0,56	0,53	
	Heur.	Temps μ	0,01	0,01	0,05	0,02	0,07	0,08	0,10	0,21	0,25	
		Temps σ	0,01	0,01	0,03	0,01	0,02	0,07	0,06	0,15	0,12	
	Tabou	App. μ	8,20	14,60	35,90	10,30	30,20	35,70	30,30	56,80	59,40	
		App. σ	3,97	10,31	24,33	3,20	17,25	24,48	15,56	40,19	29,50	
	Alg. Exact	Temps μ	0,00	0,00	0,03	0,00	0,02	0,00	0,00	0,03	0,00	
		Temps σ	0,00	0,00	0,09	0,00	0,04	0,00	0,00	0,07	0,00	
		App. μ	0,30	0,10	0,90	0,20	1,20	0,40	0,70	2,90	0,40	
		App. σ	0,67	0,32	2,18	0,42	2,39	0,70	0,95	6,21	0,70	
	Bornes	Total	Temps μ	0,02	0,17	0,20	0,07	0,16	0,79	0,48	0,62	0,93
			Temps σ	0,01	0,27	0,09	0,04	0,08	0,78	0,43	0,34	0,48
Heur.		Temps μ	0,01	0,04	0,07	0,03	0,04	0,17	0,15	0,17	0,18	
		Temps σ	0,01	0,05	0,03	0,02	0,03	0,12	0,14	0,10	0,09	
Tabou		App. μ	10,10	31,44	31,71	19,60	22,71	60,40	56,70	47,50	47,33	
		App. σ	2,88	23,47	12,38	12,40	11,01	51,43	40,89	23,81	24,98	
Alg. Exact		Temps μ	0,00	0,06	0,02	0,01	0,00	0,00	0,02	0,01	0,01	
		Temps σ	0,00	0,17	0,02	0,01	0,01	0,01	0,02	0,01	0,01	
		App. μ	3,20	6,33	2,86	5,10	2,57	2,80	6,50	3,88	3,00	
		App. σ	1,87	8,87	1,86	3,84	0,79	1,10	3,89	1,46	1,32	
Rela- tions		Total	Temps μ	0,30	3,01	10,76	2,00	13,09	76,49	21,29	60,93	246,93
			Temps σ	0,08	4,96	4,80	0,52	2,70	36,49	6,19	9,36	110,52
	Heur.	Temps μ	0,07	0,24	0,86	0,36	1,21	3,56	2,36	3,80	7,22	
		Temps σ	0,03	0,07	0,37	0,09	0,24	0,82	0,54	0,63	1,79	
	Tabou	App. μ	105,40	228,78	535,29	302,00	666,29	1441,60	873,90	1141,62	1850,00	
		App. σ	19,04	44,07	172,28	76,12	115,85	331,09	153,16	191,81	454,11	
	Alg. Exact	Temps μ	0,08	1,78	2,38	0,26	0,98	8,86	0,77	0,91	18,69	
		Temps σ	0,04	4,54	1,79	0,10	1,05	14,68	0,43	0,15	49,89	
		App. μ	44,60	89,22	156,29	125,80	176,71	251,00	271,10	268,00	344,22	
		App. σ	7,09	35,82	62,13	32,08	28,72	43,97	86,91	24,91	82,96	
Algo. au complet	Incoh.	Temps μ	0,00	0,04	0,75	0,00	0,61	1,21	0,00	2,49	4,18	
		Temps σ	0,00	0,00	0,31	0,00	0,18	0,31	0,00	0,69	0,00	
	Coh.	Fréq.	100,00	90,00	70,00	100,00	70,00	50,00	100,00	80,00	90,00	
		Temps μ	0,35	3,27	11,37	2,15	13,65	78,57	22,26	62,86	250,76	
		Temps σ	0,09	5,26	4,80	0,56	2,77	37,21	6,47	9,32	110,56	

TAB. 4.3 – Résultats d'exécution sur des *QPSAT* sans inégalité stricte en faisant varier le nombre de variables et le nombre d'événements.

Problèmes		n	25	25	25	50	50	50	100	100	100	
		m	25	50	100	50	100	150	100	150	200	
Cohé- rence	Total	Temps μ	0,03	0,09	0,39	0,09	0,32	0,74	0,60	1,26	2,17	
		Temps σ	0,02	0,05	0,30	0,06	0,12	0,76	0,38	0,99	1,47	
	Heur.	Temps μ	0,01	0,03	0,09	0,03	0,10	0,19	0,23	0,40	0,49	
		Temps σ	0,01	0,01	0,04	0,02	0,04	0,19	0,16	0,29	0,26	
	Tabou	App. μ	14,80	27,60	54,00	25,60	51,10	67,90	79,40	107,90	118,30	
		App. σ	5,69	11,95	24,77	14,03	17,14	58,07	43,85	77,38	63,44	
	Alg. Exact	Temps μ	0,01	0,01	0,09	0,01	0,01	0,03	0,03	0,03	0,01	
		Temps σ	0,01	0,01	0,19	0,01	0,01	0,04	0,03	0,04	0,02	
		App. μ	2,20	2,30	2,60	3,40	2,10	2,40	7,90	4,70	1,90	
		App. σ	1,99	1,34	1,84	2,37	1,37	3,57	10,91	5,01	1,10	
	Bornes	Total	Temps μ	0,03	0,33	0,63	0,18	0,61	2,21	0,89	1,63	3,31
			Temps σ	0,01	0,42	0,49	0,14	0,53	2,62	0,58	0,60	2,06
Heur.		Temps μ	0,01	0,04	0,10	0,05	0,10	0,18	0,22	0,23	0,40	
		Temps σ	0,01	0,02	0,06	0,03	0,06	0,14	0,19	0,12	0,22	
Tabou		App. μ	13,44	46,71	53,80	34,33	59,14	91,60	73,12	81,43	99,00	
		App. σ	4,19	20,67	22,88	18,53	49,17	82,11	50,85	27,34	57,28	
Alg. Exact		Temps μ	0,01	0,15	0,20	0,03	0,05	0,28	0,05	0,07	0,10	
		Temps σ	0,01	0,34	0,35	0,04	0,06	0,48	0,04	0,05	0,12	
		App. μ	6,22	23,57	16,80	13,11	22,00	33,20	23,50	23,29	16,56	
		App. σ	1,92	19,29	11,86	12,57	25,90	52,03	20,08	13,40	15,32	
Rela- tions		Total	Temps μ	0,56	5,02	41,07	4,65	39,90	237,10	66,90	189,76	935,13
			Temps σ	0,12	1,69	32,73	2,01	17,40	164,17	27,48	55,32	762,42
	Heur.	Temps μ	0,08	0,42	1,37	0,52	1,53	4,22	3,19	5,17	9,38	
		Temps σ	0,03	0,10	0,65	0,18	0,34	0,94	0,84	1,04	2,56	
	Tabou	App. μ	152,22	444,71	886,80	420,00	903,71	1912,60	1252,12	1556,71	2464,89	
		App. σ	24,91	86,14	396,72	108,29	185,42	446,20	299,64	257,58	652,29	
	Alg. Exact	Temps μ	0,14	0,98	10,20	0,45	2,45	30,80	1,45	1,77	74,24	
		Temps σ	0,08	0,95	12,96	0,22	3,43	51,54	0,76	0,33	162,55	
		App. μ	83,67	260,71	414,00	220,00	356,43	558,60	499,38	499,86	687,78	
		App. σ	12,81	71,70	279,67	57,68	87,88	202,96	179,23	35,91	215,63	
	Algo. au complet	Incoh.	Temps μ	0,02	0,08	0,72	0,10	0,61	1,24	0,50	2,63	5,16
			Temps σ	0,00	0,03	0,40	0,00	0,15	0,40	0,02	1,76	0,00
Coh.		Fréq.	90,00	70,00	50,00	90,00	70,00	50,00	80,00	70,00	90,00	
		Temps μ	0,65	5,49	42,32	4,97	41,10	241,40	68,77	193,38	942,70	
		Temps σ	0,13	2,07	33,34	2,15	17,83	167,47	27,88	55,95	765,67	

TAB. 4.4 – Résultats d'exécution sur des *QPSAT* avec inégalités strictes en faisant varier le nombre de variables et le nombre d'événements.

Problèmes		r	10	25	50	100	200	
Cohé- rence	Total	Temps μ	0,05	0,09	0,13	0,47	4,21	
		Temps σ	0,02	0,06	0,10	0,82	7,95	
	Heur. Tabou	Temps μ	0,02	0,03	0,04	0,09	0,14	
		Temps σ	0,01	0,02	0,03	0,13	0,23	
		App. μ	17,30	25,60	30,80	54,60	88,30	
	Alg. Exact	App. σ	5,08	14,03	22,06	82,44	142,84	
		Temps μ	0,00	0,01	0,01	0,12	3,52	
		Temps σ	0,00	0,01	0,01	0,36	7,01	
		App. μ	3,00	3,40	1,70	5,10	12,60	
	Bornes	Total	App. σ	1,89	2,37	1,64	12,71	22,24
			Temps μ	0,07	0,18	0,37	0,00	0,00
		Heur. Tabou	Temps σ	0,03	0,14	0,21	0,00	0,00
Temps μ			0,03	0,05	0,07	0,00	0,00	
Temps σ			0,02	0,03	0,05	0,00	0,00	
Alg. Exact		App. μ	22,80	34,33	54,00	0,00	0,00	
		App. σ	10,74	18,53	28,35	0,00	0,00	
		Temps μ	0,01	0,03	0,07	0,00	0,00	
		Temps σ	0,01	0,04	0,07	0,00	0,00	
Rela- tions		Total	App. μ	9,30	13,11	18,33	0,00	0,00
			App. σ	5,91	12,57	12,50	0,00	0,00
		Heur. Tabou	Temps μ	2,94	4,65	5,13	0,00	0,00
	Temps σ		0,89	2,01	2,17	0,00	0,00	
	Temps μ		0,46	0,52	0,54	0,00	0,00	
	Alg. Exact	Temps σ	0,12	0,18	0,13	0,00	0,00	
		App. μ	430,90	420,00	364,00	0,00	0,00	
		App. σ	60,63	108,29	102,09	0,00	0,00	
		Temps μ	0,36	0,45	0,54	0,00	0,00	
	Algo. au complet	Incoh.	Temps σ	0,06	0,22	0,35	0,00	0,00
			App. μ	249,60	220,00	141,67	0,00	0,00
		Coh.	App. σ	40,62	57,68	56,77	0,00	0,00
Fréq.			100,00	90,00	30,00	0,00	0,00	
		Temps μ	3,11	4,97	5,78	0,00	0,00	
		Temps σ	0,93	2,15	2,48	0,00	0,00	

TAB. 4.5 – Résultats d'exécution sur des *QPSAT* avec inégalités strictes en faisant varier le nombre de relations.

CHAPITRE 5

CONCLUSION

Dans ce travail de recherche, nous avons étudié une extension du *problème de la satisfiabilité probabiliste* incorporant des probabilités qualitatives. Le cadre général du problème consistait à considérer un ensemble d'événements et leur probabilité d'être vrai. La probabilité de chaque événement était soit totalement inconnue ou comprise à l'intérieur d'un intervalle fourni par un expert. On considérait également un ensemble de relations d'inégalité stricte ou non stricte entre les probabilités de certaines paires d'événements (probabilités qualitatives). L'incorporation de probabilités qualitatives peut s'avérer fort pratique pour le traitement de problèmes réels. En effet, dans certains cas, l'expert peut ne pas être en mesure de donner une estimation de la probabilité de certains événements mais peut tout de même nous transmettre une série de jugements du type "A est plus probable que B" ou "A est au moins aussi probable que B".

Le premier problème que l'on a alors résolu, le *problème de la cohérence des jugements de probabilité*, consistait à vérifier si les jugements de probabilité étaient cohérents. Dans le cas où les jugements étaient cohérents, on était alors intéressé à analyser la probabilité d'être vrai d'un événement supplémentaire. On voulait alors trouver les bornes inférieure et supérieure les meilleures possibles sur cette probabilité. On était également intéressé à déterminer les relations qui existent entre la probabilité du nouvel événement et celle de chacun des autres événements. Ces deux dernières interrogations conduisaient au deuxième problème étudié dans ce projet de recherche, soit le *problème de l'extension*.

Dans ce projet de recherche, on a donc développé des méthodes algorithmiques pour résoudre efficacement ces deux problèmes. Les méthodes utilisées étaient basées essentiellement sur la technique de génération de colonnes jointe à la programmation non-linéaire en variables 0–1 ainsi que l’analyse paramétrique de la programmation linéaire. Les résultats numériques ont démontré qu’il était possible d’utiliser ces méthodes pour résoudre des problèmes d’assez grande taille en des temps raisonnables. En effet, nous avons réussi à résoudre des problèmes contenant jusqu’à 100 variables, 200 événements et 100 relations en des temps moyens de calculs d’environ 16 minutes. Notons que dans ce cas, environ 5 secondes en moyenne étaient nécessaires à la résolution du *problème de la cohérence des jugements de probabilité* et du calcul des bornes. Le temps restant était utilisé à trouver les relations entre la probabilité de l’événement supplémentaire et celle de chacun des autres événements. Il s’agit donc de l’étape la plus cruciale dans la résolution d’un *problème de la satisfiabilité probabiliste qualitative*. Pour l’avenir, il serait donc intéressant de voir si des améliorations peuvent être apportées à la méthode utilisée pour déterminer les relations dans le but de réduire les temps de calculs.

Finalement, pour le traitement de problèmes réels, il serait intéressant de pouvoir incorporer des probabilités conditionnelles dans la définition du problème. Ainsi, on pourrait tenir compte d’événements pour lesquels, les experts sont en mesure d’estimer la probabilité ou de la comparer à celle d’un autre événement uniquement dans le cas où une condition est satisfaite. Nous avons montré que la formulation mathématique d’un *problème de la satisfiabilité probabiliste qualitative* avec probabilités conditionnelles donne généralement des programmes complexes contenant des termes quadratiques. Bien que le nombre de termes quadratiques soit bien souvent assez petit, la résolution de ce type de problème s’avère être une tâche difficile. Je vais donc, dans le cadre de mon projet de doctorat, tenter d’élaborer une méthode efficace pour résoudre ce type de problèmes.

Bibliographie

- [1] Adams, E.W. et H.P. Levine, On the Uncertainties Transmitted from Premises to Conclusions in Deductive Inferences, *Synthese* 30 (1975) 429–460.
- [2] Andersen, K.A., et J.N. Hooker, Bayesian Logic, *Decision Support Systems* 11 (1994) 191–210.
- [3] Audet, C., P. Hansen, B. Jaumard et G. Savard, A Branch and Cut Algorithm for Nonconvex Quadratically Constrained Quadratic Programming, *Les Cahiers du GERAD*, G-97-67, December 1997, 33 pages.
- [4] Boole, G., Proposed Question in the Theory of Probabilities, *The Cambridge and Dublin Mathematical Journal* 6 (1851) 186.
- [5] Boole, G., Further Observations on the Theory of Probabilities, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 4(2) (1851) 96–101.
- [6] Boole, G., *Collected Logical Works. Vol I, Studies in Logic and Probability*, ed. R. Rhees. LaSalle, Illinois : Open Court, 1952.
- [7] Boole, G., *An Investigation of the Laws of Thought, on which are Founded the Mathematical Theories of Logic and Probabilities*, London : Walton and Maberley, 1854 (reprint New York : Dover 1958).
- [8] Boole, G., On the Conditions by which Solutions of Questions in the Theory of Probabilities are Limited, *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science* 4(8) (1854) 91–98.
- [9] Boole, G., On a General Method in the Theory of Probabilities, *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science* 4(8) (1854) 431–444.

- [10] Boole, G., On Certain Propositions in Algebra Connected to the Theory of Probabilities, *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science* 4(9) (1855) 165–179.
- [11] Boole, G., On Propositions Numerically Definite (read posthumously by De Morgan March 16th, 1868), *Transactions of the Cambridge Philosophical Society* 11 (1871) 396–411.
- [12] Charnes, A., et W.W. Cooper, Programming with Linear Fractional Functionals, *Naval Research Logistics Quarterly* 9 (1962) 181–186.
- [13] Chesnokov, S.V., The Effect of Semantic Freedom in the Logic of Natural Language, *Fuzzy Sets and Systems* 22 (1987) 121–154.
- [14] Chvátal, V., *Linear Programming*, Freeman, New York, 1980.
- [15] Coletti, G., Conditionally Coherent Qualitative Probabilities, *Statistica* 48 (1988) 235–242.
- [16] Coletti, G., Coherent Qualitative Probability, *Journal of Mathematical Psychology* 34 (1990) 298–310.
- [17] Coletti, G., Numerical and Qualitative Judgments in Probabilistic Expert Systems, in R. Scozzafava (ed.) Proceedings of the *Workshop on Probabilistic Expert Systems*, Roma SIS (1993) 37–55.
- [18] Coletti, G., A weak Coherence Condition for Conditional Comparative Probabilities, *Uncertainty in Intelligent Systems*, (Ed. B. Bouchon-Meunier et al.) Elsevier Science Publisher B.V., (1993) 195–201.
- [19] Coletti, G., Coherent Numerical and Ordinal Probabilistic Assessments, *IEEE Transactions on Systems, Man and Cybernetics* 24(12) (1994).
- [20] Coletti, G., A. Gilio, R. Scozzafava, Conditional Events with Vague Information in Expert Systems, *Lecture Notes in Computer Science* (Eds. B. Bouchon-Meunier, R.R Yager and L.A. Zadeh), 521 (1991) 106–114.
- [21] Coletti, G., A. Gilio, R. Scozzafava, Assessment of Qualitative Judgements for Conditional Events in Expert Systems, *Lecture Notes in Computer Science* (Eds. R. Kruse and P. Siegel) 548 (1991) 135–140.
- [22] Coletti, G., A. Gilio, R. Scozzafava, Comparative Probability for Conditional Events : A new Look Through Coherence, *Theory and Decision* 35 1993 237–258.

- [23] Coletti, G., R. Scozzafava, Learning from Data by Coherent Probabilistic Reasoning, *Proceedings of the Third International Symposium on Uncertainty Modeling and Analysis and Annual Conference of the North American Fuzzy Information Processing Society* College Park, MD, 1995.
- [24] Coletti, G., R. Scozzafava, Characterization of Coherent Conditional Probabilities as a Tool for their Assessment and Extension, *Research Report, Dept. Math Univ di Perugia*, Italy, 1996.
- [25] de Finetti, B., Problemi determinati e indeterminati nel calcolo delle probabilità, *Rendiconti Reale Accademia dei Lincei* 6(XII) (1930) 367–373.
- [26] de Finetti, B., La prévision : ses lois logiques, ses sources subjectives, *Annales de l'Institut Henri Poincaré* 7 (1937) 1–68.
- [27] de Finetti, B., *Theory of Probability – A Critical Introductory Treatment*, Vol. 1, Wiley, New York, 1974.
- [28] de Finetti, B., *Theory of Probability – A Critical Introductory Treatment*, Vol. 2, Wiley, New York, 1975.
- [29] Douanya-Nguetse, G.-B., P. Hansen, B. Jaumard, Probabilistic Satisfiability and Decomposition, *Les Cahiers du GERAD*, G-94-55, December 1994, 15 pages.
- [30] Dowsland K.A., Simulated Annealing, in C.R. Reeves (ed.) *Modern Heuristic Techniques for Combinatorial Problems* London : Blackwell (1993) 20–69.
- [31] Fine T.L., *Theories of Probability*, Academic Press, New York and London, 1973.
- [32] Fortet, R., L'algèbre de Boole et ses applications en Recherche Opérationnelle, *Cahiers du Centre d'Études de Recherche Opérationnelle* 1 :4 (1959) 5–36.
- [33] Fortet, R., Applications de l'algèbre de Boole en Recherche Opérationnelle, *Revue Française d'Informatique et de Recherche Opérationnelle* 4 :14 (1960) 17–25.
- [34] Gelembé, E., Une généralisation probabiliste du problème SAT, *Comptes Rendus de l'Académie des Sciences de Paris* 315 (1992) 339–342.
- [35] Georgakopoulos, G., D. Kavvadias, et C.H. Papadimitriou, Probabilistic Satisfiability, *Journal of Complexity* 4 (1988) 1–11.

- [36] Gilmore, P.C., R.E. Gomory, A Linear Programming Approach to the Cutting Stock Problem, *Operations Research* 9 (1961) 849–859.
- [37] Glover, F., Tabu Search — Part I, *ORSA Journal on Computing* 1 (1989) 190–206.
- [38] Glover, F., Tabu Search — Part II, *ORSA Journal on Computing* 2 (1990) 4–32.
- [39] Hailperin, T., Best Possible Inequalities for the Probability of a Logical Function of Events, *American Mathematical Monthly* 72 (1965) 343–359.
- [40] Hailperin, T., *Boole's Logic and Probability*, Studies in Logic and the Foundations of Mathematics 85, North Holland, Amsterdam, first edition, 1976.
- [41] Hailperin, T., *Boole's Logic and Probability*, Studies in Logic and the Foundations of Mathematics 85, North Holland, Amsterdam, 2nd enlarged edition, 1986.
- [42] Hammer, P.L., et S. Rudeanu, *Boolean Methods in Operations Research and Related Areas*, Berlin : Springer, 1966.
- [43] Hansen, P. et B. Jaumard, Algorithms for the Maximum Satisfiability Problem, *Computing* 44 (1990) 279–303.
- [44] Hansen, P., B. Jaumard et V. Mathon, Constrained Nonlinear 0–1 Programming, *ORSA Journal on Computing* 5 (1993) 97–119.
- [45] Hansen, P. et B. Jaumard, Probabilistic Satisfiability, *Les Cahiers du GERAD* G–96–31, June 1996, 63 pages.
- [46] Hansen, P., B. Jaumard et G.B. Douanya Nguetse, Best Second Order Bounds for Two-terminal Network Reliability with Dependent Edge Failures, *Les Cahiers du GERAD* G–94–01, February 1994, 23 pages.
- [47] Hansen, P., B. Jaumard et M. Poggi de Aragão, Boole's Conditions of Possible Experience and Reasoning Under Uncertainty, *Discrete Applied Mathematics* 60 (1995) 181–193.
- [48] Jaumard, B., P. Hansen et M. Poggi de Aragão, Column Generation Methods for Probabilistic Logic, *ORSA Journal on Computing* 3 (1991) 135–148.
- [49] Kavvadias, D. et C.H. Papadimitriou, A Linear Programming Approach to Reasoning about Probabilities, *Annals of Mathematics and Artificial Intelligence* 1 (1990) 189–205.

- [50] Keynes, J.M., *A Treatise on Probability*, London : Macmillan, 1921.
- [51] Kounias S. et J. Marin, Best Linear Bonferroni Bounds, *SIAM Journal on Applied Mathematics* 30 (1976) 307–323.
- [52] Lad, F., J.M. Dickey et M.A. Rahman, The Fundamental Theorem of Prevision, *Statistica* 50 (1990) 19–38.
- [53] Lin, S. et B.W. Kernighan, An Effective Heuristic Algorithm for the Traveling Salesman Problem, *Operations Research* 21 (1973) 498–516.
- [54] Marsten, R.E., The design of the XMP Linear Programming Library, *ACM Transactions on Mathematical Software* 7(4) (1981) 481–497.
- [55] Medolaghi, La logica matematica e il calcolo delle probabilita, *Bolletino Associazione Italiani di Attuari* 18 (1907).
- [56] Nilsson, N.J., Probabilistic Logic, *Artificial Intelligence* 28(1) (1986) 71–87.
- [57] Roy, B., Transitivité et connexité, *Compt. Rend. Acad. Sci. Paris* 249 (1959) 216.
- [58] Scozzafava, R., G. Colleti, Qualitative Bayesian Inference, *Proc. of the Meeting of the American Statistical Association (Sect. on Bayesian Statistics)*, San Francisco, (1993) 184-191. Alexandria : ASA.
- [59] Warshall, S., A theorem on Boolean Matrices, *J. Assos. Comp. Mach.* 9 (1962) 11–12.
- [60] Wilbraham, H., On the Theory of Chances Developed in Professor Boole’s “Laws of Thought”, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 4(7) (1854) 465–476.
- [61] Zemel, E., Polynomial Algorithms for Estimating Network Reliability, *Networks* 12 (1982) 439–452.