



HAL
open science

Actes des 9es Journées d'Intelligence Artificielle Fondamentale

Nicolas Maudet, Bruno Zanuttini

► **To cite this version:**

Nicolas Maudet, Bruno Zanuttini. Actes des 9es Journées d'Intelligence Artificielle Fondamentale : JIAF 2015. Plate-Forme Intelligence Artificielle, Association Française pour l'Intelligence Artificielle, 2015. hal-04596515

HAL Id: hal-04596515

<https://ut3-toulouseinp.hal.science/hal-04596515>

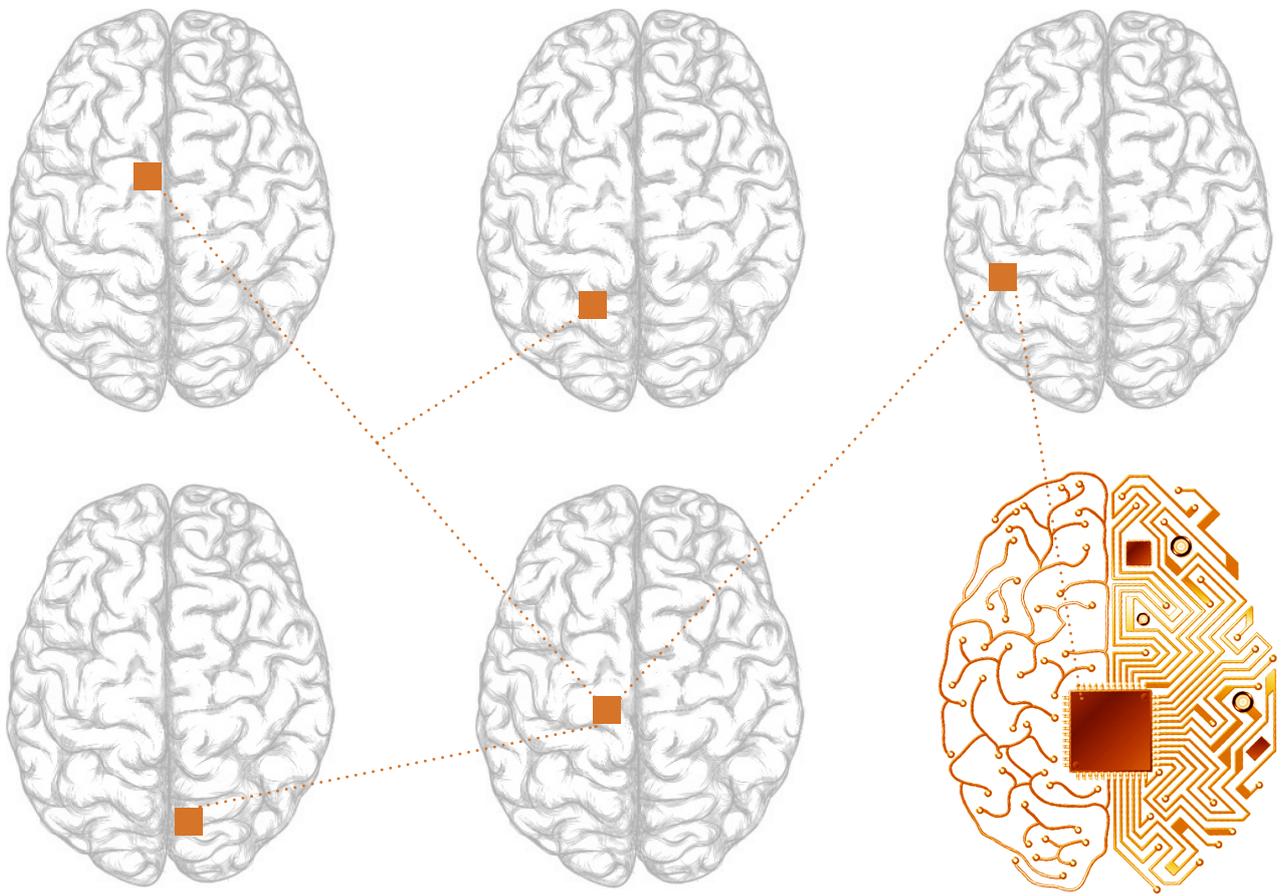
Submitted on 31 May 2024

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

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



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0
International License



PFIA 2015

Plate-forme Intelligence Artificielle
Rennes

Actes IAF

Présidents CP : Nicolas Maudet et Bruno Zanuttini



AFIA

Association française
pour l'Intelligence Artificielle

9^{es} Journées d'Intelligence Artificielle Fondamentale

Présentation de la conférence

La plate-forme accueille les Journées d'Intelligence Artificielle Fondamentales (IAF), qui émanent d'une volonté d'animation initiée dans le cadre du thème Intelligence Artificielle Fondamentale de l'ancien GDR-I3.

Les journées constituent un rendez-vous régulier de la communauté française d'Intelligence Artificielle Fondamentale. Les thématiques de recherche sont relatives aux méthodes et outils fondamentaux de l'Intelligence Artificielle. Elles portent sur :

- la définition de modèles de représentation des informations (croyances, connaissances, préférences, obligations et permissions, actions, incertitude, confiance, réputation) comme les langages des logiques classiques ou non classiques, les modèles possibilistes, les ontologies, les langages à base de contraintes, les représentations graphiques, etc.,
- la définition et l'automatisation de raisonnements sur ces informations : raisonnement spatio-temporel, dynamique des informations, révision de croyances, fusion d'informations symboliques, raisonnement par argumentation, raisonnement causal, raisonnement abductif, raisonnement à partir de cas etc.,
- la mise au point de méthodes de codage des informations et d'algorithmes de traitement efficaces : compilation de connaissances, SAT, contraintes, ASP, etc.,
- la modélisation formelle de l'interaction : entre utilisateurs et systèmes informatiques, entre entités informatiques autonomes (agents), intégration de ces deux aspects dans les divers agents conversationnels, agents de recherche, assistants personnels...
- pour des objectifs de décision, planification, ordonnancement, diagnostic, apprentissage et dans différents contextes d'application, comme par exemple le Web sémantique.

Ces journées seront composées à parts égales :

- d'exposés de synthèse,
- de communications sélectionnées par le comité de programme.

Conférencier Invité : Eyke Hüllermeier (Department of Comput Science, University of Paderborn)

Biographie

Eyke Hüllermeier est professeur à l'Université de Marburg, responsable du groupe de recherche Computational Intelligence. Depuis son doctorat à l'Université de Paderborn, Eyke a occupé plusieurs postes dans des centres de recherches tels que l'IRIT, en tant que Marie-Curie Fellow de 1998 à 2000, ou la faculté de sciences informatiques à l'Otto-von-Guericke-Universität Magdeburg. Ses domaines de recherche sont l'intelligence artificielle, les systèmes à base de connaissances et les statistiques. Il s'intéresse en particulier aux logiques floues et méthodes d'apprentissage automatique. Eyke est co-éditeur en chef de la revue *Fuzzy Sets and Systems* et membre de comités éditoriaux de nombreux journaux de référence.

Preferences in Artificial Intelligence and Machine Learning

Reasoning with preferences has gained increasing attention in the field of artificial intelligence (AI) in the recent years. Amongst others, this interest has led to the emergence of preference learning as a new branch of machine learning. Roughly speaking, preference learning refers to the development and analysis of methods for learning preference models from explicit or implicit preference information, that is, models useful for explaining revealed and predicting future preferences. Starting with a brief introduction to preferences in AI in general and machine learning in particular, this talk will focus on methods and algorithms for ranking problems. These are (learning) problems in which preferences are represented in terms of a total order over an underlying set of choice alternatives, where both the order relation and the choice set can be contextualized. Based on different types of preference feedback, the goal of the learner is to induce a model that allows for predicting such order relations in an accurate manner.

Comité de Programme

- Christian Bessière (LIRMM, Montpellier)
- Tristan Cazenave (LAMSADE, Paris)

- Laurence Cholvy (ONERA, Toulouse)
- Marie-Odile Cordier (IRISA, Rennes)
- Nadia Creignou (LIF, Marseille)
- Jérôme Euzenat (LIG, Grenoble)
- Tiago de Lima (CRIL, Lens)
- Sébastien Konieczny (CRIL, Lens)
- Fred Koriche (CRIL, Lens)
- Jérôme Lang (LAMSADE, Paris)
- Jean Lieber (LORIA, Nancy)
- Pierre Marquis (CRIL, Lens)
- Nicolas Maudet (LIP6, Paris), coprésident
- Marie-Laure Mugnier (LIRMM, Montpellier)
- Amedeo Napoli (LORIA, Nancy)
- Odile Papini (LSIS, Marseille)
- Laurent Perrussel (IRIT, Toulouse)
- Henri Prade (IRIT, Toulouse)
- Pierre Siegel (LIF, Marseille)
- Igor Stéphan (LERIA, Angers)
- Christel Vrain (LIFO, Orleans)
- Bruno Zanuttini (GREYC, Caen), coprésident

Comment citer

Exemple de citation :

*Prénom1 Nom1, Prénom2 Nom2, and Prénom3 Nom3. **Le titre d'un article.** In Actes des 9es Journées d'Intelligence Artificielle Fondamentale (IAF 2015), 2015.*

Exemple d'entrée BibTex :

```
@InProceedings{UnNom.IAF.2015,
  author =      {{Nom1, Pr\'enom1 and Nom2, Pr\'enom2 and Nom3, Pr\'enom3}},
  title =      {Le titre d'un article},
  booktitle =  {Actes des 9es Journ\'ees d'Intelligence Artificielle Fondamentale (IAF~2015)},
  year =      {2015}
}
```

Table des matières

Francesco Belardinelli. A Logic of Knowledge and Strategies with Imperfect Information	7
Khaled Skander Ben Slimane, Alexis Comte, Olivier Gasquet, Abdelwahab Heba, Olivier Lezaud, Frédéric Maris, Maël Valais. La logique facile avec TouIST (formalisez et résolvez facilement des problèmes du monde réel)	17
Sylvain Bouveret, Michel Lemaître. Séquences de choix sincères et partages équitables de biens indivisibles	25
Quentin Cohen-Solal, Maroua Bouzid, Alexandre Niveau. Une algèbre des relations temporelles granulaires pour le raisonnement qualitatif	35
Sylvie Coste-Marquis, Sébastien Konieczny, Jean-Guy Mailly, Pierre Marquis. Forçage d'extension en argumentation abstraite par optimisation booléenne	45
Nadia Creignou, Raida Ktari, Odile Papini. Révision de bases de croyances : complexité de la vérification de modèle	55
Jérôme Delobelle, Sébastien Konieczny, Srdjan Vesic. Agrégation de systèmes d'argumentation	63
Didier Dubois, Henri Prade, Agnès Rico. Le cube des oppositions – Une structure à la base de nombreux formalismes de représentation des connaissances	73
Andreas Herzig, Emiliano Lorini, Faustine Maffre. Une logique épistémique modeste basée sur les observations d'ordre supérieur	83
Nicolas Hug, Henri Prade, Gilles Richard. Raisonnement analogique pour la recommandation : premières expérimentations	93
Amélie Levray, Salem Benferhat, Karim Tabia. Transformations probabilistes-possibilistes : conditionnement, inférence et modèles graphiques	103
Bénédicte Legastelois, Marie-Jeanne Lesot, Adrien Revault D'Allonnes. Typologie d'axiomes pour une logique modale pondérée	113
Jean Lieber. Révision des croyances dans une clôture propositionnelle de contraintes linéaires	123
Nicola Olivetti, Gian Luca Pozzato. A natural sequent calculus for Lewis' counterfactual logic \forall	135
Michael Sioutis, Jean-François Condotta, Yakoub Salhi, Bertrand Mazure. Generalized Qualitative Spatio-Temporal Reasoning : Complexity and Tableau Method	145
Bruno Zanuttini. Sur des propriétés structurelles des formules de Horn	155

A Logic of Knowledge and Strategies with Imperfect Information*

Francesco Belardinelli

Laboratoire IBISC, Université d'Evry, France
belardinelli@ibisc.fr

Abstract

In this paper we put forward Epistemic Strategy Logic (ESL), a logic of knowledge and strategies in contexts of imperfect information. ESL extends Strategy Logic by adding modal operators for individual and collective knowledge. This enhanced framework allows us to represent explicitly and to reason about the knowledge agents have of their own and other agents' strategies. We provide a semantics to ESL in terms of epistemic concurrent game models, then illustrate the expressive power of ESL as a specification language for games, both of perfect and imperfect information. Notably, we show that some fixed-point characterisations of operators for strategic abilities, which normally fail in contexts of imperfect information, can be recovered in a controlled way through the interplay of epistemic and strategy modalities.

1 Introduction

In recent years logic-based formalisms for representing and reasoning about strategic abilities, both individual and coalitional, have been a thriving area of research in Artificial Intelligence and Multi-agent System [4, 5, 12]. A diverse family of multi-modal logics has been introduced to provide a formal account of complex strategic reasoning and behaviours for individual agents and groups, including Alternating-time Temporal Logic (ATL), Strategy Logic, Coalition Logic to name just a few [2, 7, 26]. In parallel with these developments, a well-established tradition in knowledge representation focuses on extending formalisms for reactive systems with epistemic operators, so as to reason about the systems' evolution, as well

as the knowledge agents have thereof [9]. Seminal contributions on extensions of linear- and branching-time temporal logics with agent-indexed epistemic modalities date back to the '80s [13, 14]. Since then, these investigations have matured into a solid body of works, which is nowadays rightly regarded as a key contribution of formal methods to computer science [22], particularly when combined with verification techniques [10, 19, 21].

By bringing together these two lines of research, [28] introduced Alternating-time Temporal Epistemic Logic (ATEL), an extension of ATL with epistemic operators; thus paving the way for a wealth of contributions on sophisticated interactions between strategies and knowledge. More directly related to the present work, [3] and [15] put forward two variants of Epistemic Strategy Logic (ESL) with perfect knowledge, by building on the (non-epistemic) Strategy Logics in [7, 23]. These investigations are relevant from both a purely theoretical perspective and an applied viewpoint. Indeed, logics of time and knowledge have been successfully deployed in the verification by model checking of distributed and Multi-agent Systems as diverse as security protocols, UAVs, web services, and e-commerce [6, 10, 21, 19].

In this paper we advance the state-of-the-art in Epistemic Strategy Logic by exploring the theoretical properties of ESL in contexts of imperfect information. Specifically, we analyse and compare the expressive power of ESL as a specification language for games, both of perfect and imperfect information. It is well-known that in logics of strategies the latter assumption easily leads to higher complexity and even undecidability of the satisfiability and model checking problems [4, 5, 27]. This is due in part to the weaker semantic properties of strategies under imperfect information. In this paper we describe these features both

*This paper improves a previous version accepted for presentation at the 8th Workshop on Logical Aspects of Multi-Agent Systems (LAMAS2015).

syntactically and semantically, and show how some fixed-point characterisations of operators for strategic abilities, which normally fail in contexts of imperfect information, can be partially recovered through the interaction of epistemic and strategy modalities. These results might point to the development of decision procedures for the cases in hand [11].

Related Work. The present contribution draws inspiration from the extensive literature on logics of strategic abilities, especially Strategy Logic (SL). This formalism has been introduced in [7] for concurrent game structures (CGS) with two-players. In [23] CGS have been extended to a multi-player setting and bind operators for strategies have been introduced in the syntax. More recently, various notions of strategy (e.g., *behavioural* [24, 25]) have been analysed within SL. Hereafter we adopt multi-agent CGS in line with [23]. However, by drawing on the *Interpreted Systems* semantics for temporal epistemic logic [9], we provide an agent-based semantics to ESL, in which agents have possibly different actions and protocols, in contrast with [23, 24, 25].

To the best of our knowledge, epistemic extensions of SL have been recently considered in [3, 6, 15]; however always in contexts of perfect information. In particular, [6] describes MCMAS-SLK, a tool to model check CGS against specifications in an epistemic extension of SL, containing also binding operators. We extend the state-of-the-art w.r.t. [3, 6, 15] by analysing ESL with imperfect information, which has not yet been attempted. We will see that this endeavour requires several syntactical and semantical innovations, and it provides interesting new insight on the epistemic dimension of strategies.

Besides Strategy Logic, the interaction between knowledge and strategies has already been thoroughly studied within ATL. The wealth of contributions on the subject is testament to the interest and relevance of the area, of which it is impossible to give an exhaustive overview. Hereafter we briefly consider the works most directly related to the present contribution. The alternating-time temporal logic ATEL was introduced in [28], and immediately imperfect information variants were considered in [17], which put forward alternating-time temporal observational logic (ATOL) and ATEL-R*, as well as the key notion of *uniform strategy*. Interestingly, [17] discusses the distinction between *de re* and *de dicto* knowledge of strategies, but a formal account is not developed explicitly. The same distinction will be considered later on in the context of ESL with imperfect information. Further, in [16] ATL is enriched with a constructive notion of knowledge. As regards (non-epistemic) ATL, more elaborate notions of strategy have also appeared:

[1] introduces commitment in strategies; while [18] defines a notion of “feasible” strategy. In future work it might be worth exploring to what extent, if any, the theoretical results available for the various flavours of ATEL transfer to ESL.

Scheme of the paper. In Section 2 we introduce the epistemic concurrent game models (ECGM), which are used in Section 3 to provide a semantics to Epistemic Strategy Logic (ESL). We show that ECGM are suitable to model games in normal form, while properties such as the existence of Nash equilibria, and knowledge thereof, can be specified in ESL. In Section 4 we analyse ESL in contexts of imperfect information. In particular, we show how some characterisations of SL operators can be recast by combining epistemic and strategy modalities. Finally, in Section 5 we discuss the results obtained and point to future research. For reasons of space, only sketches of proofs are provided.

2 Epistemic Concurrent Game Models

This section is devoted to introducing epistemic concurrent game models (ECGM) and related notions [3], starting from the definition of *agent*.

Definition 1 (Agent) *An agent is a tuple $a = \langle L_a, Act_a, Pr_a \rangle$ such that*

- L_a is the set of local states l_a, l'_a, \dots ;
- Act_a is a finite set of actions $\sigma_a, \sigma'_a, \dots$;
- $Pr_a : L_a \mapsto 2^{Act_a}$ is the protocol function.

An agent a can be thought of as situated in some local state $l_a \in L_a$, that represents the information she has access to, and as performing the actions in Act_a according to the protocol Pr_a [9]. Differently from [23], hereafter agents might have different actions and protocols. Given a set $Ag = \{a_0, \dots, a_n\}$ of agents, we define the set \mathcal{G} of *global states* s, s', \dots (resp. the set Act of *joint actions* σ, σ', \dots) as the cartesian product $\prod_{a \in Ag} L_a$ (resp. $\prod_{a \in Ag} Act_a$). We now introduce ECGM as the systems generated by the agents’ interactions. In what follows we fix a set AP of atomic propositions and denote the j th component of a tuple t as t_j or, equivalently, $t(j)$.

Definition 2 (ECGM) *Given a set $Ag = \{a_0, \dots, a_n\}$ of agents $a = \langle L_a, Act_a, Pr_a \rangle$, an epistemic concurrent game model is a tuple $\mathcal{P} = \langle Ag, I, \tau, \pi \rangle$ such that*

- $I \subseteq \mathcal{G}$ is the set of initial global states s_0 ;

- $\tau : \mathcal{G} \times Act \mapsto \mathcal{G}$ is the global transition function, where $\tau(s, \sigma)$ is defined iff $\sigma_a \in Pr_a(l_a)$ for every $a \in Ag$;
- $\pi : AP \rightarrow 2^{\mathcal{G}}$ is the interpretation function.

Intuitively, ECGM can be seen as an agent-based variant of concurrent game structures [2, 28]. An ECGM describes the evolution of a system from an initial state $s_0 \in I$, according to the transition function τ .

We now introduce some notation to be used in the following. The *transition relation* $s \rightarrow s'$ on states holds iff $\tau(s, \sigma) = s'$ for some $\sigma \in Act$. A *run* λ from a state s , or *s-run*, is an infinite sequence $s^0 \rightarrow s^1 \rightarrow \dots$, where $s^0 = s$. For $n, m \in \mathbb{N}$, with $n \leq m$, we define $\lambda(n) = s^n$ and $\lambda[n, m] = s^n, s^{n+1}, \dots, s^m$. A state s' is *reachable from* s iff $\lambda(i) = s'$ for some s -run λ and $i \geq 0$. We define \mathcal{S} as the set of states reachable from any initial state $s_0 \in I$. Further, let \sharp be a placeholder for arbitrary individual actions. Given a subset $A \subseteq Ag$ of agents, an *A-action* σ_A is an $|Ag|$ -tuple s.t. (i) $\sigma_A(a) \in Act_a$ for $a \in A$, and (ii) $\sigma_A(b) = \sharp$ for $b \notin A$. Then, Act_A is the set of all *A-actions* and $d_A(s) = \{\sigma_A \in Act_A \mid \text{for every } a \in A, \sigma_a \in Pr_a(l_a)\}$ is the set of all *A-actions* enabled at $s = \langle l_0, \dots, l_n \rangle$. A joint action σ *extends* an *A-action* σ_A , or $\sigma_A \sqsubseteq \sigma$, iff $\sigma(a) = \sigma_A(a)$ for all $a \in A$. The *outcome* $out(s, \sigma_A)$ of action σ_A at state s is the set of all states s' s.t. $\tau(s, \sigma) = s'$ for some joint action $\sigma \sqsupseteq \sigma_A$. Finally, two global states $s = \langle l_0, \dots, l_n \rangle$ and $s' = \langle l'_0, \dots, l'_n \rangle$ are *indistinguishable* for agent a , or $s \sim_a s'$, iff $l_a = l'_a$ [9]. Similarly, for a set A of agents, $s \sim_A s'$ iff $l_a = l'_a$ for every $a \in A$, i.e., $(s, s') \in \bigcap_{a \in A} \sim_a$.

To illustrate the modelling power of ECGM, we show how a relevant class of games can be modelled within this framework. We start with a game of perfect information to compare and contrast with imperfect information later on.

Example 1 *Epistemic concurrent game models are expressive enough to model games in normal form. As an example, we consider the classic version of the prisoner's dilemma [8], with players 1 and 2, who can either cooperate (C) or defect (D), and payoff ordering $\alpha > \beta > \gamma > \delta$. This can be represented in normal form as:*

		2	
		Cooperate	Defect
1	Cooperate	β, β	δ, α
	Defect	α, δ	γ, γ

The prisoner's dilemma can be modelled as an ECGM by considering agents $1 = \langle L_1, Act_1, Pr_1 \rangle$ and $2 = \langle L_2, Act_2, Pr_2 \rangle$ s.t. for $a \in \{1, 2\}$, (i) $L_a =$

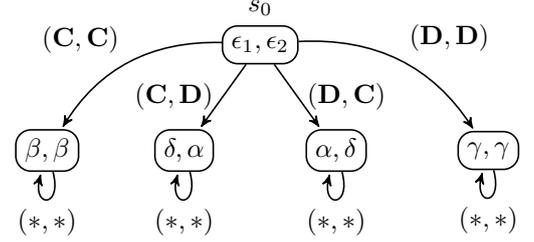


Figure 1: The ECGM \mathcal{P}_{pd} for the prisoner's dilemma.

$\{\epsilon_a, \alpha, \beta, \gamma, \delta\}$; (ii) $Act_a = \{C, D, *\}$; (iii) $Pr_a(\epsilon_a) = \{C, D\}$ and $Pr_a(\alpha) = Pr_a(\beta) = Pr_a(\gamma) = Pr_a(\delta) = \{*\}$, where $*$ represents the skip action. Also, for each payoff p we consider an atomic proposition p_a , which intuitively expresses that the local state of agent a is equal to the corresponding payoff. We then introduce the ECGM $\mathcal{P}_{pd} = \langle \{1, 2\}, \{s_0\}, \tau, \pi \rangle$ for the prisoner's dilemma, where (i) $s_0 = (\epsilon_1, \epsilon_2)$ is the initial state; (ii) the transition function τ is given as

- $\tau(s_0, (C, C)) = (\beta, \beta)$
- $\tau(s_0, (C, D)) = (\delta, \alpha)$
- $\tau(s_0, (D, C)) = (\alpha, \delta)$
- $\tau(s_0, (D, D)) = (\gamma, \gamma)$
- $\tau(s, (*, *)) = s$, for every s different from s_0

and (iii) $\pi(p_a) = \{s = \langle l_1, l_2 \rangle \mid l_a = p\}$. The ECGM \mathcal{P}_{pd} is depicted in Fig. 1. Notice that, for technical purposes, we assume that at the end of the game the agents loop on their final state.

3 Epistemic Strategy Logic

We now present an epistemic version of Strategy Logic, as a specification language for ECGM. First, for every set $A \subseteq Ag$ of agents, we introduce a set Var_A of strategy variables x_A, x'_A, \dots

Definition 3 (ESL) For $p \in AP$ and $A \subseteq Ag$, the ESL formulas ϕ are defined in BNF as follows:

$$\phi ::= p \mid \neg\phi \mid \phi \rightarrow \phi \mid X\phi \mid \phi U \phi \mid D_A\phi \mid \exists x_A\phi$$

The language ESL extends the Strategy Logic in [7] by adding the epistemic operator D_A for distributed knowledge in the set A of agents. Equivalently, ESL can be thought of as an epistemic extension of the Strategy Logic in [23], modulo the binding operator. The ESL formula $\exists x_A\phi$ is read as “the agents in A have a strategy to achieve ϕ ”. The interpretation of LTL operators X and U is standard. Observe that

epistemic formulas $K_a\phi$ for “agent a knows ϕ ”, can be defined as $D_{\{a\}}\phi$. Similarly, we write $\exists x_a\phi$ for $\exists x_{\{a\}}\phi$, i.e., “agent a has a strategy to achieve ϕ ”. The other propositional connectives, LTL operators G and F , and the universal strategy quantifier \forall can be defined as standard. Also, notice that the *nested-goal* fragment ESL[NG], the *boolean-goal* fragment ESL[BG], and the *one-goal* fragment ESL[1G] can be introduced in analogy to Strategy Logic [23]. Finally, the *free* variables $fr(\phi) \subseteq Ag$ of an ESL formula ϕ are inductively defined as follows:

$$\begin{aligned} fr(p) &= \emptyset \\ fr(\neg\phi) = fr(D_A\phi) &= fr(\phi) \\ fr(\phi \rightarrow \phi') &= fr(\phi) \cup fr(\phi') \\ fr(X\phi) = fr(\phi U \phi') &= Ag \\ fr(\exists x_A\phi) &= fr(\phi) \setminus A \end{aligned}$$

A *sentence* is an ESL formula ϕ with $fr(\phi) = \emptyset$.

We make use of ECGM to provide a semantics to ESL formulas, starting with the notion of strategy. We first consider the case of perfect information, as it provides us with a term of comparison with contexts of imperfect information. We will elaborate on their differences.

Definition 4 (Perfect Information Strategy)

Let γ be an ordinal s.t. $1 \leq \gamma \leq \omega$ and $A \subseteq Ag$ a set of agents. A γ -recall (perfect information) A -strategy is a function $f_A[\gamma] : \bigcup_{1 \leq n \leq \gamma} \mathcal{S}^n \mapsto Act_A$ s.t. $f_A[\gamma](\kappa) \in d_A(last(\kappa))$ for every $\kappa \in \bigcup_{1 \leq n \leq \gamma} \mathcal{S}^n$, where $last(\kappa)$ is the last element of κ .

Thus, a γ -recall A -strategy returns an enabled A -action for every sequence of states of length at most γ (possibly infinite). For $A = \{a\}$, $f_A[\gamma]$ can be seen as a function from $\bigcup_{1 \leq n \leq \gamma} \mathcal{S}^n$ to Act_a s.t. $f_A[\gamma](\kappa) \in Pr_a(last(\kappa)(a))$ for $\kappa \in \bigcup_{1 \leq n \leq \gamma} \mathcal{S}^n$. One key feature of perfect information contexts – which fails for the imperfect information semantics provided below – is that, for every set $A = \{a_0, \dots, a_m\}$ of agents, $f_A[\gamma]$ is tantamount to $f_{a_0}[\gamma] \times \dots \times f_{a_m}[\gamma]$, where for every $\kappa \in \bigcup_{1 \leq n \leq \gamma} \mathcal{S}^n$, $(f_{a_0}[\gamma] \times \dots \times f_{a_m}[\gamma])(\kappa)$ is defined as the set of actions $\sigma \in Act_A$ s.t. $\sigma_a = f_a[\gamma](\kappa)$ if $a \in A$, and $\sigma_a = \#$ otherwise. Therefore, whenever we assume perfect information, a group’s strategy is really the composition of its members’ strategies; we will observe that this is not always the case for imperfect information. Also, the *outcome* of strategy $f_A[\gamma]$ at state s , or $out(s, f_A[\gamma])$, is the set of all s -runs λ s.t. $\lambda(i+1) \in out(\lambda(i), f_A[\gamma](\lambda[j, i]))$ for all $i \geq 0$ and $j = \max(i - \gamma + 1, 0)$. With an abuse of notation we write $s' \in out(s, f_A[\gamma])$ to indicate that $s' \in \lambda(i)$ for some $\lambda \in out(s, f_A[\gamma])$ and $i \geq 0$. By varying γ we can define positional strategies ($\gamma = 1$), strategies with perfect recall ($\gamma = \omega$), etc. [12]. However, these different choices do not affect the following definitions and

results, so we assume that γ is fixed and omit it, unless specified otherwise. Finally, given an Ag -strategy f and an A -strategy g , let f_g^A denote the Ag -strategy s.t. for every $\kappa \in \bigcup_{1 \leq n \leq \gamma} \mathcal{S}^n$, $f_g^A(\kappa) = \sigma \in Act$, where $\sigma_A = g(\kappa)$ and $\sigma_{\bar{A}} = f(\kappa)$. Since $|out(s, f)| = 1$ for any Ag -strategy f , in the following we simply write $\lambda = out(s, f)$.

Definition 5 (Semantics of ESL) We define whether an ECGM \mathcal{P} satisfies a formula φ at state s according to context $V_0, \dots, V_n \subseteq \mathcal{S}$ and Ag -strategy f , or $(\mathcal{P}, s, \vec{V}, f) \models \varphi$, as follows (clauses for propositional connectives are straightforward and thus omitted):

$$\begin{aligned} (\mathcal{P}, s, \vec{V}, f) \models p &\text{ iff } s \in \pi(p) \\ (\mathcal{P}, s, \vec{V}, f) \models X\psi &\text{ iff for } \lambda = out(s, f), (\mathcal{P}, \lambda(1), \vec{V}, f) \models \psi \\ (\mathcal{P}, s, \vec{V}, f) \models \psi U \psi' &\text{ iff for } \lambda = out(s, f) \text{ and some } k \geq 0, \\ &\quad (\mathcal{P}, \lambda(k), \vec{V}, f) \models \psi', \text{ and for every } j, \\ &\quad 0 \leq j < k \text{ implies } (\mathcal{P}, \lambda(j), \vec{V}, f) \models \psi \\ (\mathcal{P}, s, \vec{V}, f) \models D_A\psi &\text{ iff for all } s' \in V_A, \\ &\quad s' \sim_A s \text{ implies } (\mathcal{P}, s', \vec{V}, f) \models \psi \\ (\mathcal{P}, s, \vec{V}, f) \models \exists x_A\psi &\text{ iff for some } A\text{-strategy } g, \\ &\quad (\mathcal{P}, s, \vec{V}, f_g^A) \models \psi \end{aligned}$$

where $V_A = \bigcap_{a \in A} V_a$.

An ESL formula φ is *true* at state s according to f , or $(\mathcal{P}, s, f) \models \varphi$, if $(\mathcal{P}, s, out(s, f_0), \dots, out(s, f_n), f) \models \varphi$; it is *true* at state s , or $(\mathcal{P}, s) \models \varphi$, if $(\mathcal{P}, s, f) \models \varphi$ for all Ag -strategies f ; it is *true* in \mathcal{P} , or $\mathcal{P} \models \varphi$, if $(\mathcal{P}, s_0) \models \varphi$ for all $s_0 \in I$; and it is *valid*, or $\models \varphi$, if φ is true in every ECGM. We remark that the satisfaction of formulas is independent from bound variables, that is, if for every $\kappa \in \bigcup_{1 \leq n \leq \gamma} \mathcal{S}^n$, strategies f and f' return the same $fr(\phi)$ -actions, i.e., $(f(\kappa))_{fr(\phi)} = (f'(\kappa))_{fr(\phi)}$, then $(\mathcal{P}, s, f) \models \phi$ iff $(\mathcal{P}, s, f') \models \phi$. In particular, the satisfaction of sentences is independent from strategies.

Hereafter we analyse the expressiveness of ESL. First of all, notice that the satisfaction of epistemic formulas depends on each set $V_a \subseteq \mathcal{S}$ of alternative states. Typically, in the literature each V_a is taken as the set \mathcal{S} of accessible states [9, 21], as only these are considered as actual epistemic alternatives. An agent won’t think possible to be situated in an unaccessible state $s \in \mathcal{G} \setminus \mathcal{S}$. However, by coherently developing this line of reasoning, in each state s only the accessible states in $out(s, f_a)$ may represent actual epistemic alternatives for agent a . Hence, by Def. 5 $(\mathcal{P}, s, f) \models D_A\psi$ iff for all accessible $s' \in \bigcap_{a \in A} out(s, f_a)$, $s' \sim_A s$ implies $(\mathcal{P}, s', out(s, f_0), \dots, out(s, f_n), f) \models \psi$. That is, the notion of knowledge defined on ECGM can be sum up as indistinguishability given the current state of the system and strategies. We also remark that according

to Def. 5, the operator D_A is indeed an $S5$ modality, and it satisfies all standard properties of distributed knowledge [9, 22]. Moreover, it shows some nice features when interacting with strategy modalities (see Theorem 4 below).

By the discussion above and definition of $f_{a_0} \times \dots \times f_{a_m}$, we can derive that variables for group strategies are actually redundant in ESL with perfect information. Intuitively, there is a strategy for a group $A = \{a_0, \dots, a_m\}$ of agents iff there are strategies for each agent $a \in A$. As a result, each formula $\exists x_A \phi$ (resp. $\forall x_A \phi$) quantifying on strategy variable x_A , can be rewritten as $\exists x_{a_0} \dots \exists x_{a_m} \phi$ (resp. $\forall x_{a_0} \dots \forall x_{a_m} \phi$). We state this fact formally.

Remark 1 For every ESL formula ϕ and $A = \{a_0, \dots, a_m\}$,

$$\models \exists x_A \phi \leftrightarrow \exists x_{a_0} \dots \exists x_{a_m} \phi \quad (1)$$

We will see that this represents a major difference w.r.t. contexts of imperfect information, where (1) does not normally hold.

Example 2 It is well-known that ATL modalities are expressible in (non-epistemic) Strategy Logic. Intuitively, an ATL formula $\langle\langle A \rangle\rangle \phi$ can be translated into ESL as $\exists x_A \forall x_{\bar{A}} \phi$, where, as discussed above, $\exists x_A$ is actually a shorthand for $\exists x_{a_0} \dots \exists x_{a_m}$ whenever $A = \{a_0, \dots, a_m\}$, and $\bar{A} = Ag \setminus A$.

As an example of the expressive power of ESL, we consider the following equivalences, which provide fixed-point characterisations of formulas $\theta_1 ::= \exists x_A \forall x_{\bar{A}} G\phi$, $\theta_2 ::= \exists x_A \forall x_{\bar{A}} F\phi$, and $\theta_3 ::= \exists x_A \forall x_{\bar{A}} (\phi U \phi')$, where ϕ and ϕ' are sentences:

$$\models \theta_1 \leftrightarrow \phi \wedge \exists x_A \forall x_{\bar{A}} X\theta_1 \quad (2)$$

$$\models \theta_2 \leftrightarrow \phi \vee \exists x_A \forall x_{\bar{A}} X\theta_2 \quad (3)$$

$$\models \theta_3 \leftrightarrow \phi' \vee (\phi \wedge \exists x_A \forall x_{\bar{A}} X\theta_3) \quad (4)$$

Formulas (2)-(4) are validities in the perfect information semantics. However, we will see that this is no longer the case in contexts of imperfect information.

Example 3 Strategy Logic is known to be expressive enough to specify Nash equilibria [7, 23]. Specifically, given the payoff ordering $\alpha > \beta > \gamma > \delta$ of the prisoner's dilemma in Example 1, we define the formula ψ_{NE} as follows:

$$\begin{aligned} \psi_{NE} ::= & \bigwedge_{a=1}^2 ((\exists y_a X\alpha_a \rightarrow X\alpha_a) \wedge \\ & (\neg \exists y_a X\alpha_a \wedge \exists y_a X\beta_a \rightarrow X\beta_a) \wedge \\ & (\neg \exists y_a X\alpha_a \wedge \neg \exists y_a X\beta_a \wedge \exists y_a X\gamma_a \rightarrow X\gamma_a) \wedge \\ & (\neg \exists y_a X\alpha_a \wedge \neg \exists y_a X\beta_a \wedge \neg \exists y_a X\gamma_a \wedge \exists y_a X\delta_a \rightarrow X\delta_a)) \end{aligned}$$

where by the definition of ECGM \mathcal{P}_{pd} , each atom $p_a \in \{\alpha_a, \beta_a, \gamma_a, \delta_a\}$ is true at state s iff the local state of agent a is equal to payoff p . Also, $\text{fr}(\psi_{NE}) = \{1, 2\}$.

Then, we can check that for the prisoner's dilemma ECGM \mathcal{P}_{pd} , $(\mathcal{P}_{pd}, s_0, f) \models \psi_{NE}$ iff the strategy profile $f(s_0)$ is a Nash equilibrium. In general, given a game in normal form with n players and payoff ordering $\alpha^1 > \dots > \alpha^k$, we define the SL formula ψ_{NE} characterizing Nash equilibria as follows:

$$\psi_{NE} ::= \bigwedge_{a=1}^n \bigwedge_{i=1}^k \left(\left(\bigwedge_{j=1}^{i-1} \neg \exists y_a X\alpha_a^j \right) \wedge \exists y_a X\alpha_a^i \rightarrow X\alpha_a^i \right)$$

Furthermore, the SL formula $\exists x_{Ag} \psi_{NE}$ expresses the existence of Nash equilibria. In particular, we can check that for $a \in \{1, 2\}$, $\mathcal{P}_{pd} \models K_a \exists x_1 \exists x_2 \psi_{NE}$, that is, each player knows that the prisoner's dilemma admits Nash equilibria. More specifically, for $a \in \{1, 2\}$, $\mathcal{P}_{pd} \models \exists x_1 \exists x_2 K_a \psi_{NE}$, i.e., every agent knows what the Nash equilibrium actually is for the game. By a closer comparison of ESL formulas $K_a \exists x_{Ag} \psi_{NE}$ and $\exists x_{Ag} K_a \psi_{NE}$ we can see that the former expresses de dicto knowledge of strategies, while the latter asserts knowledge de re: the players not only know that there is some strategy that is a Nash equilibrium, but they are actually capable of pointing it out. This is indeed the case as we assume that our players are perfect rational reasoners, and the prisoner's dilemma is a game of perfect information. In the following section we will see that in contexts of imperfect information it is not always possible to infer knowledge de re from de dicto knowledge.

4 ESL with Imperfect Information

In Section 3 we observed that the notion of satisfaction provided in Def. 5 assumes that agents have perfect information about the state they are situated in. Specifically, their strategies are determined by the system's global state. However, in many use cases of interest agents might have access to only a partial view of the system, as represented in their local state. Along this line, we introduce an imperfect information semantics for ESL, starting with some formal definitions. First, two histories $\kappa, \kappa' \in \mathcal{S}^n$ are indistinguishable for a set A of agents, or $\kappa \sim_A \kappa'$, iff for every $m \leq n$, $\kappa(m) \sim_A \kappa'(m)$.

We adapt ideas in [17] to define a notion of strategy suitable for contexts of imperfect information.

Definition 6 (Uniform Strategies) Let $A \subseteq Ag$ be a set of agents. A γ -recall uniform A -strategy is a γ -recall A -strategy $f_A^u[\gamma] : \bigcup_{1 \leq n \leq \gamma} \mathcal{S}^n \mapsto \text{Act}_A$ s.t. for all

histories κ and κ' in \mathcal{S}^n , if $\kappa \sim_A \kappa'$ then $f_A^u[\gamma](\kappa) = f_A^u[\gamma](\kappa')$.

In the case of a single agent a , a γ -recall uniform a -strategy can be seen as a mapping $f_a^u[\gamma] : \bigcup_{1 \leq n \leq \gamma} L_a^n \mapsto Act_a$ s.t. $f_a^u[\gamma](\kappa) \in Pr_a(last(\kappa))$ for every $\kappa \in \bigcup_{1 \leq n \leq \gamma} L_a^n$. That is, uniform strategies are functions from sequences of local states, rather than global states. This accounts for the “locality” of strategies.

We can now introduce a corresponding notion of satisfaction. In the literature on logics of strategic abilities we can identify two main variants of the satisfaction relation, depending on whether a strategy has to be successful for all undistinguishable states. As regards ATL, [27] assumes this to be the case, while [2] relaxes this constraint. Both alternatives are analysed in [17]. Hereafter we follow [2] and define a satisfaction relation \models^i for imperfect information. Then we show how the relation \models_S^i , formalising the account put forward in [27], can be represented in terms of \models^i . Formally, for an ECGM \mathcal{P} , an ESL formula φ , a state s , a context $\vec{V} \subseteq \mathcal{S}$, and a uniform Ag -strategy f , all clauses for the relation \models^i (resp. \models_S^i) are the same as in Def. 5, but the interpretation of \exists -formulas is restricted to uniform strategies as follows:

$$\begin{aligned} (\mathcal{P}, s, \vec{V}, f) \models^i \exists x_A \phi & \text{ iff for some uniform } A\text{-strategy } g, \\ & (\mathcal{P}, s, \vec{V}, f_g^A) \models^i \phi \\ (\mathcal{P}, s, \vec{V}, f) \models_S^i \exists x_A \phi & \text{ iff for some uniform } A\text{-strategy } g, \\ & \text{ for all } s' \in \vec{V}, s' \sim_A s \text{ implies} \\ & (\mathcal{P}, s', \vec{V}, f_g^A) \models_S^i \phi \end{aligned}$$

It is apparent that the definition of \models_S^i has an epistemic flavour. Indeed, it has been shown that knowledge operators are definable already in terms of ATL modalities when the satisfaction relation \models_S^i is considered [5].

Next, we state a result on the relationship between these two notions of satisfaction. First, consider the translation function $\rho : \text{ESL} \mapsto \text{ESL}$ defined as:

$$\begin{aligned} \rho(p) &= p \\ \rho(\phi \star \phi') &= \rho(\phi) \star \rho(\phi') \\ \rho(\flat \phi) &= \flat \rho(\phi) \\ \rho(\exists x_A \phi) &= \exists x_A D_A \rho(\phi) \end{aligned}$$

where \star is any binary operator, while \flat is any unary operator different from \exists . Then, we can check that the two notions of satisfaction for imperfect information are related as described by the following lemma.

Lemma 1 For every ESL formula ϕ ,

$$(\mathcal{P}, s, f) \models_S^i \phi \text{ iff } (\mathcal{P}, s, f) \models^i \rho(\phi)$$

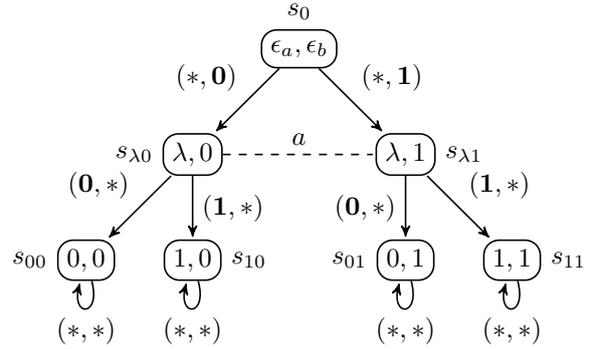


Figure 2: The ECGM \mathcal{Q} .

Sketch of Proof. The proof is by induction on the structure of ϕ . The case of interest is for existential formulas. For $\phi \equiv \exists x_A \psi$, $(\mathcal{P}, s, \vec{V}, f) \models_S^i \phi$ iff for some uniform A -strategy g , for all $s' \in \bigcap_{a \in A} V_a$, $s' \sim_A s$ implies $(\mathcal{P}, s', \vec{V}, f_g^A) \models_S^i \psi$. By induction hypothesis, $(\mathcal{P}, s', \vec{V}, f_g^A) \models^i \rho(\psi)$. Further, for all $s' \in \bigcap_{a \in A} V_a$, $s' \sim_A s$ implies $(\mathcal{P}, s', \vec{V}, f_g^A) \models^i \rho(\psi)$ iff $(\mathcal{P}, s, \vec{V}, f_g^A) \models^i D_A \rho(\psi)$, iff $(\mathcal{P}, s, \vec{V}, f) \models^i \exists x_A D_A \rho(\psi)$, i.e., $(\mathcal{P}, s, \vec{V}, f) \models^i \rho(\phi)$. In particular, the above holds for $\vec{V} = out(s, f_0), \dots, out(s, f_n)$. \square

As a result, the strategy operator \exists interpreted according to \models_S^i can be simulated in \models^i by using epistemic modalities in a *de re* manner. Lemma 1 provides one more example of the expressiveness of Epistemic Strategy Logic.

Another noteworthy result is that formulas (2)-(4) are no longer valid in the semantics of imperfect information.

Lemma 2 For every ESL formula ϕ ,

$$\not\models^i \phi \wedge \exists x_A \forall x_{\bar{A}} X \theta_1 \rightarrow \theta_1 \quad (5)$$

$$\not\models^i \phi \vee \exists x_A \forall x_{\bar{A}} X \theta_2 \rightarrow \theta_2 \quad (6)$$

$$\not\models^i \phi' \vee (\phi \wedge \exists x_A \forall x_{\bar{A}} X \theta_3) \rightarrow \theta_3 \quad (7)$$

Sketch of Proof. We illustrate the case for (6) by considering the ECGM \mathcal{Q} depicted in Fig. 2.

In this game player b chooses secretly between values 0 and 1. Then, at the successive stage, a also chooses between 0 and 1. The game is won by a if the values provided by the two players coincide, otherwise b wins. To model this game as an ECGM we introduce agents $a = \langle L_a, Act_a, Pr_a \rangle$ and $b = \langle L_b, Act_b, Pr_b \rangle$ defined as

- $L_a = \{\epsilon_a, \lambda, 0, 1\}$ and $L_b = \{\epsilon_b, 0, 1\}$;
- $Act_a = Act_b = \{0, 1, *\}$;
- $Pr_a(\epsilon_a) = Pr_a(0) = Pr_a(1) = \{*\}$ and $Pr_a(\lambda) = \{0, 1\}$; while $Pr_b(\epsilon_b) = \{0, 1\}$, $Pr_b(0) = Pr_b(1) = \{*\}$.

Also, we consider the atomic proposition eq , which intuitively expresses that the two bits in the global state are defined and equal. We then introduce the ECGM $\mathcal{Q} = \langle \{a, b\}, \{s_0\}, \tau, \pi \rangle$ as

- $s_0 = (\epsilon_a, \epsilon_b)$ is the only initial state;
- the transition function τ is given as follows for $i, j \in \{0, 1\}$:
 - $\tau((\epsilon_a, \epsilon_b), (*, \mathbf{i})) = (\lambda, i)$
 - $\tau((\lambda, i), (\mathbf{j}, *)) = (j, i)$
 - $\tau((i, j), (*, *)) = (i, j)$
- $\pi(eq) = \{s_{00}, s_{11}\}$.

In Fig. 2 a dashed line stands for epistemic indistinguishability.

Now we check that $\mathcal{Q} \not\models (3)$. Indeed, it is the case that $(\mathcal{Q}, s_{\lambda 0}, out(s_0, \vec{f}), f) \models^i \exists x_a \forall x_b Feq$ whenever agent a does $\mathbf{0}$ at local state λ , or $g_a(\lambda) = \mathbf{0}$. Also, $(\mathcal{Q}, s_{\lambda 1}, out(s_0, \vec{f}), f) \models^i \exists x_a \forall x_b Feq$ whenever a does $\mathbf{1}$ at λ . Hence, $(\mathcal{Q}, s_0, f) \models^i \exists x_a \forall x_b X(\exists x_a \forall x_b Feq)$. However, it is not the case that $(\mathcal{Q}, s_0, f) \models^i \exists x_a \forall x_b Feq$ if we only consider uniform strategies, as agent a should perform different actions in states $s_{\lambda 0}$ and $s_{\lambda 1}$ that she is not able to distinguish. \square

Furthermore, we remark that in each state $s \in \{s_{\lambda 0}, s_{\lambda 1}\}$ agent a has *de dicto* knowledge of a uniform strategy to achieve Feq , that is, $(\mathcal{Q}, s, out(s_0, \vec{f}), f) \models^i K_a \exists x_a \forall x_b Feq$. However, she has no *de re* knowledge of such strategy, as it is the case that $(\mathcal{Q}, s, out(s_0, \vec{f}), f) \not\models^i \exists x_a K_a \forall x_b Feq$. Hence, in marked contrast with perfect information, in contexts of imperfect information *de dicto* knowledge of strategies does not imply knowledge *de re* in general.

Now observe that for the satisfaction relation \models_S^i the converse of (2)-(4) fail.

Lemma 3 For every ESL formula ϕ ,

$$\not\models_S^i \theta_1 \rightarrow \phi \wedge \exists x_A \forall x_{\bar{A}} X \theta_1 \quad (8)$$

$$\not\models_S^i \theta_2 \rightarrow \phi \vee \exists x_A \forall x_{\bar{A}} X \theta_2 \quad (9)$$

$$\not\models_S^i \theta_3 \rightarrow \phi' \vee (\phi \wedge \exists x_A \forall x_{\bar{A}} X \theta_3) \quad (10)$$

Sketch of Proof. We provide a witness for (9) by means of the ECGM \mathcal{Q}' in Fig. 3(a). Formally, we introduce agents $a' = \langle L_{a'}, Act_{a'}, Pr_{a'} \rangle$ and $b' = \langle L_{b'}, Act_{b'}, Pr_{b'} \rangle$ s.t. (i) $L_{a'} = L_a$ and $L_{b'} = L_b$; (ii) $Act_{a'} = Act_a$ and $Act_{b'} = \{*\}$; and (iii) $Pr_{a'}(\epsilon_a) = \{\mathbf{0}, \mathbf{1}\}$, and $Pr_{a'}(\lambda) = Pr_{a'}(\mathbf{0}) = Pr_{a'}(\mathbf{1}) = \{*\}$; while $Pr_{b'}(\epsilon_b) = Pr_{b'}(\mathbf{0}) = Pr_{b'}(\mathbf{1}) = \{*\}$. Then, the ECGM $\mathcal{Q}' = \langle \{a', b'\}, \{s'_0\}, \tau', \pi' \rangle$ is specified as (i) $s'_0 = s_0 = (\epsilon_a, \epsilon_b)$; (ii) the transition function τ' is defined as in Fig. 3(a); and (iii) $\pi'(eq) = \{s'_{00}\}$. Also

in Fig. 3(a) a dashed line denotes epistemic indistinguishability.

We show that $\mathcal{Q}' \not\models (3)$. First, notice that $(\mathcal{Q}', s'_0, out(s_0, \vec{f}), f) \models_S^i \exists x_{a'} \forall x_{b'} Feq$ by using the uniform strategy $g_{a'}$ s.t. $g_{a'}(s'_0) = \mathbf{0}$ and $g_{a'}(s'_{\lambda 0}) = *$. However, $(\mathcal{Q}', s'_{\lambda 0}, out(s_0, \vec{f}), f) \not\models_S^i \exists x_{a'} \forall x_{b'} Feq$, as there is no uniform a' -strategy g' s.t. $(\mathcal{Q}', s'_{\lambda 1}, out(s_0, \vec{f}), f_{g'}) \models_S^i \forall x_{b'} Feq$. Hence, $(\mathcal{Q}', s'_0, f) \not\models_S^i \exists x_{a'} \forall x_{b'} X \theta_2$. \square

We now show that, differently from perfect information and Remark 1, in contexts of imperfect information an ESL formula $\exists x_A \phi$, for $A = \{a_0, \dots, a_m\}$, is not equivalent to $\exists a_0 \dots \exists a_m \phi$. In particular, while $\exists a_0 \dots \exists a_m \phi \rightarrow \exists x_A \phi$ is valid, as the existence of a uniform strategy for each agent $a \in A$ implies the existence of a uniform strategy for group A , the converse does not hold. To see this, consider the modification \mathcal{Q}'' of the ECGM \mathcal{Q} above, depicted in Fig. 3(b), which includes also an agent c , who has no strategic power but complete information on the current state¹. Clearly, in s'_0 the group $A = \{a, c\}$ of agents has a uniform A -strategy g to achieve Feq , that is, $(\mathcal{Q}, s_0, f) \models^i \exists x_A \forall x_b Feq$ for $g(s_0) = (*, \#, *)$, $g(s_{\lambda 0}) = (\mathbf{0}, \#, *)$, and $g(s_{\lambda 1}) = (\mathbf{1}, \#, *)$. However, agent a alone has no uniform strategy to achieve $\exists x_c \forall x_b Feq$ in s'_0 , as states $s'_{\lambda 0}$ and $s'_{\lambda 1}$ are still undistinguishable to a . Hence, $(\mathcal{Q}, s_0, f) \not\models^i \exists x_a \exists x_c \forall x_b Feq$. As a consequence, variables for group strategies cannot be dispensed with in contexts of imperfect information.

Finally, we show the main result of this paper, namely the fixed-point characterisations (2)-(4) of strategy operators can be partially recovered in ESL. Specifically, this result is achieved by combining epistemic and strategy modalities as stated in the following theorem.

Theorem 4 Recall that $\theta_1 ::= \exists x_A \forall x_{\bar{A}} G \phi$, $\theta_2 ::= \exists x_A \forall x_{\bar{A}} F \phi$, and $\theta_3 ::= \exists x_A \forall x_{\bar{A}} (\phi U \phi')$. If we consider positional strategies ($\gamma = 1$), then

$$\models^i \theta_1 \leftrightarrow \phi \wedge$$

$$\exists x_A \forall x_{\bar{A}} X (\exists x_A \forall x_{\bar{A}} (G \phi \wedge (D_A \theta_1 \rightarrow \exists x_A \forall x_{\bar{A}} D_A G \phi))) \quad (11)$$

$$\models^i \theta_2 \leftrightarrow \phi \vee$$

$$\exists x_A \forall x_{\bar{A}} X (\exists x_A \forall x_{\bar{A}} (F \phi \wedge (D_A \theta_2 \rightarrow \exists x_A \forall x_{\bar{A}} D_A F \phi))) \quad (12)$$

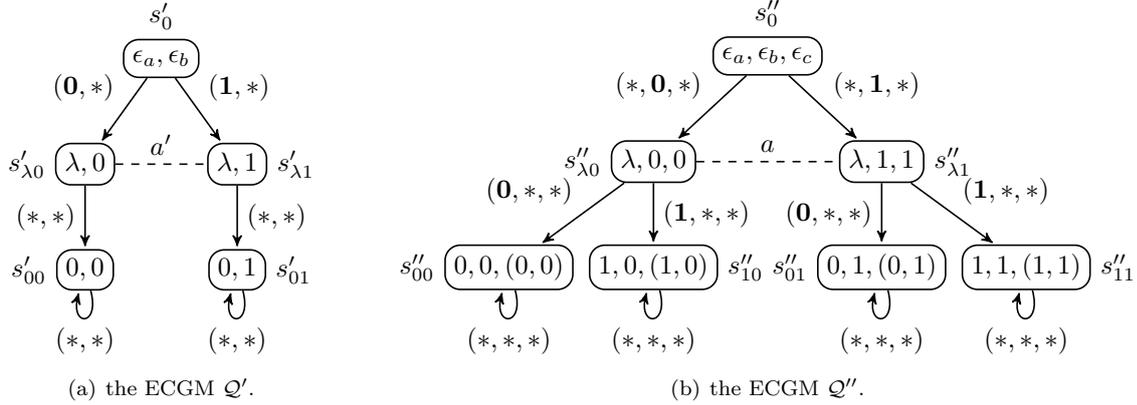
$$\models^i \theta_3 \leftrightarrow \phi' \vee (\phi \wedge$$

$$\exists x_A \forall x_{\bar{A}} X (\exists x_A \forall x_{\bar{A}} (\phi U \phi' \wedge (D_A \theta_3 \rightarrow \exists x_A \forall x_{\bar{A}} D_A \phi U \phi')))) \quad (13)$$

Sketch of Proof. We briefly show (12), the proof for the other cases is similar.

\Rightarrow We can prove that $(\mathcal{P}, s, f) \models \theta_2 \rightarrow (\neg \phi \rightarrow \exists x_A \forall x_{\bar{A}} X \theta_2)$ as in Example 2. Further, $(\mathcal{P}, s, f) \models$

¹We thank an anonymous reviewer at ECAI14 for pointing out this model.



$D_A\theta_2 \rightarrow \exists x_A \forall x_{\bar{A}} D_A F\phi$ as we are using uniform strategies.

\Leftarrow Suppose that $(\mathcal{P}, s, f) \models \exists x_A \forall x_{\bar{A}} X(\exists x_A \forall x_{\bar{A}} (F\phi \wedge (D_A\theta_2 \rightarrow \exists x_A \forall x_{\bar{A}} D_A F\phi)))$, that is, for some uniform A -strategy g , for all uniform \bar{A} -strategy h and $\lambda = out(s, f_{g,h}^{A,\bar{A}})$, $(\mathcal{P}, \lambda(1), out(s, \vec{f}), f_{g,h}^{A,\bar{A}}) \models \exists x_A \forall x_{\bar{A}} (F\phi \wedge (D_A\theta_2 \rightarrow \exists x_A \forall x_{\bar{A}} D_A F\phi))$, i.e., for some uniform A -strategy g' , for all uniform \bar{A} -strategy h' and $\lambda' = out(\lambda(1), f_{g',h'}^{A,\bar{A}})$, $(\mathcal{P}, \lambda'(1), out(s, \vec{f}), f_{g',h'}^{A,\bar{A}}) \models F\phi \wedge (D_A\theta_2 \rightarrow \exists x_A \forall x_{\bar{A}} D_A F\phi)$. Now we have to show that A -strategies g and g' can be composed so as to obtain a uniform A -strategy g'' witnessing $(\mathcal{P}, s, f) \models \exists x_A \forall x_{\bar{A}} F\phi$. To this end, we notice that $(\mathcal{P}, \lambda'(1), out(s, \vec{f}), f_{g',h'}^{A,\bar{A}}) \models D_A\theta_2 \rightarrow \exists x_A \forall x_{\bar{A}} D_A F\phi$, that is, for all $s' \in \bigcap_{a \in A} out(s, f_a)$, $s' \sim_A \lambda'(1)$ implies $(\mathcal{P}, s', out(s, \vec{f}), f_{g',h'}^{A,\bar{A}}) \models \exists x_A \forall x_{\bar{A}} F\phi \rightarrow \exists x_A \forall x_{\bar{A}} D_A F\phi$, i.e., the strategy $f_{g',h'}^{A,\bar{A}}$ to achieve $F\phi$ does not depend on the particular $s' \in \bigcap_{a \in A} out(s, f_a)$ s.t. $s' \sim_A \lambda'(1)$ (since $f_{g',h'}^{A,\bar{A}}$ is positional). As a consequence, strategies g and g' can actually be composed into g'' similarly as in Example 2, thus obtaining the desired result. \square

We remark that the conjunct $D_A\theta_a \rightarrow \exists x_A \forall x_{\bar{A}} D_A \Box(\phi, \phi')$ in the RHS of (11)-(13), for $\Box \in \{G, F, U\}$, guarantees that we can derive *de re* knowledge of strategies from *de dicto* knowledge, which in turn ensures the existence of uniform strategies; hence ruling out counterexamples such as the ECGM \mathcal{Q} in Fig. 2. Observe that (11)-(13) are not really fixed-points as θ -formulas appear negatively on the RHS. Nonetheless, the interest of validities such as (11)-(13) lies in the fact that similar characterisations feature prominently in decision procedures for the satisfiability and model checking problems of temporal logics [11, 20]. We envisage that (11)-(13) may play a similar role w.r.t ESL with

imperfect information. From a more philosophical perspective, such equivalences shed light on the meaning of strategies operators. For instance, according to (12), a group A of agents has a (uniform) strategy to eventually achieve ϕ (irrespective of the other agents' behaviour) iff either ϕ holds or at the next step they have a strategy g to eventually achieve ϕ , and know that if they can eventually achieve ϕ , then they can do so by playing g .

As a special case, for a single agent a we have the following equivalences, where $\theta_4 ::= \exists x_a \forall x_{\bar{a}} G\phi$, $\theta_5 ::= \exists x_a \forall x_{\bar{a}} F\phi$, and $\theta_6 ::= \exists x_a \forall x_{\bar{a}} (\phi U \phi')$ for $\bar{a} = Ag \setminus \{a\}$, and distributed knowledge is replaced with individual knowledge.

$$\begin{aligned} & \models^i \theta_4 \leftrightarrow \phi \wedge \exists x_a \forall x_{\bar{a}} X(\exists x_a \forall x_{\bar{a}} (G\phi \wedge (K_a\theta_4 \rightarrow \exists x_a \forall x_{\bar{a}} K_a G\phi))) \\ & \models^i \theta_5 \leftrightarrow \phi \vee \exists x_a \forall x_{\bar{a}} X(\exists x_a \forall x_{\bar{a}} (F\phi \wedge (K_a\theta_5 \rightarrow \exists x_a \forall x_{\bar{a}} K_a F\phi))) \\ & \models^i \theta_6 \leftrightarrow \phi' \vee (\phi \wedge \\ & \quad \exists x_a \forall x_{\bar{a}} X(\exists x_a \forall x_{\bar{a}} (\phi U \phi') \wedge (K_a\theta_6 \rightarrow \exists x_a \forall x_{\bar{a}} K_a \phi U \phi'))) \end{aligned}$$

We conclude by briefly discussing the modifications necessary to extend Theorem 4 to strategies defined on arbitrary γ . Without going into details, we just remark that by defining the relation of indistinguishability on histories, it is possible to introduce a perfect recall semantics for ESL, where the relation of satisfaction is defined on histories, rather than single states. Given this perfect recall version of the imperfect information semantics for ESL, we can prove Theorem 4 for arbitrary γ .

5 Conclusions

In this paper we analysed Epistemic Strategy Logic, an extension of Strategy Logic [7, 23] with epistemic operators for individuals and coalitions, in contexts of imperfect information. We considered uniform strategies within the semantics of epistemic concurrent game models (ECGM), and investigated the modelling and

expressive power of both ECGM and ESL. Specifically, we showed that a rich class of games can be modelled as ECGM, while the existence of Nash equilibria and knowledge thereof can be specified in ESL. We discussed some semantical features of Epistemic Strategy Logic. We showed that imperfect information allows to distinguish between *de dicto* and *de re* knowledge of strategies, while these are equivalent in contexts of perfect information. Notably, we proved that characterisations of ATL operators – even though not fixed-points – can be recovered in contexts of imperfect information through the interplay of epistemic and strategy operators. These results might point to future developments w.r.t. the satisfiability and model checking problems.

A number of extensions of the proposed framework are envisaged. Firstly, the nested-goal, boolean-goal, and one-goal fragment of SL are known to have better computational properties than full Strategy Logic [23]. It is likely that the corresponding fragments of ESL enjoy similar qualities. Secondly, ESL can be extended with further modalities for group knowledge, e.g. common knowledge, for enhanced expressiveness. Thirdly, various assumptions can be imposed on ECGM, for instance perfect recall, no learning, and synchronicity. These extensions, while enhancing the modelling power of ECGM, are also likely to increase their computational complexity.

Acknowledgements: we would like to thank the LAMAS reviewers for useful comments on a previous version of the paper.

References

- [1] Thomas Agotnes, Valentin Goranko, and Wojciech Jamroga. Alternating-time temporal logics with irrevocable strategies. In *Proceedings of the 11th Conference on Theoretical Aspects of Rationality and Knowledge*, TARK '07, pages 15–24, New York, NY, USA, 2007. ACM.
- [2] Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. Alternating-time temporal logic. *J. ACM*, 49(5):672–713, 2002.
- [3] Francesco Belardinelli. Reasoning about knowledge and strategies: Epistemic strategy logic. In *Proceedings of the 2nd International Workshop on Strategic Reasoning*, 2014.
- [4] N. Bulling and W. Jamroga. Comparing variants of strategic ability: how uncertainty and memory influence general properties of games. *Autonomous Agents and Multi-Agent Systems*, 28(3):474–518, 2014.
- [5] Nils Bulling, Jurgen Dix, and Wojciech Jamroga. Model checking logics of strategic ability: Complexity*. In Mehdi Dastani, Koen V. Hindriks, and John-Jules Charles Meyer, editors, *Specification and Verification of Multi-agent Systems*, pages 125–159. Springer US, 2010.
- [6] P. Cermák, A. Lomuscio, F. Mogavero, and A. Murano. MCMAS-SLK: A model checker for the verification of strategy logic specifications. In A. Biere and R. Bloem, editors, *Computer Aided Verification - 26th International Conference, CAV 2014, Held as Part of the Vienna Summer of Logic, VSL 2014, Vienna, Austria, July 18-22, 2014. Proceedings*, volume 8559 of *Lecture Notes in Computer Science*, pages 525–532. Springer, 2014.
- [7] Krishnendu Chatterjee, Thomas A. Henzinger, and Nir Piterman. Strategy logic. *Inf. Comput.*, 208(6):677–693, 2010.
- [8] D. Easley and J. Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, New York, NY, USA, 2010.
- [9] R. Fagin, J.Y. Halpern, Y. Moses, and M.Y. Vardi. *Reasoning About Knowledge*. The MIT Press, 1995.
- [10] Peter Gammie and Ron van der Meyden. Mck: Model checking the logic of knowledge. In Rajeev Alur and Doron Peled, editors, *CAV*, volume 3114 of *Lecture Notes in Computer Science*, pages 479–483. Springer, 2004.
- [11] V. Goranko and D. Shkatov. Tableau-based decision procedures for logics of strategic ability in multiagent systems. *ACM Trans. Comput. Log.*, 11(1), 2009.
- [12] Valentin Goranko and Wojciech Jamroga. Comparing semantics of logics for multi-agent systems. *Synthese*, 139(2):241–280, 2004.
- [13] Joseph Y. Halpern and Moshe Y. Vardi. The complexity of reasoning about knowledge and time: Extended abstract. In Juris Hartmanis, editor, *STOC*, pages 304–315. ACM, 1986.
- [14] Joseph Y. Halpern and Moshe Y. Vardi. The complexity of reasoning about knowledge and time. i. lower bounds. *J. Comput. Syst. Sci.*, 38(1):195–237, 1989.
- [15] Xiaowei Huang and Ron van der Meyden. An epistemic strategy logic. In *Proceedings of the 2nd*

- International Workshop on Strategic Reasoning*, 2014.
- [16] Wojciech Jamroga and Thomas Ågotnes. Constructive knowledge: what agents can achieve under imperfect information. *Journal of Applied Non-Classical Logics*, 17(4):423–475, 2007.
- [17] Wojciech Jamroga and Wiebe van der Hoek. Agents that know how to play. *Fundam. Inform.*, 63(2-3):185–219, 2004.
- [18] G. Jonker. Feasible strategies in alternating-time temporal epistemic logic. Master’s thesis, University of Utrecht, 2003.
- [19] M. Kacprzak, W. Nabialek, A. Niewiadomski, W. Penczek, A. Pólrola, M. Szreter, B. Wozna, and A. Zbrzezny. Verics 2007 - a model checker for knowledge and real-time. *Fundamenta Informaticae*, 85(1-4):313–328, 2008.
- [20] Orna Kupferman, Moshe Y. Vardi, and Pierre Wolper. An Automata-Theoretic Approach to Branching-Time Model Checking. *Journal of the ACM*, 47(2):312–360, 2000.
- [21] A. Lomuscio, H. Qu, and F. Raimondi. MCMAS: A model checker for the verification of multi-agent systems. In A. Bouajjani and O. Maler, editors, *CAV*, volume 5643 of *Lecture Notes in Computer Science*, pages 682–688. Springer, 2009.
- [22] J.-J. Ch. Meyer and W. van der Hoek. *Epistemic Logic for AI and Computer Science*, volume 41 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1995.
- [23] F. Mogavero, A. Murano, G. Perelli, and M. Vardi. Reasoning about strategies: On the model-checking problem. *ACM Trans. Comput. Log.*, 15(4):34, 2014.
- [24] F. Mogavero, A. Murano, and L. Sauro. On the boundary of behavioral strategies. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25-28, 2013*, pages 263–272. IEEE Computer Society, 2013.
- [25] F. Mogavero, A. Murano, and L. Sauro. Strategy games: a renewed framework. In A. Bazzan, M. Huhns, A. Lomuscio, and P. Scerri, editors, *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS ’14, Paris, France, May 5-9, 2014*, pages 869–876. IFAA-MAS/ACM, 2014.
- [26] Marc Pauly. A modal logic for coalitional power in games. *J. Log. Comput.*, 12(1):149–166, 2002.
- [27] Pierre-Yves Schobbens. Alternating-time logic with imperfect recall. *Electr. Notes Theor. Comput. Sci.*, 85(2):82–93, 2004.
- [28] W. van der Hoek and M. Wooldridge. Cooperation, knowledge, and time: Alternating-time temporal epistemic logic and its applications. *Studia Logica*, 75(1):125–157, 2003.

La logique facile avec TouIST

(formalisez et résolvez facilement des problèmes du monde réel)

Skander Ben Slimane², Alexis Comte², Olivier Gasquet^{1 2}, Abdelwahab Heba²,
Olivier Lezaud², Frédéric Maris^{1 2}, Maël Valais²

¹IRIT (Institut de Recherche en Informatique de Toulouse)

²Université Paul Sabatier

Toulouse, France

{gasquet,maris@irit.fr}

Résumé

Les solveurs SAT sont des outils puissants pour résoudre des problèmes logiques de taille réelle, mais leur utilisation nécessite des connaissances solides. Elle peut être vue par rapport à la logique comme l'utilisation d'un langage d'assemblage par rapport à la programmation. Il manque un langage de haut niveau pour permettre à des utilisateurs divers de tirer facilement profit de ces outils. TouIST vise à combler cette lacune. Il est dédié à la logique propositionnelle et ses principales fonctions sont (1) d'offrir un langage logique de haut niveau pour exprimer succinctement des formules complexes (par exemple des formules décrivant les règles du Sudoku, des problèmes de planification. . .) et (2) de trouver des modèles à ces formules en utilisant un solveur adéquat et performant, que l'utilisateur n'a pas besoin de connaître. Il consiste en une interface conviviale qui propose plusieurs facilités syntaxiques et qui fait appel à des solveurs suffisamment puissants pour permettre de résoudre automatiquement de grandes instances de problèmes difficiles (emplois du temps, Sudokus. . .). Il peut interagir avec différents démonstrateurs : solveurs SAT pur mais également solveurs SMT (SAT modulo théories - comme la théorie linéaire sur les réels, etc). Il peut donc être utilisé aussi bien par des débutants pour des problèmes purement propositionnels, que par des étudiants de cycles supérieurs ou même des chercheurs et ingénieurs, par exemple pour résoudre des problèmes de planification impliquant de grands ensembles d'actions et des contraintes numériques.

1 Historique

O. Gasquet et F. Maris enseignent à l'Université Paul Sabatier à Toulouse. Ils enseignent la logique à

différents niveaux, en démarrant des cours d'introduction à la logique propositionnelle, jusqu'à des sujets avancés pour les étudiants en Master, comme la logique modale ou la planification basée sur la logique. S. Ben Slimane, A. Comte, A. Heba, O. Lezaud et M. Valais sont des étudiants en Master de la même université. Ils ont mis en oeuvre TouIST durant les trois mois de leur projet de Master.

Motivation des étudiants

Au début des études de premier cycle, nous (enseignants) avons constaté que la motivation des étudiants peut être améliorée en leur montrant que la logique est très utile pour les informaticiens et que l'informatique ne consiste pas seulement à écrire du code C ou Java. Classiquement, la logique est motivée par des exemples abstraits ou, au mieux, par des exemples ludiques. A un moment, nous avons pensé qu'il serait préférable de leur montrer et pas seulement de leur dire qu'avec un peu de connaissance, la logique peut être utilisée pour résoudre des problèmes difficiles que la taille empêche de résoudre facilement à la main ou exigerait une programmation assez complexe en C ou tout autre langage de programmation.

Genèse de SATTOULOUSE

Lors de la conférence ICTTL'2011, nous avons présenté SATTOULOUSE [2], dédié à la logique propositionnelle, dont les principales fonctions étaient (1) d'offrir un langage logique de haut niveau pour exprimer

succinctement des formules complexes et (2) de trouver des modèles à ces formules en utilisant un solveur SAT performant.

Bien sûr, il existe de nombreux outils logiques comme des prouveurs, assistants de preuves, éditeurs de tables de vérité... sur Internet, même PROLOG aurait pu être utilisé, mais aucun ne correspond à nos exigences qui sont :

- l’outil doit être très facile à installer et à utiliser, sans syntaxe complexe ;
- le prouveur peut être utilisé comme une boîte noire sans savoir comment il fonctionne ;
- aucune mise en forme normale, aucun ordonnancement de clauses, ou aucune coupure PROLOG ne doivent être requis ;
- seulement une petite connaissance en logique devrait être nécessaire.

Comme nous ne pouvions pas trouver un outil existant satisfaisant ces exigences, en 2010, nous avons commencé à développer le nôtre, et nous sommes arrivés à l’idée de développer simplement une interface qui permet d’utiliser très confortablement un prouveur SAT (à savoir SAT4J [1]) : cet outil avait été appelé SATTOULOUSE et est décrit dans [2]. Avec cet outil, les étudiants pouvaient expérimenter par eux-même qu’un langage logique n’est pas seulement descriptif mais peut conduire à des calculs qui résolvent des problèmes de la vie réelle. En particulier, avec SATTOULOUSE, ils pouvaient résoudre des Sudokus assez facilement, ainsi que beaucoup d’autres problèmes combinatoires (emplois du temps, coloration de carte, circuits électroniques...).

Voici les principales facilités qu’offraient SATTOULOUSE :

- les formules entrées n’ont pas besoin d’être sous forme clausale et des connecteurs arbitraires peuvent être utilisés, la mise sous forme normale est faite dynamiquement pendant la saisie au clavier de l’utilisateur ;
- des facilités d’utilisation de grandes conjonctions ou disjonctions sont offertes comme dans :

$$\bigwedge_{i \in \{1..9\}} \bigvee_{j \in \{1..9\}} \bigwedge_{n \in \{1..9\}} \bigwedge_{m \in \{1..9\}, m \neq n} (p_{i,j,n} \rightarrow \neg p_{i,j,m})$$

- démarrer le solveur consiste à cliquer sur un bouton ;
- l’outil affiche un modèle dans la syntaxe de la formule entrée.

Ainsi, il est possible de montrer la puissance de la logique propositionnelle à des étudiants qui ont été formés quelques heures à la formalisation de phrases en logique et qui ont acquis les notions de bases de validité et satisfiabilité pour résoudre automatiquement des Sudokus.

Travaux pratiques avec SATTOULOUSE

Mais ce n’est pas toute l’histoire, puisque le même solveur SAT peut être utilisé pour résoudre de nombreux autres problèmes combinatoires aussi facilement que pour le Sudoku : ils suffisent juste de formaliser les contraintes. Nos étudiants sont invités à le faire pour : des emplois du temps, des colorations de cartes,... SATTOULOUSE a été utilisé pendant trois ans par environ 400 étudiants avec une grande satisfaction. En particulier, les étudiants l’ont utilisé pour effectuer des devoirs à long terme dans l’esprit de la programmation de projets : nous leur donnons un problème logique à résoudre (trop gros pour être résolu à la main), ils doivent le formaliser et ensuite utiliser cette formalisation pour le résoudre. Par exemple, un problème de stockage de produits chimiques qui doivent être stockés dans des salles identiques/contigües/non-contigües en fonction de leur degré de compatibilité. Les étudiants doivent résoudre un cas impliquant beaucoup de produits chimiques.

Limites de SATTOULOUSE et genèse de TouIST

Mais pendant ces années, nous avons remarqué quelques limitations dommageables de SATTOULOUSE : de nombreux bugs, des défauts dans l’interface, le manque de modularité (si l’on souhaite changer le prouveur SAT utilisé), l’ambiguïté et les limites de son langage, etc.

Par exemple, des problèmes impliquant des contraintes de cardinalité, comme les règles du jeu de Takuzu¹ qui nécessitent de compter des 0 et des 1, ne peuvent être facilement formalisés : le manque des fonctionnalités permettant d’exprimer des choses comme “exactement 5 parmi 10 propositions sont vraies”.

De plus, SATTOULOUSE n’offre pas la possibilité de parcourir l’ensemble des modèles fournis par le solveur, il en retourne seulement un.

Les leçons tirées de trois années d’utilisation de SATTOULOUSE sont que beaucoup de nos étudiants en informatique ont clairement pris conscience que la logique avait des applications réelles en ce qui concerne la résolution de problèmes, et beaucoup d’entre-eux ont acquis une capacité dans la formalisation de problèmes. Mais les défauts de SATTOULOUSE rendent le débogage vraiment difficile, d’une part parce qu’un seul modèle est affiché et en raison de la façon dont ce modèle est affiché, et d’autre part à cause des faibles capacités d’édition dont il dispose. En outre, seuls des problèmes combinatoires purs peuvent être traités, ce qui limite lourdement la prétention de résolution d’une

1. Connu aussi sous le nom de Binero.
<http://fr.wikipedia.org/wiki/Takuzu>

large gamme de problèmes par **SAToulouse** concernant les problèmes du monde réel.

Un autre inconvénient de **SAToulouse**, pas nécessairement lié à l’enseignement de la logique, est son incapacité à être utilisé à partir de la ligne de commande : des chercheurs ou des ingénieurs qui souhaiteraient l’utiliser intensivement trouveraient fastidieux de taper des problèmes en entrée.

Enfin, l’extension à des théories plus riches est également quelque-chose qui peut intéresser les chercheurs, les ingénieurs ou les étudiants de cycles supérieurs. **SAToulouse** n’est certainement pas adapté pour la satisfiabilité modulo théories ou pour résoudre des problèmes de planification alors que la même architecture logicielle pourrait être utilisée en changeant juste le solveur.

Il y a quelques mois, nous avons commencé à développer un nouveau logiciel qui comblerait ces lacunes et remplirait nos attentes. Nous l’avons appelé **TouIST** qui signifie **TOU**louse **I**ntegrated **S**atisfiability **T**ool et doit être prononcé “twist”. **TouIST** est bien sûr à la disposition du public pour téléchargement à partir du site suivant :

<https://github.com/olzd/touist/releases>

Pour résumer, voici quelques fonctionnalités offertes par **TouIST** et que **SAToulouse** ne propose pas :

- définition d’ensembles de domaines : $\bigwedge_{i \in A}$ vs. $\bigwedge_{i \in \{Paris, London, Roma, Madrid\}}$
- liaisons multiples sur les indices : $\bigwedge_{i \in A, j \in B}$ vs. $\bigwedge_{i \in \dots} \bigwedge_{j \in \dots}$
- calculs riches sur les indices ainsi que sur les ensembles de domaines : $\bigwedge_{i \in (A \cup (B \cap C))}$
- primitives de contraintes de cardinalité intégrées : “auMoins” (resp. “auPlus”, “exactement”) tant de valeurs sont vraies parmi ces valeurs
- les prédicats peuvent également être des variables définies sur des ensembles de domaines : $\bigwedge_{X \in \{A, B\}, i \in \{1, 2\}} X(i)$ vs. $\bigwedge_{i \in \{1, 2\}} (A(i) \wedge B(i))$
- littéraux spéciaux définissant des contraintes entre nombres entiers ou réels : $(x + y \leq z)$
- parcours facile des modèles successivement calculés par les solveurs
- expressions régulières permettant un filtrage des littéraux pertinents
- possibilité d’utiliser le logiciel en ligne de commande et/ou batch
- nombreuses fonctionnalités d’édition et améliorations

2 Vue d’ensemble de TouIST

TouIST est composé de trois modules, mais l’utilisateur standard ne verra que l’un d’entre eux : l’inter-

face. Dans la suite nous insistons principalement sur cette dernière plutôt que sur le traducteur et le solveur. L’architecture globale est illustrée par la figure 1 :

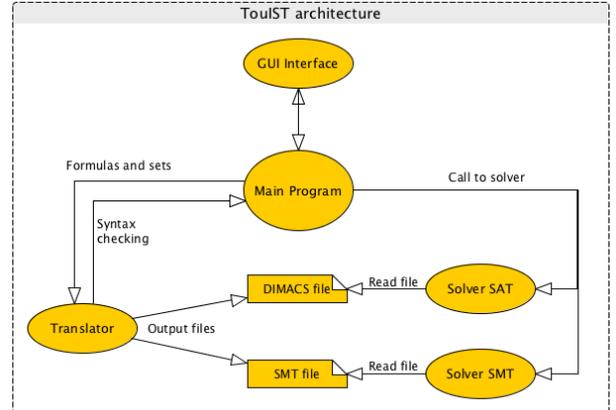


FIGURE 1 – Architecture de TouIST

Avec **TouIST** on accède à un éditeur puissant et convivial pour éditer des formules logiques complexes et des contraintes variées comme :

$$\bigwedge_{i \in \{1..9\}} (P_i \longrightarrow Q_{i+1})$$

qui abrège confortablement :

$$(P_1 \longrightarrow Q_2) \wedge (P_2 \longrightarrow Q_3) \wedge \dots \wedge (P_9 \longrightarrow Q_{10}).$$

Une fois qu’un ensemble de formules a été donné à l’interface, sa satisfiabilité peut être vérifiée : l’interface peut l’envoyer au prouveur qui retourne un modèle, affiché comme le montre la figure 2 si un tel modèle existe. Ensuite par l’intermédiaire de l’interface, l’utilisateur peut par exemple demander d’autres modèles (bouton “Next” de l’interface). Contrairement à **SAToulouse** qui aurait nécessité de modifier les formules pour interdire le modèle et de relancer le solveur, **TouIST** conserve une instance du solveur en attente, ce qui permet d’obtenir les modèles suivants bien plus rapidement.

Name	Value
C(3)	False
B(3)	False
A(3)	False
C(2)	False
C(1)	False
B(2)	False
A(2)	False
B(1)	False
A(1)	False

FIGURE 2 – Affichage de modèle

Les modèles renvoyés par le solveur sont totaux : une valeur est affectée à chacune des variables appa-

raissant dans les formules envoyées au solveur. L'utilisateur peut sélectionner uniquement les propositions vraies ou les propositions fausses. Il peut également sélectionner des sous-ensembles de variables en tapant une expression régulière pour les filtrer.

3 Détail de ce qui peut être fait avec TouIST

3.1 Ensembles de domaines

Avec le temps, nous avons remarqué que nous avons souvent besoin d'écrire des choses comme :

$$\bigwedge_{i \in \{1..9\}} \bigwedge_{j \in \{1..9\}} \bigwedge_{m \in \{A,B,C,D,E,F,G,H,I\}} \left(P_{i,j,m} \longrightarrow \bigwedge_{n \in \{A,B,C,D,E,F,G,H,I\} | m \neq n} \neg P_{i,j,n} \right)$$

Si on lit $P_{i,j,m}$ comme "il y a la lettre m dans la case (i, j) " d'une grille 9×9 , la formule ci-dessus exprime qu'il y a *au plus* une lettre parmi 'A' ... 'I' dans chaque case.

Ces ensembles $\{1..9\}$ et $\{A, B, C, D, E, F, G, H, I\}$ sont des *ensembles de domaines*, et avec TouIST l'utilisateur peut définir autant d'ensembles de domaines qu'il le souhaite, par exemple :

$\$N = [1..9]$
 $\$L = [A, B, C, D, E, F, G, H, I]$

et ainsi écrire la formule précédente comme :

$$\bigwedge_{i \in N} \bigwedge_{j \in N} \bigwedge_{m \in L} P_{i,j,m} \longrightarrow \bigwedge_{n \in L | m \neq n} \neg P_{i,j,n}$$

De plus, les opérations usuelles sur les ensembles (\cup , \cap , \setminus , ...) peuvent être utilisées pour définir d'autres ensembles.

3.2 Formules propositionnelles

Les formules de TouIST sont basées sur des variables propositionnelles (qui peuvent être indicées) et les opérateurs logiques usuels (\wedge , \vee , \longrightarrow , \neg , \leftrightarrow). Ainsi on peut taper des formules usuelles simples comme $Pluie \longrightarrow Nuages$. Mais en plus, nous fournissons des opérateurs logiques de haut niveau qui permettent d'exprimer des assertions complexes dans une forme très compacte.

Conjonctions et disjonctions généralisées

Elles permettent d'exprimer des conjonctions et des disjonctions sur des formules contenant des paramètres qui varient, par exemple :

- $\bigwedge_{i \in N} P_i$, où N est l'ensemble de domaine défini précédemment.
- Elle représente la conjonction $P_1 \wedge P_2 \wedge \dots \wedge P_9$.
- $\bigvee_{i \in E} P_i$.

Bien sûr, ces opérateurs peuvent être imbriqués comme dans :

$$\bigwedge_{i \in N} \bigwedge_{j \in N} \bigvee_{m \in L} P_{i,j,m}$$

qui indique que dans chaque cellule se trouve au moins une lettre.

Contraintes de cardinalité

C'était l'un des sujets "laissé pour le futur" de [2]. Ces opérateurs logiques moins classiques sont disponibles dans TouIST : ils permettent de réduire drastiquement la taille de certaines formules, ce sont : \leq , \geq et $\#$.

L'exemple suivant décrit leur signification :

- $\leq_{i \in N}^2 P_i$ représente "pour au plus deux valeurs de $i \in N$ $P(i)$ est vraie";
- $\geq_{i \in N}^2 P_i$ représente "pour au moins deux valeurs de $i \in N$ $P(i)$ est vraie";
- $\#_{i \in N}^2 P_i$ représente "pour exactement deux valeurs de $i \in N$ $P(i)$ est vraie".

La disjonction généralisée est en fait un cas particulier de ceci (au moins une est vraie), la conjonction aussi (au plus 0 est fausse), et le ou exclusif peut être vu comme exactement une parmi deux est vraie.

Rappelons qu'avec des opérateurs logiques classiques et avec N contenant 9 éléments, $\leq_{i \in N}^3 P_i$ devrait nécessiter une formule contenant 84 propositions P_i puisque cela revient à choisir 3 parmi 9 ce qui donne $\binom{9}{3}$ possibilités, et ni \bigwedge ni \bigvee n'aideraient beaucoup.

Contraintes et calculs sur des indices

Souvent nous avons besoin d'ajouter des contraintes sur les indices, par exemple :

$$\bigwedge_{i \in E} \bigwedge_{j \in E | i \neq j} P_{i,j}$$

qui signifie que $P_{i,j}$ est vraie lorsque $i \neq j$.

C'était la seule contrainte disponible dans SAToulouse, maintenant dans TouIST la gamme de possibilités a été largement enrichie. Les contraintes peuvent inclure des opérateurs usuels de comparaison comme $<$, $>$, \leq , \geq , \neq , $=$ et ces comparaisons peuvent s'appliquer non seulement aux indices, mais aussi à toute expression arithmétique impliquant des indices et $+$, $-$, $*$, $/$, mod , $\sqrt{\quad}$.

Exprimer une phrase comme “chaque case (i, j) contient un nombre qui n’est pas égal à $i + j$ ” donnera :

$$\bigwedge_{i \in N} \bigwedge_{j \in N} \bigvee_{k \in N | k \neq i+j} P_{i,j,k}$$

Bien sûr, *toutes ces phrases* pourraient être exprimées avec les opérateurs logiques usuels bruts, mais ceci serait un travail fastidieux. Néanmoins, les étudiants doivent savoir ce qui est derrière la scène, et qu’une telle formule est l’abréviation de quelque chose comme :

$$P_{1,1,1} \vee P_{1,1,3} \vee P_{1,1,4} \dots P_{1,2,1} \vee P_{1,2,2} \vee P_{1,2,4} \vee \dots$$

qui est très long et terne.

3.3 Aspects techniques

Langage d’entrée vs Langage d’affichage

Les formules que nous avons vues précédemment sont écrites dans le *langage d’affichage* (style L^AT_EX), mais tous ces symboles ne sont pas disponibles sur les claviers, ainsi pour écrire les formules et ensembles de domaines, l’utilisateur utilisera le *langage d’entrée*. Par exemple, la formule précédente avec l’ensemble associé N sera tapée (les variables sont préfixées par \$) :

```
bigand $i,$j in $N,$N:
  bigor $k in $N when $k != $i+$j:
    P($i,$j,$k)
  end
end
```

Mais TouIST les affiche immédiatement en style L^AT_EX comme on peut le voir dans le panneau droit montré sur la figure 3. La définition de l’ensemble N est faite dans l’onglet “Ensembles”.

En outre, les formules peuvent être tapées à la main dans la fenêtre d’édition, ou introduites dans une sorte d’éditeur dirigé par la syntaxe, en raffinant progressivement l’arbre syntaxique, ou encore elles peuvent être importées à partir d’un fichier externe.

4 Sujets avancés pour les étudiants de cycles supérieurs

Dans ce qui suit, nous présentons très succinctement quelques fonctionnalités avancées de TouIST. Elles devraient intéresser les étudiants de cycles supérieurs et leurs enseignants bien sûr. Elles concernent SMT (SAT Modulo Theories), la planification par satisfaction de base de clauses (planification SAT) et leur combinaison la planification SMT.

4.1 SMT : SAT Modulo Theories

Certains problèmes combinatoires nécessitent néanmoins de traiter des calculs sur les nombre naturels ou réels. Ceci peut être fait en utilisant seulement la logique propositionnelle (par exemple, $2+3=5$ pourrait être codé par $add_{2,3,5}$), mais c’est très inconfortable à moins qu’il n’y ait que quelques additions à faire. Ne parlons même pas des opérations de multiplication ou plus complexes. L’idée derrière la genèse de SMT a été de combiner des solveurs SAT avec un solveur arithmétique dans le but d’améliorer le traitement de la partie arithmétique du raisonnement. Dans de nombreux cas, ceci n’améliorera pas seulement l’efficacité du solveur, mais permettra aussi d’exprimer les contraintes arithmétiques des problèmes d’une manière radicalement plus compacte.

Pensez au jeu de Kamaji² où le joueur doit grouper des nombres adjacents dans une grille de sorte que leur somme soit égale à un nombre fixe. Résoudre le jeu nécessite essentiellement un raisonnement logique mais a aussi besoin d’un peu d’arithmétique (addition).

Si $x_{i,j}$ pour chaque case (i, j) est un entier et $G(i, j, i, k)$ représente le fait que les cases (i, j) à (i, k) de la ligne i forment un groupe, la contrainte de somme pourrait être exprimée par :

$$\sum_{m \in E} x_{i,m} = N$$

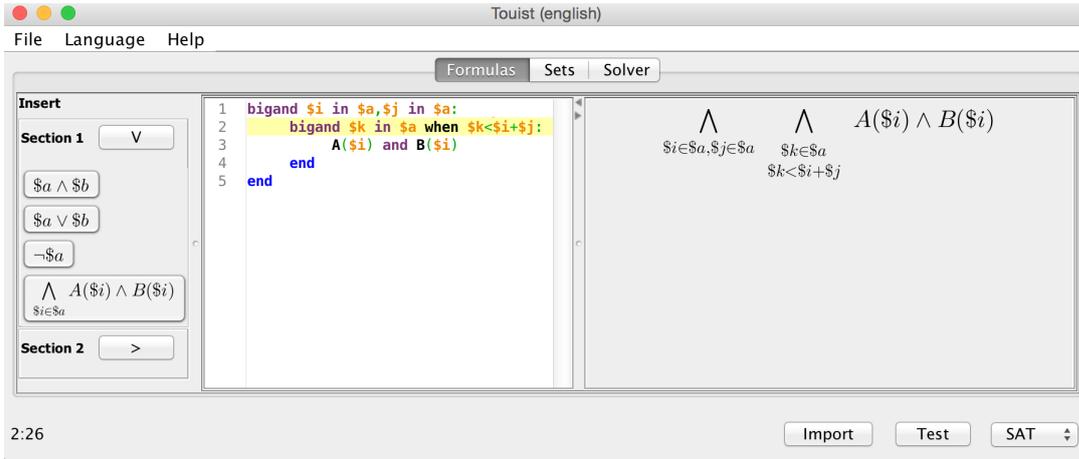
où N est le nombre fixe et E est $\{j, j+1, \dots, k\}$. La logique propositionnelle pure n’est certainement pas adaptée pour de telles phrases !

4.2 TouIST pour la planification classique SAT

En Intelligence Artificielle, la *planification* est un processus cognitif qui consiste à générer automatiquement, au travers d’une procédure formelle, un résultat articulé sous la forme d’un système de décision intégré appelé *plan*. Le plan est généralement sous la forme d’une collection organisée d’*actions* et il doit permettre à l’univers d’évoluer de l’état *initial* à un état qui satisfait le *but*. Dans le cadre classique, le plus restrictif, on considère les actions comme des transitions instantanées sans prendre en compte le temps.

La planification par satisfaction de base de clauses (planification SAT) a été introduite avec le planificateur SATPLAN [4]. Dans cette approche, on travaille directement sur un ensemble fini de variables propositionnelles. Deux actions identiques pouvant apparaître à des endroits différents d’un même plan doivent pouvoir être différenciées et on leur associe donc des propositions différentes. Comme on ne connaît pas à l’avance

2. <http://fr.wikipedia.org/wiki/Kamaji>

FIGURE 3 – Affichage en style \LaTeX

la longueur d'un plan solution d'un problème, on ne peut pas créer un codage unique permettant de le résoudre puisqu'il faudrait créer une infinité de variables propositionnelles pour représenter toutes les actions de tous les plans possibles. La solution la plus commune consiste alors à créer un codage représentant tous les plans d'une longueur k fixée. La base de clause ainsi obtenue est donnée en entrée à un solveur SAT qui retourne, lorsqu'il existe, un modèle de cette base. Le décodage de ce modèle permet alors d'obtenir un plan-solution. Si la résolution du codage ne donne pas de modèle, la valeur de k est augmentée et le processus réitéré. Pour la complétude du procédé, tous ces modèles doivent correspondre exactement à tous les plans solutions d'une longueur fixée du problème.

Une différence importante de TouIST en comparaison avec SATTOULOUSE est sa capacité à prendre en compte à la fois des formules logiques et ensembles de domaines. Par exemple, si l'on veut résoudre un problème de planification particulier, SATTOULOUSE est facile à utiliser pour décrire le problème et le résoudre via un solveur SAT. Mais si notre objectif est de résoudre plusieurs problèmes de planification génériques, nous pouvons profiter de la flexibilité de TouIST qui permet à l'utilisateur de décrire une méthode générique de résolution avec des règles encodées comme formules et d'utiliser des ensembles de domaines pour décrire chaque problème de planification particulier. De nombreuses règles de codage pour la résolution de problèmes de planification ont déjà été proposées [4, 5, 7]. Comme exemple d'une telle règle nous donnons ci-après un codage des *frame-axioms*. Si un fait est faux à une étape $i - 1$ d'un plan solution et devient vrai à l'étape i , alors la disjonction des actions qui peuvent établir le fait à l'étape i du plan est vraie. C'est à dire, au moins une action qui établit le fait doit

avoir été appliquée.

$$\bigwedge_{i \in \{1..longueur_plan\}} \bigwedge_{f \in Faits} \left((\neg f(i-1) \wedge f(i)) \Rightarrow \bigvee_{a \in Actions / f \in Effets(a)} a(i) \right)$$

4.3 TouIST pour la planification temporelle SMT

Pour résoudre des problèmes de planification réels, l'une des principales difficultés à surmonter consiste à prendre en compte la dimension temporelle. En effet, de nombreux problèmes du monde réel nécessitent, pour être résolus ou exécutés plus efficacement, la prise en compte de la durée des actions, des instants auxquels des événements se produisent, ou encore la concurrence d'actions.

En complément de SAT, notre nouvelle plateforme TouIST est capable de gérer des théories comme la différence logique ou l'arithmétique linéaire sur les nombres entiers (QF_IDL, QF_LIA) ou les nombres réels (QF_RDL, QF_LRA), et d'appeler un solveur SMT pour trouver une solution. Pour être résolus, les problèmes de planification temporelle issus du monde réel nécessitent une représentation continue du temps, et donc, l'utilisation de nombres réels dans les codages logiques. TouIST peut être utilisé pour résoudre de tels problèmes impliquant des actions avec durée, des événements exogènes et des buts temporellement étendus, par exemple avec les règles de codage proposées dans [6]. Nous donnons ci-après un codage des exclusions mutuelles temporelles d'actions. Si deux actions a et b

produisant respectivement une proposition p et sa négation sont actives dans le plan ($a \wedge b$ est vrai), alors l'intervalle de temps $[\tau(a \mid \rightarrow p), \tau(a \rightarrow \mid p)]$ correspondant à l'activation de p , et l'intervalle de temps $[\tau(b \mid \rightarrow \neg p), \tau(b \rightarrow \mid \neg p)]$ correspondant à l'activation de $\neg p$ sont disjoints. Dans ce cas, il faut ajouter une disjonction pour imposer que la fin de l'un de ces intervalles soit strictement avant le début de l'autre.

$$\bigwedge_{a \in \text{Actions}} \bigwedge_{b \in \text{Actions}} \bigwedge_{f \in \text{Faits} \mid f \in \text{Effets}(a) \wedge \neg f \in \text{Effets}(b)} \left((a \wedge b) \Rightarrow \left(\begin{array}{l} (\tau(b \rightarrow \mid \neg f) < \tau(a \mid \rightarrow f)) \\ \vee (\tau(a \rightarrow \mid f) < \tau(b \mid \rightarrow \neg f)) \end{array} \right) \right)$$

5 Conclusion

À notre connaissance, il n'existe pas d'autre outil qui cible un public aussi large, ni la même grande classe de problèmes, ni avec le même confort d'utilisation. La plupart des outils pédagogiques existants (soit l'implémentation de tables de vérité ou tableaux sémantiques) qui pourraient faire le travail de recherche d'un modèle ne peuvent pas gérer efficacement de gros problèmes, et de vrais outils capables de les traiter ne sont certainement pas conçus pour être utilisés par des débutants en logique, et même pas par la plupart des étudiants de cycles supérieurs. Des outils avancés conçus pour des sujets d'études supérieures, comme Mozart [8] ou Alloy [3] ont une courbe d'apprentissage raide qui peut dissuader les débutants et les utilisateurs non-spécialistes.

Nous croyons que TouIST sera utile pour les débutants en logique ainsi que pour les utilisateurs avancés grâce à sa grande gamme d'applications et sa facilité d'utilisation.

Références

- [1] Daniel Le Berre and Anne Parrain. The sat4j library, release 2.2. *JSAT*, 7(2-3) :59–6, 2010.
- [2] Olivier Gasquet, François Schwarzenruber, and Martin Strecker. Satoulouse : the computational power of propositional logic shown to beginners. In P. Blackburn, H. van Ditmarsch, M. Manzano, and F. Soler-Tosca, editors, *Third International Congress on Tools for Teaching Logic (ICTTL'2011)*, volume 6680 of *Lecture Notes in Computer Science*, pages 77–84. Springer, 2011.
- [3] Daniel Jackson. *Software Abstractions : Logic, Language, and Analysis*. The MIT Press, 2006.
- [4] Henry Kautz and Bart Selman. Planning as satisfiability. In *IN ECAI-92*, pages 359–363. Wiley, 1992.
- [5] Amol Dattatraya Mali and Subbarao Kambhampati. On the utility of plan-space (causal) encodings. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence and Eleventh Conference on Innovative Applications of Artificial Intelligence, 1999*, pages 557–563, 1999.
- [6] Frédéric Maris and Pierre Régnier. Tlp-gp : New results on temporally-expressive planning benchmarks. In *International Conference on Tools with Artificial Intelligence (ICTAI)*, volume 1, pages 507–514. IEEE Computer Society, 2008.
- [7] Jussi Rintanen, Keijo Heljanko, and Ilkka Niemelä. Planning as satisfiability : Parallel plans and algorithms for plan search. *Artif. Intell.*, 170(12) :1031–1080, 2006.
- [8] Peter Van Roy, editor. *Multiparadigm Programming in Mozart/Oz, Second International Conference, MOZ 2004, Charleroi, Belgium, October 7-8, 2004, Revised Selected and Invited Papers*, volume 3389 of *Lecture Notes in Computer Science*. Springer, 2005.

Séquences de choix sincères et partages équitables de biens indivisibles

Sylvain Bouveret¹

Michel Lemaître

¹ Univ. Grenoble Alpes, LIG, France

sylvain.bouveret@imag.fr michel.lemaitre.31@gmail.com

Résumé

Le contexte est celui du partage d'un ensemble d'objets indivisibles entre plusieurs agents ayant des préférences numériques additives sur les objets. Dans ce contexte, le partage par séquence de choix sincères joue un rôle naturel et remarquable. Ce protocole consiste à donner la main successivement à chaque agent dans l'ordre défini par la séquence, chacun à son tour choisissant son objet préféré parmi les objets restants.

Nous caractérisons d'une part les allocations séquençables (*i.e.* pouvant être obtenues par séquence de choix sincères), et d'autre part les instances pour lesquelles toute allocation est séquençable.

Nous montrons ensuite que toute allocation Pareto-optimale est séquençable mais que l'inverse n'est pas vrai. D'où l'existence d'une « échelle d'efficacité » comportant trois paliers : Pareto-optimalité, séquençabilité sans Pareto-optimalité, et non séquençabilité.

Nous montrons également qu'il existe des allocations sans envie non séquençables, mais que toute allocation de type *Competitive Equilibrium with Equal Income* (CEEI) [13, 7] est séquençable.

Abstract

We investigate fair division of indivisible goods, where the agents have numerical additive preferences on the objects. Using a sincere picking sequence is a natural and remarkable way to allocate the objects in this context. The idea is the following : at each stage, a designated agent chooses an object among those that remain available.

In this paper, we characterize sequenceable allocations, and the instances for which each allocation is sequenceable. Then we show that each Pareto-optimal allocation is sequenceable, but that, rather surprisingly, the converse is not true. It leads to the definition a "scale of efficiency" having three steps : Pareto-optimality, sequenceability without Pareto-optimality, and non-sequenceability.

We also show that an allocation can be non-sequenceable but envy-free. However, every Competitive Equilibrium with Equal Income [13, 7] is sequenceable.

1 Introduction

Nous nous penchons dans cet article sur un problème de décision collective classique : le partage de biens indivisibles. Dans ce problème, un ensemble de biens ou d'objets indivisibles doit être partagé entre des agents (individus, entités,...), en tenant compte du mieux possible des préférences des agents pour ces objets. Ce problème a des applications pratiques nombreuses et variées, parmi lesquelles, pour n'en nommer que quelques-unes, l'allocation de ressources spatiales [21, 2], l'allocation de travaux à des employés ou à des élèves, d'articles à des relecteurs, de places de cours à des étudiants [13], en passant par la répartition de droits d'émission de polluants.

Si ce problème classique a été étudié de longue date par les économistes [10, 17, 8], il est l'objet depuis quelques années de nombreux travaux dans le domaine de l'informatique et de l'intelligence artificielle, en vertu de ses aspects fortement combinatoires. Ces travaux se sont intéressés entre autres à la représentation des préférences liée à ces problèmes de partage [3], à la complexité liée au calcul de partages satisfaisant certaines propriétés d'équité [22, 4, 24] ou aux aspects algorithmiques de ce calcul [6, 1].

On peut envisager le problème de partage sous deux points de vue. La première possibilité est de recourir à une entité supposée désintéressée, en charge de recueillir de manière *centralisée* les préférences de tous les agents concernés sur les objets, et de calculer une allocation prenant en compte ces préférences, et sa-

tisfaisant certains critères d'équité (l'absence d'envie par exemple) et d'efficacité (la Pareto-optimalité par exemple), ou optimisant une fonction d'utilité collective bien choisie. À l'opposé, on peut adopter un point de vue totalement *distribué*, en partant par exemple d'une allocation initiale et en laissant les agents échanger des objets entre eux par négociation [26, 14].

Un point de vue intermédiaire entre ces deux approches consiste à allouer les objets aux agents par l'utilisation d'un *protocole* de partage. Un tel protocole peut être vu comme une séquence de questions posées aux agents et permettant de calculer un partage de manière interactive. Si cette notion de protocole est majoritairement au cœur des travaux concernant le partage de biens divisibles — *cake-cutting*, voir l'ouvrage de Brams et Taylor [11] pour une référence sur le sujet — elle a été étudiée également dans le cadre du partage d'objets indivisibles [11, 9].

Nous nous intéressons dans cet article à un protocole de partage particulier : le partage par *séquence sincère*. Ce protocole, très simple à comprendre et à mettre en œuvre, fonctionne de la manière suivante. Une autorité centrale fixe au préalable une séquence d'agents, comportant autant d'agents que d'objets (s'il y a plus d'objets que d'agents, certains agents apparaissent plusieurs fois dans la séquence). Puis, on demande à chaque agent apparaissant dans la séquence de choisir à son tour l'objet qu'il préfère parmi ceux restant. Ainsi par exemple, selon la séquence $\langle 1, 2, 2, 1 \rangle$, l'agent 1 va choisir en premier, puis l'agent 2 choisira deux objets consécutifs, et enfin l'agent 1 prendra le dernier objet. Ce protocole simple, utilisé en fait dans de nombreuses situations de la vie quotidienne¹ a été étudié pour la première fois par Kohler et Chandrasekaran [20]. Brams et Taylor [12] se sont ensuite penchés sur une version particulière de ce protocole, les séquences alternées, dans laquelle la séquence d'agents est contrainte à une alternance équilibrée $\langle 1, 2, 2, 1, \dots \rangle$ ou stricte $\langle 1, 2, 1, 2, \dots \rangle$ d'agents. Enfin, Bouveret et Lang [5] ont donné une formalisation plus générale et systématique de ce protocole, dont les propriétés (en particulier liées à la théorie des jeux) ont été mises en évidence par Kalinowski *et al.* [19, 18].

Parmi l'ensemble des travaux liés à ce protocole de partage par séquences sincères, nous pouvons distinguer les travaux de Brams *et al.* [8, 10]. Ces travaux, qui se placent dans un contexte de partage avec préférences ordinales dans lequel les agents ne peuvent recevoir que des parts de même taille, font un lien intéressant entre la notion de séquence sincère et le critère classique de Pareto-optimalité, en démontrant entre autres que non seulement tout partage obtenu

par exécution du protocole de séquence sincère est Pareto-optimal, mais aussi que tout partage Pareto-optimal peut être obtenu par l'exécution du protocole de séquence sincère pour une séquence particulière.

Dans cet article, nous reprenons les idées de ces travaux afin d'analyser les liens entre séquences et diverses propriétés d'efficacité et d'équité dans un modèle plus général, dans lequel les agents ont des préférences numériques additives sur les objets. Nos contributions sont les suivantes. Nous donnons une formalisation du lien entre allocation et séquence sincère, en introduisant la notion de *séquençabilité*, et en mettant en évidence une caractérisation simple de la séquençabilité d'une allocation. Nous montrons ensuite que dans ce contexte plus général que celui des travaux précédents, de manière étonnante, il n'y a équivalence entre séquençabilité et Pareto-optimalité. Cela nous permet de mettre en évidence une « échelle d'efficacité » permettant, à la manière de l'« échelle d'équité » que nous avons introduite dans un travail antérieur [7], de caractériser le degré d'efficacité d'un partage. Nous mettons de plus en évidence un lien entre séquençabilité et une autre propriété économique importante : l'équilibre compétitif à revenus égaux.

L'article est organisé comme suit. La section 2 décrit le modèle de notre problème de partage ainsi que le protocole d'allocation par séquence sincère. La section 3 s'intéresse au problème consistant à décider si une allocation peut être obtenue par une séquence de choix sincères. Dans les deux sections suivantes, nous analysons la relation entre séquences de choix sincères et deux propriétés classiques des allocations : la Pareto-optimalité (section 4) et l'équilibre compétitif à revenus égaux (section 5). Nous explorons enfin les propriétés pratiques liées à la notion de séquençabilité dans la section 6 dédiée aux expérimentations.

2 Modèle et définitions

Le problème de partage de biens indivisibles consiste à allouer un ensemble fini d'*objets* $\mathcal{O} = \{1, \dots, M\}$ à un ensemble fini d'*agents* $\mathcal{A} = \{1, \dots, N\}$.

Une *sous-allocation* sur $\mathcal{O}' \subseteq \mathcal{O}$ est un vecteur $\vec{\pi}^{\mathcal{O}'} = \langle \pi_1^{\mathcal{O}'}, \dots, \pi_N^{\mathcal{O}'} \rangle$, où $\pi_i^{\mathcal{O}'} \subseteq \mathcal{O}'$ est la *part* de l'agent i sur \mathcal{O}' , avec $i \neq j \Rightarrow \pi_i^{\mathcal{O}'} \cap \pi_j^{\mathcal{O}'} = \emptyset$ (un objet n'est alloué qu'à un seul agent) et $\cup_{i \in \mathcal{A}} \pi_i^{\mathcal{O}'} = \mathcal{O}'$ (tous les objets de \mathcal{O}' sont alloués). On dit que $\vec{\pi}^{\mathcal{O}'}$ est une sous-allocation de $\vec{\pi}^{\mathcal{O}}$ lorsque $\pi_i^{\mathcal{O}'} \subseteq \pi_i^{\mathcal{O}}$ pour tout agent i . Une sous-allocation $\vec{\pi}^{\mathcal{O}}$ sur tous les objets sera notée simplement $\vec{\pi}$ et appelée *allocation*.

La construction d'une allocation satisfaisante s'appuie nécessairement sur la connaissance des préférences des agents sur les objets. Nous ferons l'hypo-

1. C'est par exemple le protocole utilisé dans le jeu des *Colons de Catane* pour l'allocation des ressources initiales.

thèse classique de *préférences numériques additives* : chaque agent i possède une *fonction d'utilité* $u_i : 2^{\mathcal{O}} \rightarrow \mathbb{R}^+$ qui mesure la satisfaction $u_i(\pi)$ qu'il obtient si on lui alloue la part π , ainsi définie :

$$u_i(\pi) \stackrel{\text{def}}{=} \sum_{\ell \in \pi} W(i, \ell),$$

où $W(i, \ell)$ représente le poids donné par l'agent i à l'objet ℓ . Cette hypothèse, bien que restrictive, est faite par de nombreux auteurs et fournit un bon compromis entre expressivité des préférences et concision.

Étant donné un agent i et un ensemble d'objets O , on désignera par $\text{best}(O, i)$ le sous-ensemble des objets de préférence maximale de l'agent i dans le sous-ensemble d'objets O . Formellement : $\ell \in \text{best}(O, i)$ si et seulement si $\ell \in O$ et pour tout $\ell' \in O : W(i, \ell) \geq W(i, \ell')$. Une (sous-)allocation $\vec{\pi}^{|O'}$ sera dite *frustrante* si aucun agent ne reçoit un de ses objets préférés dans $\vec{\pi}^{|O'}$ (formellement : $\text{best}(O', i) \cap \vec{\pi}_i^{|O'} = \emptyset$ pour tout agent i), et *non frustrante* si au moins un des agents reçoit un de ses objets préférés dans $\vec{\pi}^{|O'}$.

Les résultats présentés dans cet article font intervenir des hypothèses sur les préférences des agents. Les définitions suivantes précisent ce point.

Nous dirons que *les préférences des agents sont strictes sur les objets* si, pour chaque agent, les poids qu'il donne aux objets sont tous différents (donc ordonnés strictement). Formellement : pour tout agent i et tout couple d'objets $\ell \neq m$, on a $W(i, \ell) \neq W(i, m)$.

Nous dirons que *les préférences des agents sont strictes sur les parts* si, pour chaque agent, des parts distinctes ont des utilités distinctes. Formellement : pour tout agent i et tout couple de parts $\pi \neq \pi'$, on a $u_i(\pi) \neq u_i(\pi')$.

Nous dirons que *les préférences des agents sont ordonnées identiquement* lorsque tous les agents ordonnent leurs préférences sur les objets de la même façon. Formellement : il existe une permutation $\eta : \mathcal{O} \rightarrow \mathcal{O}$ telle que pour tout agent i et tout couple d'objets ℓ et m , si $\eta(\ell) < \eta(m)$ alors $W(i, \eta(\ell)) \geq W(i, \eta(m))$.

Remarque 1. *La stricticité sur les parts implique la stricticité sur les objets, mais la réciproque est fautive.*

Les données du modèle sont ainsi résumées :

Définition 1. *Une instance de problème de partage de biens indivisibles à préférences numériques additives $I = \langle \mathcal{A}, \mathcal{O}, W \rangle$ est définie par :*

- un ensemble $\mathcal{A} = \{1, \dots, i, \dots, N\}$ de N agents,
- un ensemble $\mathcal{O} = \{1, \dots, \ell, \dots, M\}$ de M objets,
- une fonction $W : \mathcal{A} \times \mathcal{O} \rightarrow \mathbb{R}^+$, où $W(i, \ell)$ est le poids donné par l'agent i à l'objet ℓ .

$\mathcal{P}(I)$ désignera l'ensemble des allocations pour I .

Séquence de choix sincères De manière informelle, le partage par *séquence de choix sincères* consiste à donner la main successivement à chaque agent dans l'ordre défini par la séquence, chacun à son tour choisissant l'objet ou l'un des objets qu'il préfère parmi les objets restants.

Définition 2. *Soit une instance $I = \langle \mathcal{A}, \mathcal{O}, W \rangle$. Une séquence de choix sincères est un vecteur de \mathcal{A}^M .*

Nous utiliserons souvent le terme « séquence » à la place de « séquence de choix sincères ».

On désigne par $\mathcal{S}(I)$ l'ensemble des séquences possibles pour l'instance I .

On dit que la séquence $\vec{\sigma}$ engendre l'allocation $\vec{\pi}$ si et seulement si $\vec{\pi}$ peut être obtenue par l'algorithme non déterministe² 1.

Algorithme 1 : Exécution d'une séquence

Entrées : une instance $I = \langle \mathcal{A}, \mathcal{O}, W \rangle$ et une séquence $\vec{\sigma} \in \mathcal{S}(I)$

Sorties : une allocation $\vec{\pi} \in \mathcal{P}(I)$

- 1 $\vec{\pi} \leftarrow$ allocation vide (telle que $\forall i \in \mathcal{A} : \pi_i = \emptyset$);
 - 2 $\mathcal{O}_1 \leftarrow \mathcal{O}$;
 - 3 **pour** t **de** 1 **à** M **faire**
 - 4 $i \leftarrow \sigma_t$;
 - 5 **choisir** l'objet $o_t \in \text{best}(\mathcal{O}_t, i)$;
 - 6 $\pi_i \leftarrow \pi_i \cup \{o_t\}$;
 - 7 $\mathcal{O}_{t+1} \leftarrow \mathcal{O}_t \setminus \{o_t\}$
-

Définition 3. *Une allocation $\vec{\pi}$ est dite séquençable s'il existe une séquence $\vec{\sigma}$ qui engendre $\vec{\pi}$, et non séquençable si aucune séquence ne peut engendrer $\vec{\pi}$.*

Cette définition établit l'existence d'une *correspondance* $s(I) = (\mathcal{S}(I), \mathcal{P}(I), \Sigma(I))$ définie par $(\vec{\sigma}, \vec{\pi}) \in \Sigma(I)$ si et seulement si $\vec{\pi}$ peut être engendrée par $\vec{\sigma}$.

Exemple 1. *Soit l'instance I représentée par la matrice de poids³ suivante :*

$$\begin{pmatrix} 8 & 2 & 1 \\ 5 & 1 & 5 \end{pmatrix}$$

L'ensemble des séquences possible $\mathcal{S}(I)$ est :

$$\begin{array}{lll} a = \langle 1, 1, 1 \rangle & b = \langle 1, 1, 2 \rangle & c = \langle 1, 2, 1 \rangle \\ d = \langle 1, 2, 2 \rangle & e = \langle 2, 1, 1 \rangle & f = \langle 2, 1, 2 \rangle \\ g = \langle 2, 2, 1 \rangle & h = \langle 2, 2, 2 \rangle. & \end{array}$$

2. L'algorithme contient une instruction **choisir** à partir de laquelle le contrôle se sépare en plusieurs branches, lesquelles construisent toutes les allocations engendrées par $\vec{\sigma}$.

3. Dans cet exemple et dans les suivants, nous représenterons les préférences des agents par une matrice dans laquelle la valeur de la ligne i et de la colonne ℓ représente le poids $W(i, \ell)$.

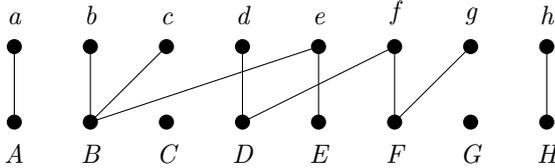
L'ensemble des allocations possibles $\mathcal{P}(I)$ est :

$$\begin{aligned} A &= \langle \{1, 2, 3\}, \{\} \rangle & B &= \langle \{1, 2\}, \{3\} \rangle \\ C &= \langle \{1, 3\}, \{2\} \rangle & D &= \langle \{1\}, \{2, 3\} \rangle \\ E &= \langle \{2, 3\}, \{1\} \rangle & F &= \langle \{2\}, \{1, 3\} \rangle \\ G &= \langle \{3\}, \{1, 2\} \rangle & H &= \langle \{\}, \{1, 2, 3\} \rangle. \end{aligned}$$

La correspondance $s(I)$ est telle que

$$\Sigma(I) = \{(a, A), (b, B), (c, B), (d, D), (e, B), (e, E), (f, D), (f, F), (g, F), (h, H)\}.$$

Sous forme graphique :



On remarque par exemple que la séquence e engendre deux allocations possibles : E et B , selon que l'agent 2 choisisse soit l'objet 1 soit l'objet 3 qu'il préfère tous deux également. L'allocation B peut être engendrée par les trois séquences b, c et e . En revanche les deux allocations C et G sont non séquençables (aucune séquence ne peut les produire).

Remarque 2. Pour une instance donnée I , nous avons $|\mathcal{S}(I)| = |\mathcal{P}(I)| = N^M$.

Remarque 3. Le nombre d'objets alloués à un agent par une séquence est égal au nombre de fois où l'agent apparaît dans la séquence. Formellement : pour tout $(\vec{\sigma}, \vec{\pi}) \in \Sigma(I)$ et tout pour tout agent i nous avons $|\pi_i| = \sum_{\ell \in \mathcal{O}} [\sigma_\ell = i]$ où $[z = t]$ vaut 1 si l'égalité est vérifiée et 0 sinon.

3 Correspondance entre séquences de choix sincères et allocations

3.1 Caractérisation des allocations séquençables

Nous avons vu plus haut dans l'exemple 1 que certaines allocations sont non séquençables. Les remarques et exemples suivants précisent ce point.

Remarque 4. Toute allocation frustrante est non séquençable.

Démonstration. Par la contraposée : dans toute allocation séquençable, un agent au moins (le premier de la séquence) obtient un objet préféré. \square

Exemple 2. Soit l'instance représentée par la matrice suivante :

$$\begin{pmatrix} 2 & *1 & *1 \\ *1 & 2 & 2 \end{pmatrix}$$

L'allocation $\langle \{2, 3\}, \{1\} \rangle$ marquée par des $*$ est non séquençable car elle est frustrante.

Toutefois, on peut trouver une allocation non séquençable qui alloue son objet préféré à un agent (comme dans l'exemple 1, allocations C et G) ou même à tous, comme dans l'exemple suivant.

Exemple 3. Soit l'instance représentée par la matrice suivante :

$$\begin{pmatrix} *9 & 8 & 2 & *1 \\ 2 & *5 & *1 & 4 \end{pmatrix}$$

Dans $\vec{\pi} = \langle \{1, 4\}, \{2, 3\} \rangle$ marquée par des $*$, chaque agent reçoit son objet préféré. Pourtant, une fois les deux objets 1 et 2 obligatoirement alloués par toute séquence produisant $\vec{\pi}$, il subsiste la sous-allocation

$$\begin{pmatrix} 2 & *1 \\ *1 & 4 \end{pmatrix}$$

non séquençable parce que frustrante. L'allocation $\vec{\pi}$ de départ est donc non séquençable.

Cette propriété de contenir une sous-allocation frustrante caractérise exactement l'ensemble des allocations non séquençables. Plus formellement :

Proposition 1. Soit $I = \langle \mathcal{A}, \mathcal{O}, W \rangle$ une instance et $\vec{\pi}$ une allocation de cette instance. Les deux propositions suivantes sont équivalentes.

- (A) $\vec{\pi}$ est séquençable.
- (B) Aucune sous-allocation de $\vec{\pi}$ n'est frustrante (dans toute sous-allocation, un agent au moins reçoit un objet préféré).

Démonstration. (B) implique (A). Supposons donc que pour tout sous-ensemble d'objets $\mathcal{O}' \subseteq \mathcal{O}$ il existe au moins un agent qui obtient un de ses objets préférés dans $\vec{\pi}^{\mathcal{O}'}$. Nous allons montrer que $\vec{\pi}$ est séquençable, en construisant explicitement une séquence qui l'engendre. Soit $\vec{\sigma}$ la séquence d'agents et $\vec{\mathcal{O}} \in (2^{\mathcal{O}})^M$ une séquence d'ensembles d'objets conjointement définis de la manière suivante :

- $\mathcal{O}_1 = \mathcal{O}$ et σ_1 est un agent qui reçoit un de ses objets préférés dans $\vec{\pi}^{\mathcal{O}_1}$;
- $\mathcal{O}_{t+1} = \mathcal{O}_t \setminus \{o_t\}$, où $o_t \in \text{best}(\mathcal{O}_t, \sigma_t)$ et σ_t est un agent qui reçoit un de ses objets préférés dans $\vec{\pi}^{\mathcal{O}_t}$, pour $t \geq 1$.

D'après l'hypothèse sur $\vec{\pi}$, on peut vérifier que la séquence $\vec{\sigma}$ est parfaitement définie. De plus, $\vec{\pi}$ est l'une des allocations engendrées par la séquence $\vec{\sigma}$.

(A) implique (B) par contraposée. Soit $\vec{\pi}$ une allocation contenant une sous-allocation frustrante $\vec{\pi}^{\mathcal{O}'}$. Supposons qu'il existe une allocation $\vec{\sigma}$ engendrant $\vec{\pi}$. Remarquons que dans l'algorithme 1, lorsqu'un objet est alloué à un agent, tous les objets strictement meilleurs pour lui ont déjà été alloués à une étape antérieure. Soit $\ell \in \mathcal{O}'$, et soit i l'agent qui reçoit ℓ dans $\vec{\pi}$. Puisque $\vec{\pi}^{\mathcal{O}'}$ est frustrante, il existe un autre objet $m \in \mathcal{O}'$ tel que $W(i, m) > W(i, \ell)$. D'après la remarque précédente, m est donc nécessairement alloué avant ℓ dans l'exécution de la séquence σ . On peut déduire, du même raisonnement sur m et l'agent j qui le reçoit, qu'il existe un autre objet p alloué avant m

dans l'exécution de la séquence. L'ensemble \mathcal{O}' étant fini, en itérant le raisonnement, on retombera nécessairement sur un objet déjà rencontré, conduisant à un cycle dans la relation de précédence des objets dans l'exécution de la séquence. C'est une contradiction : aucune séquence ne peut donc engendrer $\vec{\pi}$. \square

Outre le fait de caractériser une allocation séquençable, la proposition 1, ou plus précisément sa preuve, met en évidence une approche constructive permettant de tester si une allocation est séquençable, et si c'est le cas, de calculer une séquence qui engendre cette allocation. Cela nous conduit au résultat suivant :

Proposition 2. *Soit une instance $I = \langle \mathcal{A}, \mathcal{O}, W \rangle$ et une allocation $\vec{\pi}$ de cette instance, on peut décider en temps $O(N \times M^2)$ si $\vec{\pi}$ est séquençable.*

Algorithme 2 : Séquençage d'une allocation

Entrées : une instance $I = \langle \mathcal{A}, \mathcal{O}, W \rangle$
et une allocation $\vec{\pi}$ de I

Sorties : une séquence $\vec{\sigma}$ engendrant $\vec{\pi}$,
ou *Non séquençable*

```

1  $\vec{\sigma} \leftarrow ()$ ;
2  $\mathcal{O}' \leftarrow \mathcal{O}$ ;
3 pour  $t$  de 1 à  $M$  faire
4   si  $\exists i$  tel que  $\text{best}(\mathcal{O}', i) \cap \pi_i \neq \emptyset$  alors
5      $\vec{\sigma} \leftarrow \vec{\sigma} \cdot i$ ;
6     soit  $\ell \in \text{best}(\mathcal{O}', i) \cap \pi_i$ ;
7      $\mathcal{O}' \leftarrow \mathcal{O}' \setminus \{\ell\}$ ;
8   sinon renvoyer Non séquençable ;
9 renvoyer  $\vec{\sigma}$ ;
```

Démonstration. Montrons que l'algorithme 2 renvoie une séquence $\vec{\sigma}$ engendrant l'allocation $\vec{\pi}$ donnée en entrée si et seulement s'il en existe une. Supposons que l'algorithme renvoie une séquence $\vec{\sigma}$. Alors, par définition de la séquence (dans la boucle des lignes 4 à 7), à chaque étape t , $i = \sigma_t$ peut choisir un objet appartenant à π_i , qui soit l'un de ses meilleurs objets. Réciproquement, supposons que l'algorithme renvoie *Non séquençable*. Alors à une étape t donnée, $\forall i$, $\text{best}(\mathcal{O}', i) \cap \pi_i = \emptyset$. Par définition $\vec{\pi}^{\mathcal{O}'}$ est donc, à cette étape, une sous-allocation frustrante de $\vec{\pi}$. D'après la proposition 1, $\vec{\pi}$ n'est donc pas séquençable.

La boucle des lignes 4 à 7 peut être exécutée en temps $O(N \times M)$, car la recherche des meilleurs objets dans les préférences de chaque agent peut être effectuée en temps $O(M)$. La boucle étant exécutée M fois, l'algorithme s'exécute globalement en temps $O(N \times M^2)$. \square

3.2 Préférences strictes sur les objets

Nous caractérisons les instances pour lesquelles la correspondance $s(I)$ est une application.

Proposition 3. *Si les préférences sont strictes sur les objets alors la correspondance $s(I)$ est une application $s(I) : \mathcal{S}(I) \mapsto \mathcal{P}(I)$, et réciproquement.*

Démonstration. Si, comme dans l'exemple 3, les préférences des agents sont strictes sur les objets, alors dans l'algorithme 1, l'agent i qui a la main dans la séquence n'a qu'un seul choix possible pour l'objet ℓ dans $\text{best}(\mathcal{O}, i)$, et toute séquence engendre une et une seule allocation.

Réciproquement, si les préférences ne sont pas strictes sur les objets, un des agents au moins (supposons sans perte de généralité qu'il s'agit de l'agent 1) donne le même poids à deux objets distincts (supposons sans perte de généralité qu'il s'agit des objets 1 et 2). Supposons qu'il y ait exactement t objets classés au-dessus. Alors on peut voir facilement que la séquence

$$\underbrace{111\dots 111}_{t+1 \text{ fois}} \underbrace{222\dots 222}_{M-t-1 \text{ fois}}$$

engendre deux allocations, selon le choix de l'agent 1 entre les objets 1 et 2 à l'étape $t+1$. \square

Cette application est une injection et non une bijection en général, car il peut exister des allocations non séquençables, même si les préférences sont strictes sur les objets, comme le montre l'exemple 3 ci-dessus.

3.3 Préférences ordonnées identiquement

Proposition 4. *Toutes les allocations d'une instance à préférences ordonnées identiquement sont séquençables. Réciproquement, si toutes les allocations d'une instance sont séquençables, ses préférences sont ordonnées identiquement.*

Démonstration. Sens direct : soit une instance à préférences ordonnées identiquement, et une allocation quelconque $\vec{\pi}$. Dans toute sous-allocation de $\vec{\pi}$ au moins un agent obtient un objet préféré (parce que l'ordonnement est identique) donc ne peut être frustrante. Par la proposition 1, $\vec{\pi}$ est séquençable.

Réciproque, par la contraposée : soit une instance dont les préférences ne sont pas identiquement ordonnées. Il existe donc deux objets distincts ℓ et m et deux agents distincts i et j tels que $W(i, \ell) \geq W(j, \ell)$ et $W(i, m) \leq W(j, m)$, l'une des deux inégalités étant stricte (supposons sans perte de généralité qu'il s'agit de la première). La sous-allocation $\vec{\pi}^{\{\ell, m\}}$ telle que $\pi_i^{\{\ell, m\}} = \{m\}$ et $\pi_j^{\{\ell, m\}} = \{\ell\}$ est frustrante. Par la proposition 1 toute allocation $\vec{\pi}$ contenant cette sous-allocation frustrante (donc telle que $m \in \pi_i$ et $\ell \in \pi_j$) est non séquençable. \square

Nous caractérisons maintenant les instances pour lesquelles la correspondance $s(I)$ est une bijection.

Proposition 5. *Pour une instance donnée, les deux énoncés suivants sont équivalents.*

(A) *Les préférences sont strictes sur les objets et ordonnées identiquement.*

(B) La correspondance $s(I)$ est une bijection.

Démonstration. (A) implique (B) : par la proposition 3 la correspondance est une application. La proposition 4 implique que cette application est surjective. Le domaine et le codomaine de cette application ayant la même taille (remarque 2), c'est une bijection.

(B) implique (A) : la correspondance étant une application, par la proposition 3, les préférences des agents sont strictes sur les objets. Puisque c'est une bijection toutes les allocations sont séquençables. Par la proposition 4 les préférences sont ordonnées identiquement. \square

4 Séquences de choix sincères et Pareto-optimalité

Une allocation est Pareto-optimale s'il n'existe aucune autre allocation qui la domine. Dans notre contexte, l'allocation $\vec{\pi}'$ domine l'allocation $\vec{\pi}$ si pour tout agent i , $u_i(\pi'_i) \geq u_i(\pi_i)$, avec au moins un agent pour lequel l'inégalité est stricte. La notion de Pareto-optimalité capture l'idée d'efficacité : lorsqu'une allocation est Pareto-optimale, il est impossible d'augmenter strictement l'utilité d'un agent sans diminuer strictement celle d'un autre, autrement dit les objets sont répartis sans perte, efficacement.

Lorsque l'on engendre une allocation à partir d'une séquence (algorithme 1), d'une certaine façon on applique une forme faible d'efficacité : chaque choix successif est localement optimal. On peut alors se poser la question : toute séquence de choix sincères engendret-elle nécessairement une allocation Pareto-optimale ? Brams *et al.* [8, 10] répondent positivement à cette question en démontrant l'équivalence entre séquençabilité d'une allocation et Pareto-optimalité dans leur modèle de problème de partage. Dans ce modèle, les seules allocations autorisées sont celles qui donnent le même nombre d'objets à chaque agent. Les agents expriment des préférences ordinales sur les objets individuels, et leurs parts sont comparées grâce à une hypothèse de séparabilité sur les préférences.

De manière surprenante, ce résultat d'équivalence ne tient plus dans notre modèle plus général, comme le montre le contre-exemple tout simple suivant.

Exemple 4. Soit l'instance à 2 agents et 2 objets représentée par la matrice de poids

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

La séquence $\langle 1, 1 \rangle$ donne l'allocation $\langle \{1, 2\}, \{\} \rangle$ de vecteur d'utilités $\langle 1, 0 \rangle$, mais l'allocation $\langle \{1\}, \{2\} \rangle$, de vecteur d'utilités $\langle 1, 1 \rangle$, domine la première.

Dans l'exemple 4, les préférences ne sont pas ordonnées identiquement, et les préférences des agents

ne sont pas strictes sur les parts (la part $\{1, 2\}$ a la même utilité que la part $\{1\}$). On peut alors se poser la nouvelle question : lorsque les préférences sont de même ordre et strictes sur les parts, est-ce que toute séquence engendre nécessairement une allocation Pareto-optimale ? La réponse est encore non ! Voici un nouveau contre-exemple.

Exemple 5. Soit l'instance représentée par la matrice suivante (on remarque que les préférences sont ordonnées identiquement et strictes sur les parts) :

$$\begin{pmatrix} 5 & 4 & 2 \\ 8 & 2 & 1 \end{pmatrix}$$

Construisons le tableau de toutes les séquences possibles avec les allocations qu'elles engendrent.

séquence	allocation	utilités	Pareto-opt. ?
$\langle 1, 1, 1 \rangle$	$\langle \{1, 2, 3\}, \{\} \rangle$	$\langle 11, 0 \rangle$	OUI
$\langle 1, 1, 2 \rangle$	$\langle \{1, 2\}, \{3\} \rangle$	$\langle 9, 1 \rangle$	OUI
$\langle 1, 2, 1 \rangle$	$\langle \{1, 3\}, \{2\} \rangle$	$\langle 7, 2 \rangle$	OUI
$\langle 1, 2, 2 \rangle$	$\langle \{1\}, \{2, 3\} \rangle$	$\langle 5, 3 \rangle$	NON
$\langle 2, 1, 1 \rangle$	$\langle \{2, 3\}, \{1\} \rangle$	$\langle 6, 8 \rangle$	OUI
$\langle 2, 1, 2 \rangle$	$\langle \{2\}, \{1, 3\} \rangle$	$\langle 4, 9 \rangle$	OUI
$\langle 2, 2, 1 \rangle$	$\langle \{3\}, \{1, 2\} \rangle$	$\langle 2, 10 \rangle$	OUI
$\langle 2, 2, 2 \rangle$	$\langle \{\}, \{1, 2, 3\} \rangle$	$\langle 0, 11 \rangle$	OUI

La séquence $\langle 1, 2, 2 \rangle$ engendre une allocation dominée par celle engendrée par la séquence $\langle 2, 1, 1 \rangle$.

Les deux exemples précédents nous montrent qu'une séquence de choix sincères n'engendre pas nécessairement une allocation Pareto-optimale. La proposition suivante répond à la question réciproque.

Proposition 6. Toute allocation Pareto-optimale est séquençable.

Avant de livrer la démonstration proprement dite, et pour en faciliter la compréhension, nous allons la concrétiser sur un exemple [7, exemple 5]. L'idée consiste à montrer la contraposée : toute allocation non séquençable est dominée.

Exemple 6. Soit l'instance suivante, donnée par sa matrice de poids.

$$W = \begin{pmatrix} 2 & 12 & 7* & \dagger 15 & * \dagger 11 \\ * \dagger 12 & 15 & \dagger 11 & * 7 & 2 \\ 15 & * \dagger 20 & 9 & 2 & 1 \end{pmatrix}$$

L'allocation $\vec{\pi}$ marquée par des * n'est pas séquençable. En effet, toute séquence susceptible de l'engendrer doit obligatoirement débiter par les agents 3 puis 2, nous laissant la sous-allocation frustrante \vec{p} marquée par des étoiles ci-dessous

$$\begin{pmatrix} 7* & 15 & * 11 \\ 11 & * 7 & 2 \\ 9 & 2 & 1 \end{pmatrix}$$

et à partir de laquelle tout séquençement est impossible. Choisissons un agent quelconque recevant un objet non préféré dans \vec{p} , par exemple l'agent $a_1 = 2$, et soit $o_1 = 3$ cet objet préféré (de poids 11). L'agent qui reçoit o_1 dans \vec{p} est $a_2 = 1$. Cet agent quant à lui préfère l'objet $o_2 = 4$ (de poids 15), détenu par a_1 , déjà rencontré. Ainsi nous avons construit un cycle $(a_1, o_1) \rightarrow (a_2, o_2) \rightarrow (a_1, o_1)$, soit $(2, 3) \rightarrow (1, 4) \rightarrow (2, 3)$, qui nous indique exactement comment construire une sous-allocation qui domine \vec{p} , en remplaçant dans \vec{p} les possessions $(a_1 \leftarrow o_2)(a_2 \leftarrow o_1)$ par les possessions $(a_1 \leftarrow o_1)(a_2 \leftarrow o_2)$. Ainsi chaque agent impliqué dans le cycle obtient un objet strictement meilleur que celui qu'il possédait précédemment. En opérant les mêmes substitutions dans l'allocation d'origine $\vec{\pi}$, on obtient une allocation $\vec{\pi}'$ qui domine $\vec{\pi}$ (marquée par des \dagger dans la matrice W ci-dessus).

Et maintenant la démonstration complète.⁴

Démonstration. Nous allons donc prouver la contraposée : toute allocation non séquençable est dominée. D'après la proposition 1, dans une allocation non séquençable, il existe nécessairement une sous-allocation frustrante. Soit une allocation $\vec{\pi}$ non séquençable, et soit donc \vec{p} une sous-allocation frustrante $\vec{\pi}$ (qui peut être $\vec{\pi}$ elle-même). Nous allons, à partir de \vec{p} , construire une autre sous-allocation qui la domine. Choisissons un agent quelconque impliqué dans \vec{p} , soit a_1 , recevant un objet non préféré dans \vec{p} . Soit o_1 un objet préféré de a_1 dans \vec{p} , et soit a_2 l'unique agent qui reçoit o_1 dans \vec{p} . Remarquons que a_2 est distinct de a_1 , car sinon a_1 aurait obtenu un objet préféré. Soit o_2 un objet préféré de a_2 . Remarquons que o_2 est distinct de o_1 , car sinon a_2 aurait un objet préféré. Soit a_3 l'unique agent qui reçoit o_2 dans \vec{p} . Et ainsi de suite, nous formons une chaîne unique partant de a_1 et qui alterne agents et objets, et dans laquelle deux agents successifs et deux objets successifs sont distincts. Il arrive nécessairement une étape à laquelle on retrouve un agent déjà rencontré. Soit a_i cet agent déjà rencontré, et o_k le dernier objet de la séquence (a_i est l'unique agent qui reçoit o_k ; remarquons que o_k ne peut pas avoir déjà été rencontré sinon on aurait déjà rencontré l'agent a_i dont o_k est un objet préféré). Nous avons construit un cycle $(a_i, o_i) \rightarrow (a_{i+1}, o_{i+1}) \cdots (a_k, o_k) \rightarrow (a_i, o_i)$ dont tous les agents et les objets sont différents, et qui comporte au moins deux agents et deux objets. La sous-allocation cherchée qui domine \vec{p} découle directement du cycle : il suffit d'allouer à chaque agent a_j impliqué dans le cycle l'objet préféré o_j (à la place d'un autre objet strictement moins préféré qui lui était précédemment alloué), les autres agents et objets hors du cycle sans changement. Plus précisément, les possessions de \vec{p} (et donc de $\vec{\pi}$) :

$$(a_i \leftarrow o_k)(a_{i+1} \leftarrow o_i) \cdots (a_k \leftarrow o_{k-1})$$

sont remplacées par :

4. L'idée de cycle est également utilisée par Varian [27, page 79] et Lipton *et al.* [22, Lemma 2.2] dans des démonstrations en rapport avec l'absence d'envie.

$(a_i \leftarrow o_i)(a_{i+1} \leftarrow o_{i+1}) \cdots (a_k \leftarrow o_k)$ où $(a \leftarrow o)$ dénote le fait que a possède o . Les mêmes substitutions opérées dans $\vec{\pi}$ fournissent une allocation $\vec{\pi}'$ qui domine $\vec{\pi}$. \square

Corollaire 1. *Toute allocation frustrante ne peut être Pareto-optimale (de manière équivalente, dans toute allocation Pareto-optimale, un agent au moins reçoit un objet préféré).*

La proposition 6 implique l'existence, pour une instance donnée, de trois classes d'allocations selon leur « degré d'efficacité » : (1) les allocations non-séquençables (donc non Pareto-optimales), (2) les allocations séquençables mais non Pareto-optimales, et (3) les allocations Pareto-optimales (donc séquençables). Ce qui est intéressant et nouveau ici est le degré intermédiaire.

5 Absence d'envie, équilibre compétitif à revenus égaux, et séquençabilité

Il est naturel d'étudier formellement le comportement des protocoles par séquence vis-à-vis des propriétés d'efficacité et d'équité qui souvent motivent leur utilisation. Nous avons étudié dans la section précédente le lien existant entre efficacité et séquençabilité. En ce qui concerne l'équité, nous allons nous pencher sur les critères logiques d'absence d'envie et d'équilibre compétitif, et examiner leur lien avec la séquençabilité.

Pour commencer, la propriété d'absence d'envie [16, 27] est sans doute la plus classique des propriétés d'équité. La voici exprimée dans notre modèle.

Définition 4. *Soit une instance d'un problème de partage de biens indivisibles à préférences numériques additives. On dit que l'allocation $\vec{\pi}$ vérifie la propriété d'absence d'envie (ou est sans envie), lorsque $u_i(\pi_i) \geq u_i(\pi_j)$, $\forall (i, j) \in \mathcal{A}^2$ (aucun agent ne préfère strictement la part d'un autre).*

La notion d'équilibre compétitif est un concept connu depuis longtemps en économie [28, 15]. Le même concept en imposant de plus des revenus égaux devient un équilibre compétitif à revenus égaux (*competitive equilibrium from equal incomes, CEEI*) [23, page 177, par exemple], et fournit un critère d'équité très fort récemment exploré en intelligence artificielle [25, 13, 7]. Voici la définition de ce concept adapté à notre modèle.

Définition 5. *Soit $(\mathcal{A}, \mathcal{O}, W)$ une instance d'un problème de partage de biens indivisibles à préférences numériques additives, $\vec{\pi}$ une allocation, et $\vec{p} \in [0, 1]^M$ un vecteur de prix. On dit que le couple $(\vec{\pi}, \vec{p})$ forme un équilibre compétitif à revenus égaux (*competitive**

equilibrium from equal incomes, CEEI), si

$$\forall i \in \mathcal{A} : \pi_i \in \operatorname{argmax}_{\pi \subseteq \mathcal{O}} \left\{ u_i(\pi) : \sum_{\ell \in \pi} p_\ell \leq 1 \right\}.$$

En d'autres termes, π_i est l'une des parts maximales que i peut s'offrir avec un budget de 1, sachant que le prix de chaque objet ℓ est p_ℓ .

On dira que l'allocation $\vec{\pi}$ est CEEI s'il existe un vecteur \vec{p} tel que $(\vec{\pi}, \vec{p})$ forme un CEEI.

Comme nous l'avons montré dans un article précédent [7], pour notre modèle, toute allocation CEEI est sans envie. On se demande dans cette section si une allocation sans envie ou CEEI est séquençable. La réponse est négative pour l'absence d'envie.

Proposition 7. *Il existe des allocations sans envie non séquençables, même si les préférences des agents sont strictes sur les parts.*

Démonstration. Un contre-exemple avec préférences strictes sur les parts est donné par l'exemple 6 ci-dessus, pour lequel on vérifie facilement que l'allocation $\vec{\pi}$ marquée par des * est sans envie et non séquençable. \square

En ce qui concerne le critère CEEI la réponse est positive, ajoutant encore un intérêt à ce critère d'équité.

Proposition 8. *Toute allocation CEEI est séquençable.*⁵

Démonstration. Montrons qu'il est impossible qu'une allocation soit à la fois non séquençable et CEEI. Soit $\vec{\pi}$ une allocation non séquençable. Nous pouvons reprendre les mêmes termes et notations que la preuve de la proposition 6, en particulier en ce qui concerne le cycle de dominance des agents. Soit \mathcal{C} l'ensemble des agents concernés par le cycle. L'allocation $\vec{\pi}$ contient les parts suivantes :

$$\begin{aligned} \pi_{a_i} &= \{o_k\} \cup \tau_i \\ \pi_{a_{i+1}} &= \{o_i\} \cup \tau_{i+1} \\ &\dots \\ \pi_{a_k} &= \{o_{k-1}\} \cup \tau_k \end{aligned}$$

tandis que l'allocation $\vec{\pi}'$, qui la domine, contient les parts

$$\begin{aligned} \pi'_{a_i} &= \{o_i\} \cup \tau_i \\ \pi'_{a_{i+1}} &= \{o_{i+1}\} \cup \tau_{i+1} \\ &\dots \\ \pi'_{a_k} &= \{o_k\} \cup \tau_k \end{aligned}$$

5. On sait [7, proposition 8] que toute allocation CEEI est Pareto-optimale si les préférences sont strictes sur les parts. D'après la remarque 1 et la proposition 6 du présent article, si les préférences sont strictes sur les parts, alors toute allocation CEEI est séquençable. La présente proposition 8 est plus générale : elle ne fait aucune hypothèse de préférence stricte sur les parts (ni sur les objets).

les autres parts étant inchangées de $\vec{\pi}$ à $\vec{\pi}'$.

Supposons que $\vec{\pi}$ soit CEEI. Cette allocation doit donc vérifier deux types de contraintes. Tout d'abord $\vec{\pi}$ doit respecter la contrainte de prix (en notant $p(\pi) \stackrel{\text{def}}{=} \sum_{\ell \in \pi} p_\ell$)

$$\forall i \in \mathcal{C} : p(\pi_i) \leq 1 \quad (1)$$

Ensuite, $\vec{\pi}$ doit être optimale : toute part de plus d'utilité pour un agent que celle qu'il a dans $\vec{\pi}$ coûte plus que 1. Sachant qu'effectivement $\forall i \in \mathcal{C} : u_i(\pi'_i) > u_i(\pi_i)$ (car $\vec{\pi}'$ substitue des objets strictement préférés à des objets strictement moins préférés dans $\vec{\pi}$), cela se traduit par

$$\forall i \in \mathcal{C} : p(\pi'_i) > 1 \quad (2)$$

En sommant respectivement les équations 1 et 2, sachant que toutes les parts en question sont disjointes, on obtient

$$p(\cup_{j \in \mathcal{C}} \pi_j) \leq |\mathcal{C}| \quad \text{et} \quad p(\cup_{j \in \mathcal{C}} \pi'_j) > |\mathcal{C}|$$

Or $\cup_{j \in \mathcal{C}} \pi_j = \cup_{i \in \mathcal{C}} \pi'_i$ (car l'allocation $\vec{\pi}'$ est obtenue à partir de $\vec{\pi}$ par simple échange d'objets entre les agents de \mathcal{C}). Les deux équations précédentes sont contradictoires. \square

6 Expérimentations

Nous avons mis en évidence, section 4, une « échelle » permettant de caractériser l'efficacité d'une allocation, et constituée de trois paliers : non séquençable (NS), séquençable et non Pareto-optimal (SnP), et Pareto-optimal (PO). On peut se poser naturellement la question de savoir, pour une instance donnée, quelle proportion des allocations se situe à chaque niveau de l'échelle. Nous donnons un début de réponse dans cette section en caractérisant de manière expérimentale la répartition des allocations entre les différents niveaux. De plus, nous analysons le lien entre équité et efficacité en croisant l'échelle d'efficacité avec l'échelle d'équité décrite dans un travail précédent [7], et constituée de cinq niveaux,⁶ du moins équitable au plus équitable : aucun critère satisfait (–), juste part max-min (MFS), proportionnalité (PFS), juste part min-max (mFS), absence d'envie (EF).

Notre protocole expérimental est le suivant. Nous avons généré 100 instances de problèmes de partages à 3 agents et 10 objets, selon le modèle de génération aléatoire uniforme ou gaussien décrit dans notre travail précédent [7]. Pour chacune de ces instances, nous avons généré l'ensemble des allocations possibles (soit $3^{10} = 59049$) et identifié le niveau d'équité et d'efficacité de chacune. Le résultat apparaît en figure 1.

Dans cette figure, les histogrammes représentent la valeur moyenne (sur l'ensemble des instances) du

6. L'échelle initiale comporte en fait un niveau supplémentaire (CEEI), que nous n'utilisons pas ici car il est computationnellement difficile de caractériser les allocations CEEI.

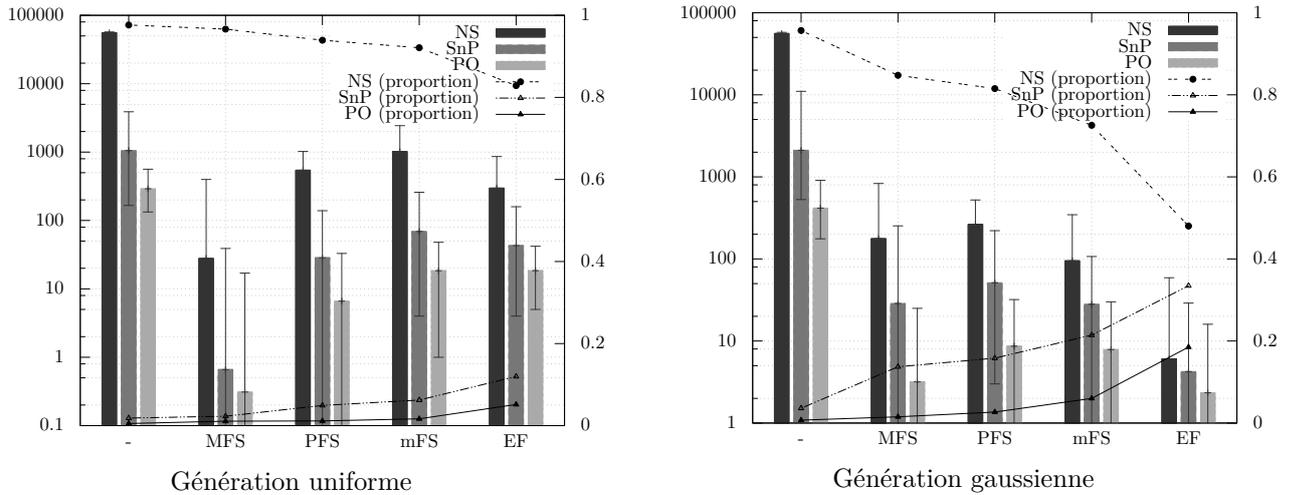


FIGURE 1 – Répartition du nombre d'allocations par couple de critères équité / efficacité satisfaits.

nombre d'allocations par couple de critères (avec l'intervalle min-max représenté sous forme de barre d'erreur), en échelle logarithmique. Les trois courbes représentent la proportion moyenne, par critère d'équité, du nombre d'allocations satisfaisant les trois critères d'efficacité donnés, en échelle linéaire.

Nous pouvons observer plusieurs faits intéressants. Tout d'abord, l'immense majorité des allocations ne possède ni des bonnes propriétés d'efficacité, ni des bonnes propriétés d'équité, comme en témoigne le nombre d'allocations non séquençables et ne possédant aucune propriété d'équité. Deuxième remarque, la répartition des allocations sur l'échelle d'efficacité semble corrélée au critère d'équité satisfait : ainsi, on trouve une plus grande proportion d'allocations séquençables parmi les allocations sans envie que parmi celles qui ne satisfont aucun critère d'équité. Enfin, il existe une plus forte proportion d'allocations satisfaisant de bonnes propriétés d'équité et d'efficacité pour le modèle de génération uniforme. Si cela semble assez logique pour l'échelle d'équité (car une attirance « similaire » pour les objets — ce qui est le cas pour le modèle gaussien — a une plus forte chance de créer des conflits entre agents), l'explication n'est pas si claire pour l'échelle d'efficacité.

7 Conclusion

Il est relativement fréquent d'avoir à partager un lot d'objets indivisibles entre plusieurs individus, et pour remplacer la loi du plus fort, l'usage d'un protocole donnant la main séquentiellement aux participants est un procédé naturel et d'usage répandu, certainement vieux comme le monde. Le cas emblématique mais aucunement isolé est celui du partage d'un patrimoine

entre plusieurs héritiers, lorsqu'ils souhaitent éviter le recours à des compensations monétaires.

Dans cet article, nous avons exploré ce protocole de partage dans un modèle simple et communément admis, dans lequel les agents expriment leurs préférences par des utilités numériques additives.

Nous avons montré qu'une allocation donnée est séquençable (c'est-à-dire peut être obtenue par l'application d'une séquence précise) si et seulement si elle ne contient aucune « sous-allocation frustrante » — dans laquelle aucun agent n'obtient un de ses objets préférés.

La manière dont les agents expriment individuellement ou mutuellement leurs préférences a été utilisée pour caractériser la correspondance entre l'ensemble des allocations et des séquences possibles d'une instance, en particulier pour caractériser les instances pour lesquelles toute allocation est séquençable.

Il est par ailleurs naturel d'étudier formellement les protocoles par séquence en rapport avec deux exigences souvent opposées : d'une part l'efficacité, et d'autre part l'équité. À cet égard, nous avons montré que toute allocation Pareto-optimale est séquençable mais que l'inverse n'est pas vrai. D'où l'existence d'une « échelle d'efficacité » comportant trois paliers : Pareto-optimalité, séquençabilité sans Pareto-optimalité, et non séquençabilité. D'autre part, en rapport avec l'échelle de critères logiques d'équité étudiés dans un précédent travail [7], nous avons montré qu'il existe des allocations sans envie non séquençables, mais que toute allocation de type équilibre compétitif à revenus égaux (CEEI) est séquençable. Des résultats expérimentaux montrent les corrélations entre ces deux échelles.

Malgré leur relative simplicité, tous ces résultats

sont nouveaux à notre connaissance.

Les prolongements de ces travaux préliminaires se situent dans l'étude de séquences particulières adaptées aux exigences d'équité (séquences alternées par exemple) ainsi qu'à des investigations sur d'autres modalités d'expression des préférences des agents (préférences à niveaux, préférences ordinales).

Références

- [1] N. Bansal and M. Sviridenko. The Santa Claus problem. In *Proc of STOC'06*, 2006.
- [2] N. Bianchessi et al. A heuristic for the multisatellite, multiorbit and multiuser management of earth observation satellites. *EJOR*, 177(2), 2007.
- [3] S. Bouveret, U. Endriss, and J. Lang. Conditional importance networks : A graphical language for representing ordinal, monotonic preferences over sets of goods. In *Proc. of IJCAI'09*, 2009.
- [4] S. Bouveret and J. Lang. Efficiency and envy-freeness in fair division of indivisible goods : Logical representation and complexity. *JAIR*, 32 :525–564, 2008.
- [5] S. Bouveret and J. Lang. A general elicitation-free protocol for allocating indivisible goods. In *Proc. of IJCAI'11*, 2011.
- [6] S. Bouveret and M. Lemaître. Computing leximin-optimal solutions in constraint networks. *Artificial Intelligence*, 173(2) :343–364, 2009.
- [7] S. Bouveret and M. Lemaître. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *JAAMAS*, 2015.
- [8] S. J. Brams, P. H. Edelman, and P. C. Fishburn. Fair division of indivisible items. *Theory and Decision*, 5(2), 2004.
- [9] S. J. Brams, D. M. Kilgour, and C. Klumler. The undercut procedure : an algorithm for the envy-free division of indivisible items. *Social Choice and Welfare*, 39(2-3) :615–631, 2012.
- [10] S. J. Brams and D. King. Efficient fair division – help the worst off or avoid envy? *Rationality and Society*, 17(4), 2005.
- [11] S. J. Brams and Alan D. Taylor. *Fair Division – From Cake-cutting to Dispute Resolution*. Cambridge Univ. Press, 1996.
- [12] S. J. Brams and Alan D. Taylor. *The Win-win Solution. Guaranteeing Fair Shares to Everybody*. W. W. Norton & Company, 2000.
- [13] E. Budish. The combinatorial assignment problem : Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6), 2011.
- [14] Y. Chevaleyre, U. Endriss, and N. Maudet. On maximal classes of utility functions for efficient one-to-one negotiation. In *Proc. of IJCAI'05*, 2005.
- [15] I. Fisher. *Mathematical Investigations in the Theory of Value and Prices, and Appreciation and Interest*. Augustus M. Kelley, Publishers, 1892.
- [16] D. Foley. Resource allocation and the public sector. *Yale Econ. Essays*, 7(1), 1967.
- [17] D. K. Herreiner and C. Puppe. A simple procedure for finding equitable allocations of indivisible goods. *Social Choice and Welfare*, 19, 2002.
- [18] T. Kalinowski, N. Narodytska, and T. Walsh. A social welfare optimal sequential allocation procedure. In *Proc. of IJCAI'13*, 2013.
- [19] T. Kalinowski, N. Narodytska, T. Walsh, and L. Xia. Strategic behavior when allocating indivisible goods sequentially. In *Proc. of AAAI'13*, 2013.
- [20] David A Kohler and R Chandrasekaran. A class of sequential games. *Operations Research*, 19(2) :270–277, 1971.
- [21] M. Lemaître, G. Verfaillie, and N. Bataille. Exploiting a common property resource under a fairness constraint : a case study. In *Proc. of IJCAI'99*, 1999.
- [22] R. Lipton, E. Markakis, E. Mossel, and A. Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of EC'04*, 2004.
- [23] H. Moulin. *Fair Division and Collective Welfare*. MIT Press, 2003.
- [24] N. Nguyen, T. T. Nguyen, M. Roos, and J. Rothe. Computational complexity and approximability of social welfare optimization in multiagent resource allocation. *JAAMAS*, 28(2) :256–289, 2014.
- [25] A. Othman, T. Sandholm, and E. Budish. Finding approximate competitive equilibria : efficient and fair course allocation. In *Proceedings of AAMAS'10*, 2010.
- [26] T. W. Sandholm. Contract types for satisficing task allocation : I. theoretical results. In Sandip Sen, editor, *Proceedings of the AAAI Spring Symposium : Satisficing Models*, pages 68–75, Menlo Park, California, 1998. AAAI Press.
- [27] H. R. Varian. Equity, Envy and Efficiency. *Journal of Economic Theory*, 9 :63–91, 1974.
- [28] L. Walras. *Éléments d'économie politique pure ou Théorie de la richesse sociale*. L. Corbaz, 1 edition, 1874.

Une algèbre des relations temporelles granulaires pour le raisonnement qualitatif*

Quentin Cohen-Solal Maroua Bouzid Alexandre Niveau

GREYC-CNRS
U. Caen, France

{quentin.cohen-solal, maroua.bouzid-mouaddib, alexandre.niveau}@unicaen.fr

Résumé

Dans cet article, nous proposons un formalisme qualitatif pour représenter et raisonner sur le temps à différentes échelles, appelées granularités. En particulier, nous étendons certains des résultats d'Euzenat [1995] à la prise en compte des relations entre les points et les intervalles. Notre formalisme est plus expressif que la plupart des algèbres de relations temporelles proposées dans la littérature. En effet, certaines de nos relations sont plus restrictives que celles de l'algèbre d'Allen [1983], tandis que d'autres le sont moins. En particulier, ce formalisme permet la modélisation de relations imprécises, graduelles et intuitives, telles que « juste avant » ou « touche presque ». En outre, nous exposons plusieurs résultats concernant la conversion d'une relation d'une granularité à une autre. Enfin, nous présentons un algorithme qui calcule la clôture algébrique d'un réseau de contraintes temporelles dans le cadre de notre formalisme, pouvant être utilisé pour en vérifier la cohérence.

Abstract

In this paper, we propose a qualitative formalism for representing and reasoning about time at different scales. It extends some of the results of Euzenat [1995] and overcomes their major limitations, allowing one to reason about relations between points and intervals. Our approach is more expressive than most algebras of temporal relations : for instance, some relations are more relaxed than those in Allen's [1983] algebra, while others are stricter. In particular, it enables the modeling of imprecise, gradual, or intuitive relations, such as "just before" or "almost meet". In addition, we give several results about how a relation changes when considered at different granularities. Finally, we provide an algorithm to compute the algebraic closure of a temporal constraint

network in our formalism, which can be used to check its consistency.

1 Introduction

Les agents autonomes doivent souvent être capables de raisonner sur le temps ; l'étude des formalismes temporels qualitatifs est un moyen de parvenir à ce but. Les plus populaires de ces formalismes sont les algèbres des relations temporelles, dans lesquelles les localisations temporelles des événements (appelés *primitives* ou *entités*) sont définies relativement les unes par rapport aux autres, sans aucune donnée quantitative. L'*algèbre des points* de Vilain *et al.* [1986] (plus tard étudiée par Ladkin et Maddux [1994] et Hirsch [1996]) définit trois relations élémentaires entre des points. L'*algèbre des intervalles* d'Allen [1983] caractérise 13 relations élémentaires entre deux intervalles. L'*algèbre des points et des intervalles* de Vilain [1982], développée par Meiri [1996] et Krokhin et Jonsson [2002], étend simultanément ces deux dernières algèbres en ajoutant des relations entre un point et un intervalle, pour un total de 26 relations. Dans chacune de ces trois algèbres, des *opérateurs* permettent de déduire de nouvelles relations entre les entités, et donc de raisonner sur des événements.

Cependant, ces formalismes ne considèrent ni des données ayant des précisions différentes, ni le concept de représentation et raisonnement multi-échelle. Ce concept permet de représenter les événements à différents niveaux de détail, appelés *granularités*, tels que les jours, les mois ou les semaines. Grâce aux granularités, il est possible d'organiser les connaissances hiérarchiquement et de raisonner à chaque échelle. Par exemple, dans le cadre des relations temporelles, on peut dire qu'un intervalle A rencontre un autre inter-

*Ceci est la version française de l'article « An Algebra of Granular Temporal Relations for Qualitative Reasoning », à paraître dans les actes de la conférence IJCAI 2015.

valle B à une granularité grossière (*i.e.* d'un point de vue général), mais que A est avant B à une granularité fine (*i.e.* d'un point de vue plus précis). Les algèbres usuelles ne permettent pas de traiter de telles connaissances sans aboutir à des incohérences. Afin de surmonter ce problème, Euzenat a proposé une extension granulaire de l'algèbre des points ainsi qu'une extension de l'algèbre des intervalles [Euzenat, 1995]. Chacune d'entre elles fournit une table décrivant comment les relations changent quand elles sont considérées à une granularité plus fine (conversion descendante) ou à une granularité plus grossière (conversion ascendante).

Toutefois, la table de conversion des relations entre intervalles n'est pas toujours applicable : si un intervalle devient un point lorsqu'il est considéré à une granularité plus grossière, la conversion ascendante d'une relation entre cet intervalle et un autre est impossible, puisque seules les relations entre intervalles sont autorisées. Par exemple, considérons deux intervalles $A = [01:30, 03:10]$ et $B = [03:15, 05:07]$ ayant lieu le même jour. A la granularité des minutes, A est avant B , tandis qu'à la granularité des heures, A rencontre B . A la granularité des jours, A et B sont indiscernables ; la relation entre A et B est donc la relation d'« égalité des points », qui n'existe pas dans l'algèbre d'Allen, et n'est donc pas présente dans la table de conversion d'Euzenat. En conséquence, les conditions pour que la conversion ascendante soit applicable dépendent, en particulier, de la durée des intervalles, qui n'est pas toujours connue. L'algèbre d'Allen et donc son extension granulaire ne décrivent pas toutes les relations possibles lorsqu'il y a plus d'une granularité : de ce fait, raisonner est impossible lorsque certaines granularités sont très grossières.

Pour surmonter ces limitations, nous proposons dans cet article une extension granulaire de l'algèbre des points et des intervalles, dans le but de représenter et de raisonner sur des connaissances temporelles imprécises, provenant par exemple de données exprimées en langage naturel ou de sources n'ayant pas les mêmes échelles temporelles. De ce point de vue, notre formalisme peut être considéré comme une alternative à l'algèbre des intervalles flous de Schockaert *et al.* [2006], dans laquelle il est aussi possible de modéliser des relations imprécises. En particulier, nous nous intéressons aux granularités qualitatives, c'est-à-dire aux granularités pour lesquelles aucune information quantitative (telle que la durée des granules) n'est spécifiée : nous n'utilisons que leurs échelles relatives. Ces granularités qualitatives permettent plus de flexibilité dans le processus de modélisation et sont, par exemple, nécessaires pour analyser une information temporelle exprimée dans le langage naturel, où les granularités ne sont pas exactement connues et ne peuvent donc

pas être quantitativement modélisées.

La section suivante rappelle les concepts principaux des granularités temporelles que nous utilisons dans cet article. Ensuite, dans la troisième section, nous introduisons les relations granulaires entre points et intervalles. La quatrième section présente notre algèbre des relations temporelles granulaires et ses opérateurs, incluant les conversions granulaires. Dans la section 5, nous proposons un algorithme polynomial pour calculer la clôture algébrique d'un réseau de contraintes temporelles granulaires, qui peut être utilisé pour vérifier sa cohérence dans le cadre des granularités qualitatives. Dans la sixième section, nous discutons des travaux connexes. Enfin, la conclusion et nos perspectives de travaux futurs sont présentées dans la Section 7.

2 Granularités temporelles

2.1 Définitions

Plusieurs travaux formalisant les granularités temporelles ont été développés. Notre formalisme s'appuie sur le cadre de Bettini *et al.* [2000], tel que présenté par Euzenat et Montanari [2005]. Cette approche utilise la théorie des ensembles pour définir les granularités. Dans ce qui suit, nous rappelons les concepts de base dont nous avons besoin pour définir notre formalisme.

Le temps est modélisé par un *domaine temporel* T , qui est un ensemble totalement ordonné, discret ou dense. Ses éléments sont appelés *points temporels*.

Définition 1. Une granularité g est une fonction d'un ensemble ordonné discret I_g vers l'ensemble des parties du domaine temporel T telle que :

$$\begin{aligned} \forall i, j, k \in I_g, \quad i < k < j \wedge g_i \neq \emptyset \wedge g_j \neq \emptyset &\implies g_k \neq \emptyset \\ \forall i, j \in I_g, \quad i < j &\implies \forall x \in g_i, \forall y \in g_j, x < y \end{aligned}$$

Ainsi, chaque granularité est une séquence de sous-ensembles du domaine temporel qui préserve l'ordre naturel des points temporels. Les ensembles g_i sont appelés des *granules* de la granularité g . Par abus de notation, on considère que T est lui-même une granularité, représenté par la fonction $T: i \in T \mapsto \{i\}$, c'est-à-dire que $I_T = T$ (même s'il n'est pas discret) et $T_i = \{i\}$.

Bettini *et al.* ont défini plusieurs relations entre granularités ; nous nous intéressons à deux d'entre elles en particulier.

Définition 2. Soient g, h deux granularités sur T .

- g est *plus fine* que h , noté $g \preceq h$, si et seulement si $\forall i \in I_g, \exists j \in I_h: g_i \subseteq h_j$, *i.e.* chaque granule de g est inclus dans un granule de h .

- h est *plus grossière* que g^1 , noté $g \sqsubseteq h$, si et seulement si $\forall j \in I_h, \exists S \subseteq I_g : h_j = \bigcup_{i \in S} g_i$, i.e. chaque granule de h est composé de granules de g .

Les relations \preceq et \sqsubseteq sont des ordres partiels ; elles donnent deux caractérisations différentes de la notion intuitive d'être « plus précis », chacune nécessitant que les deux granularités soient alignées. Si $g \preceq h$ et $g \sqsubseteq h$, alors l'ensemble des granules de g est une partition de l'ensemble des granules de h .

Bettini *et al.* ont aussi défini deux opérateurs de conversion entre granularités. Ils associent à un granule d'une granularité donnée un ensemble correspondant de granules d'une autre granularité. La conversion ascendante \uparrow , qui convertit un granule vers une granularité « moins précise », est définie par :

$$\forall i \in I_g, \uparrow_g^h i = \{j \in I_h : g_i \subseteq h_j\}$$

La conversion descendante \downarrow , qui convertit un granule vers une granularité « plus précise », est définie par :

$$\forall j \in I_h, \downarrow_g^h j = \left\{ S \subseteq I_g : h_j = \bigcup_{i \in S} g_i \right\}$$

Proposition 3. Soient g et h deux granularités sur T , et soient $i \in I_g$ et $j \in I_h$. Si $g \preceq h$ (resp. $g \sqsubseteq h$), alors $\uparrow_g^h i$ (resp. $\downarrow_g^h j$) contient exactement un élément.

2.2 Calendrier grégorien

Pour représenter des relations entre événements, nous devons d'abord choisir un ensemble de granularités. Le calendrier grégorien, dans lequel les échelles des mois, années, semaines et jours sont des granularités (les échelles des heures, minutes et secondes peuvent facilement être ajoutées), est sûrement le plus utilisé. Pour chaque couple (g, h) de ces granularités tel que g est (intuitivement) « plus précise » que h , les propriétés $g \preceq h$ et $g \sqsubseteq h$ sont toutes deux vérifiées (par exemple, « mois \preceq années » et « mois \sqsubseteq années »), sauf pour les couples semaines-mois et semaines-années : puisque leurs granules ne sont pas alignés, les semaines ne sont pas plus fines que les mois et les années, et les mois et les années ne sont pas plus grossiers que les semaines.

Notons que les granularités ne sont pas nécessairement une partition du domaine temporel, puisqu'il peut y avoir des trous. En effet, si nous considérons la granularité des semaines de travail, dans laquelle les granules sont les périodes du lundi au vendredi (les week-ends sont exclus), nous avons « semaines de travail \preceq semaines » et « jours \sqsubseteq semaines de travail »,

1. Cette relation est appelée « g groups into h » par Bettini *et al.*

mais nous n'avons pas « semaines de travail \sqsubseteq semaines » et « jours \preceq semaines de travail » [Bettini *et al.*, 2000].

3 Relations temporelles avec des granularités

Dans cette section, nous commençons par définir les relations entre points dans le contexte des granularités temporelles, basées sur le cadre formel de Bettini *et al.* [2000]. Ensuite, nous nous intéressons aux relations granulaires entre points et intervalles. Enfin, nous montrons certaines des possibilités de modélisation offertes par notre formalisme.

3.1 Relations granulaires entre points

Nous commençons par définir formellement les relations granulaires entre points, ce qui n'avait pas été fait dans le travail original d'Euzenat [1995]. Grâce à cette formalisation, nous pouvons exhiber des conditions suffisantes pour utiliser nos tables de conversions granulaires (et également celles d'Euzenat). Dans notre formalisme, nous considérons trois *relations élémentaires* possibles entre deux points à une granularité donnée, définies comme suit :

Définition 4. Soient $a, b \in T$. Les relations granulaires entre points « avant à la granularité g » $<^g$, « égal à la granularité g » $=^g$, et « après à la granularité g » $>^g$ sont définies par :

$$a <^g b \iff \exists i, j \in I_g : i < j \wedge a \in g_i \wedge b \in g_j$$

$$a =^g b \iff \exists i \in I_g : a, b \in g_i$$

$$a >^g b \iff \exists i, j \in I_g : i > j \wedge a \in g_i \wedge b \in g_j$$

On dit qu'un point temporel a est *représentable* à une granularité g si $a \in \bigcup_{i \in I_g} g_i$, et on le note par $a \propto g$. Lorsque deux points temporels sont représentables à une granularité g , alors les trois relations sont exhaustives : si $a \propto g$ et $b \propto g$, alors $a <^g b$ ou $a =^g b$ ou $a >^g b$. Elles sont aussi exclusives puisque l'on ne peut avoir plus d'une relation granulaire élémentaire entre deux points.

3.2 Relations granulaires entre points et intervalles

Nous définissons une entité temporelle E comme un couple de points temporels $(e^-, e^+) \in T^2$ tel que $e^- \leq e^+$. Ce couple représente l'ensemble des points situés entre e^- et e^+ dans T , c'est-à-dire $\{t \in T : e^- \leq t \leq e^+\}$. Une entité temporelle E est dite représentable à une granularité g si $e^- \propto g$ et $e^+ \propto g$; nous le noterons par $E \propto g$. À une granularité g , E peut être de deux

r_{AB}	$a^- R a^+$	$b^- R b^+$	$a^- R b^-$	$a^- R b^+$	$a^+ R b^-$	$a^+ R b^+$
$<$	$=$	$=$	$<$	$<$	$<$	$<$
$\cdot a$	$=$	$<$	$>$	$>$	$>$	$>$
$\cdot b$	$=$	$<$	$<$	$<$	$<$	$<$
b	$<$	$<$	$<$	$<$	$<$	$<$
o	$<$	$<$	$<$	$<$	$>$	$<$
m	$<$	$<$	$<$	$<$	$=$	$<$
d	$<$	$<$	$>$	$<$	$>$	$<$
$\cdot d$	$=$	$<$	$>$	$<$	$>$	$<$
s	$<$	$<$	$=$	$<$	$>$	$<$
$\cdot s$	$=$	$<$	$=$	$<$	$=$	$<$
f	$<$	$<$	$>$	$<$	$>$	$=$
$\cdot f$	$=$	$<$	$>$	$=$	$>$	$=$
e	$<$	$<$	$=$	$<$	$>$	$=$
$=$	$=$	$=$	$=$	$=$	$=$	$=$

TABLE 1 – Définitions des relations granulaires entre intervalles et points à partir des relations granulaires entre points à la même granularité (les exposants sont omis pour des raisons de clarté).

types : si $e^- =^g e^+$, alors E est un point, tandis que si $e^- <^g e^+$, alors E est un intervalle.

Il y a ainsi trois catégories de relations élémentaires possibles entre deux entités à une granularité g donnée. La première concerne les relations d'intervalles : est avant (*before*), rencontre (*meets*), chevauche (*overlaps*), démarre (*starts*), est pendant (*during*), et finit (*finishes*), notées b^g , m^g , o^g , s^g , d^g , et f^g ; leurs inverses, notés \bar{r}^g (avec r^g une de ces relations); et la relation égal, notée e^g . La deuxième concerne les relations entre points : avant $<^g$, égal $=^g$, et après $>^g$. Enfin, la troisième catégorie concerne les relations entre un point et un intervalle : avant $\cdot b^g$, durant $\cdot d^g$, démarre $\cdot s^g$, finit $\cdot f^g$, et après $\cdot a^g$; et leurs inverses, les relations entre un intervalle et un point, qui sont aussi notées par \bar{r}^g , avec r^g la relation correspondante entre un point et un intervalle.

Les définitions de ces relations entre entités temporelles à une granularité donnée se trouvent dans la table 1; ce sont les mêmes que dans le cadre non granulaire, c'est-à-dire sur le domaine temporel [Krokhin et Jonsson, 2002]. Par exemple, soit $A = (a^-, a^+)$ et $B = (b^-, b^+)$ deux entités temporelles; $A m^g B$ est vrai si $a^- <^g a^+$, $b^- <^g b^+$, et $a^+ =^g b^-$. Pour obtenir les définitions des relations inverses, il faut remplacer « $<$ » par « $>$ » et « $>$ » par « $<$ » dans les colonnes 4 à 7, puis permuter les colonnes 2 et 3 ainsi que les colonnes 5 et 6. Dans la suite, on notera par \mathcal{R} l'ensemble des 26 relations élémentaires.

Dans de nombreuses situations, les bornes des entités ne sont pas connues, et généralement, même la relation élémentaire entre deux entités n'est pas connue. Souvent, en effet, de nombreuses relations élémentaires sont possibles, et il n'y a aucun moyen d'en choisir une.

La définition suivante est utilisée pour représenter de telles ambiguïtés.

Définition 5. Une *relation générale* R est un ensemble de relations élémentaires : $R = \{r_1, \dots, r_n\} \subseteq \mathcal{R}$. On dit que R est vraie entre deux entités temporelles A, B à la granularité g , noté $A R^g B$ ou $A (r_1 \cdots r_n)^g B$, si et seulement si l'expression $\bigvee_{i=1}^n A r_i^g B$ est vraie.

Par exemple, la relation « à la granularité h , A est un intervalle et est avant ou rencontre B , ou A est un point et est avant B qui est un intervalle » est représentée par $A (\cdot b b m)^h B$.

3.3 Modéliser des relations temporelles imprécises

Les relations granulaires sont généralement imprécises. En fait, quand on change de granularité, les relations peuvent devenir ambiguës (voir section 4.1). Ainsi, on peut interpréter la relation granulaire « rencontre » comme « rencontre presque », avec une précision qui dépend de la granularité. Par exemple, on peut modéliser « rencontre à peu près » en utilisant une certaine granularité et « rencontre pratiquement » en utilisant une granularité plus fine. Les granularités permettent de relaxer les relations entre points et intervalles, ce qui est utile pour modéliser des phénomènes naturels ayant une synchronisation imparfaite ou des relations temporelles exprimées en langage naturel.

De plus, en combinant différentes relations granulaires exprimées à différentes granularités entre deux mêmes entités, nous pouvons modéliser de nouvelles relations intuitives et imprécises. La plupart de ces

relations combinées sont plus restrictives que les relations classiques. Soient g et h des granularités telles que g est « plus précise » que h ; la relation intuitive « juste avant » peut être modélisée par « $A (\cdot b)^g B$ et $A (=)^h B$ », ou par « $A (b)^g B$ et $A (m)^h B$ », ou par « $A (\bar{a})^g B$ et $A (\bar{f})^h B$ », etc., en fonction des types des entités temporelles aux deux granularités (si les types sont inconnus, une relation générale peut être utilisée : « $A (\cdot bb)^g B$ et $A (=e)^h B$ », etc.). Suivant l'application envisagée, il peut être important de garder à l'esprit que cette modélisation de « juste avant » a un sens plus fort que celui du sens commun. Cependant, elle est pertinente, car en langage naturel, la précision associée aux relations est inconnue, comme elle peut varier d'une personne à une autre. Cela peut être représenté dans notre formalisme en utilisant les granularités qualitatives, c'est-à-dire en ne spécifiant aucune information à propos de la structure des granularités, excepté leurs précisions relatives grâce aux relations \preceq et \sqsubseteq . Les granularités qualitatives permettent de modéliser et d'utiliser un concept de proximité non métrique.

4 Algèbre des relations temporelles granulaires et raisonnement

Dans cette section, nous définissons les opérateurs de notre algèbre et leurs règles, qui nous permettent de déduire de nouvelles relations. Ensuite, nous montrons les avantages des relations granulaires entre points et intervalles pour le raisonnement.

4.1 Conversion entre granularités

Savoir qu'une relation R est vraie à une granularité g entre A et B peut informer sur la relation entre A et B à une autre granularité. De manière plus intuitive, si A est avant B à une certaine granularité, B ne peut être avant A à une autre granularité. Cependant, l'information déduite manque souvent de précision. En effet, le changement de granularité peut induire une perte d'information. Ainsi, convertir une relation élémentaire à une granularité différente peut donner une relation générale. Dans la suite, nous introduisons une table de conversion, qui montre comment on peut déduire de l'information d'une granularité à une autre. Ensuite, nous présentons des conditions suffisantes pour que cette table puisse s'appliquer.

Définition 6. Pour chaque relation élémentaire non inverse $r \in \mathcal{R}$, nous définissons la *conversion ascendante* $\uparrow r$ et la *conversion descendante* $\downarrow r$ de r comme une relation générale en utilisant la table 2. La conversion d'une relation élémentaire inverse est ensuite définie par $\downarrow \bar{r} = \bigcup_{s \in \downarrow r} \bar{s}$ et $\uparrow \bar{r} = \bigcup_{s \in \uparrow r} \bar{s}$.

r	$\uparrow r$	$\downarrow r$
b	$b \cdot \bar{a} \cdot b < \bar{f} \cdot s m =$	b
$\cdot a$	$\cdot a > \cdot f =$	$\cdot a \bar{b}$
$\cdot b$	$\cdot b < \cdot s =$	$\cdot b b$
$<$	$< =$	$< \cdot b \cdot \bar{a} b$
o	$o m \cdot s s \cdot \bar{f} \bar{f} e =$	o
m	$m \cdot s \cdot \bar{f} =$	$m o b$
d	$\cdot s s \cdot d d \cdot f f e =$	d
$\cdot d$	$\cdot s \cdot d \cdot f =$	$\cdot d d$
s	$s \cdot s e =$	$s o d$
$\cdot s$	$\cdot s =$	$m o \cdot s s \cdot d d \cdot b b$
f	$f \cdot f e =$	$f d \bar{o}$
$\cdot f$	$\cdot f =$	$\bar{m} \bar{o} \cdot d d \cdot f f \cdot a \bar{b}$
e	$e =$	$e o \bar{o} s \bar{s} f \bar{f} d \bar{d}$
$=$	$=$	\mathcal{R} (toutes)

TABLE 2 – Conversion des relations entre points et intervalles.

Finalement, la conversion d'une relation générale est donnée par $\uparrow R = \bigcup_{r \in R} \uparrow r$ et $\downarrow R = \bigcup_{r \in R} \downarrow r$.

Théorème 7. Soient A et B deux entités temporelles, g et h deux granularités, et R une relation générale.

- Si $g \sqsubseteq h$, alors $A R^h B \implies A (\downarrow R)^g B$.
- Si $g \preceq h$, alors $A R^g B \implies A (\uparrow R)^h B$.

Démonstration. Nous représentons chaque relation élémentaire entre points et intervalles comme une conjonction de relations entre les bornes des entités, en utilisant la table 1. Ensuite, en combinant la propriété « g est plus fine que h » ou « g est plus grossière que h » avec la définition des granularités, nous obtenons une conjonction de disjonctions de relations de points entre les bornes des entités, que nous pouvons reconverter en une disjonction de relations entre points et intervalles, grâce à la distributivité entre conjonction et disjonction.

Par exemple, si $A s^g B$, alors $b^- =^g a^- <^g a^+ <^g b^+$ et ainsi il existe $i, j, k \in I_g$ tels que $a^-, b^- \in g_i$, $a^+ \in g_j$, $b^+ \in g_k$, et $i < j < k$. Selon la définition 1, comme $i < j < k$, nous trouvons $a^-, b^- < a^+ < b^+$. Puisque $g \preceq h$, il existe $i', j', k' \in I_h$ tels que $g_i \sqsubseteq h_{i'}$, $g_j \sqsubseteq h_{j'}$, et $g_k \sqsubseteq h_{k'}$. Par conséquent, $a^-, b^- \in h_{i'}$ et ainsi $a^- =^h b^-$; de plus, par contraposition du second point de la définition 1, nous déduisons que $i' \leq j' \leq k'$. Par conséquent, $a^- =^h b^- \leq^h a^+ \leq^h b^+$,

ce qui correspond à quatre des lignes de la table 1. Finalement, nous concluons que $A (s \cdot se =)^h B$.

Concernant les relations générales, puisque $A R^g B \iff \bigvee_{r \in R} A r^g B$, en utilisant le même mécanisme que ci-dessus et sachant que la disjonction est commutative et associative, on peut montrer le résultat. \square

Comme exemple d'utilisation de ce théorème, considérons la relation $A (\cdot b b m)^h B$ mentionnée dans la section 3.2 : si $g \sqsubseteq h$, on peut conclure que $A (m o b \cdot b)^g B$ par conversion descendante.

Ce théorème montre que les conversions granulaires de relations entre entités temporelles ont des conditions suffisantes qui sont faciles à vérifier². De plus, nous pouvons aussi définir une table de conversion entre granularités qui ne sont pas alignées. Cependant, cela n'est pas nécessaire, car une telle conversion reviendrait à appliquer une conversion alignée descendante suivi d'une conversion alignée ascendante, ce qui est presque toujours possible en utilisant le domaine temporel comme intermédiaire, grâce à la propriété suivante.

Proposition 8. *Soit g une granularité sur T .*

- $T \sqsubseteq g$ est toujours vraie.
- Si $\bigcup_{i \in I_g} g_i = T$ alors $T \preceq g$.

Ainsi, dans le cas du calendrier grégorien ou de celui des granularités qui sont généralement utilisées dans le cadre du langage naturel, $T \preceq g$ et $T \sqsubseteq g$ sont vérifiées. Par conséquent, pour la majorité des applications que nous envisageons, notre table peut être utilisée même si les conditions ne sont pas satisfaites, ce qui rend possible, par exemple, une conversion entre les semaines et les mois ou les semaines et les années.

4.2 Le raisonnement dans le cadre de notre algèbre temporelle

Les algèbres temporelles permettent de faire des déductions à propos des relations entre entités temporelles grâce à leurs opérateurs. Les algèbres temporelles classiques ont trois opérateurs : la composition \circ , l'intersection \cap , et l'inversion $\bar{\cdot}$. Dans le cadre de notre formalisme, ces opérateurs ne sont utilisés qu'entre relations définies à la même granularité (voir leur définition par Meiri [1996]). De plus, la composition classique permet de déduire le type des entités temporelles

2. Ces conditions suffisantes s'appliquent aussi à la table de conversion des relations entre points d'Euzenat, et également à sa table de conversion des relations entre intervalles — mais uniquement si les intervalles ne deviennent pas des points lorsque la granularité de destination est plus grossière ; cependant, cette dernière condition nécessite de connaître les durées des granules et des intervalles.

(point ou intervalle) s'il y a une ambiguïté. Pour définir notre algèbre, nous ajoutons à ces trois opérateurs nos opérateurs de conversion granulaire \uparrow et \downarrow (voir table 2), qui permettent de convertir des relations d'une granularité à une autre. Les règles pour déduire de nouvelles relations à partir des opérateurs sont résumées dans la table 3.

4.3 Raisonner avec des relations imprécises

Comme expliqué dans la section 3.3, grâce aux relations définies sur plusieurs granularités, on peut représenter, par exemple, la relation « A est juste avant B et B finit peu après » par $A b^{mn} B$ et $A (=)^{hour} B$, et la relation « A est réellement avant C » par $A (\cdot b b)^{hour} C$. Il est intéressant de noter qu'en utilisant nos opérateurs, on déduit que $B b^{mn} C$, alors que cette conclusion n'aurait pas pu être obtenue si nous n'avions que $A b B$ et $A b C$ (comme cela serait le cas sans granularités).

Par ailleurs, contrairement à ce que l'on pourrait penser, il n'est pas suffisant de convertir toutes les relations à la granularité désirée (par exemple la plus fine) et ensuite de ne composer les relations qu'à cette granularité. Par exemple, soit g une granularité telle que $T \preceq g$ et A, B, C des entités temporelles telles que $A (<)^T B$, $B (>)^T C$, $A (<>)^T C$, et $A (=)^g B$, $B (<>)^g C$, $A (<>)^g C$. Par conversion descendante et composition, on ne déduit rien. Cependant, avec une conversion ascendante, on trouve $B (>)^g C$; par composition, on a $A (>)^g C$; et par conversion descendante, on déduit $A (>)^T C$.

De plus, combiner les relations entre intervalles avec les relations entre points ou les relations entre un point et un intervalle à une granularité plus grossière permet de déduire des relations plus précises. En effet, grâce à cette combinaison, on peut alors prendre en compte le fait que certaines bornes d'entités temporelles deviennent indiscernables (*i.e.* égales à la granularité grossière) alors que d'autres non, et en déduire l'ordre de ces bornes. Par exemple, si a et b sont indiscernables et que c n'est pas indiscernable de a ou b , alors c n'est pas entre a et b à une granularité plus fine. Par conséquent, raisonner avec toutes les granularités en même temps fournit plus d'informations et permet de détecter plus d'incohérences.

5 Cohérence

Dans cette section, nous nous intéressons à la vérification de la cohérence de *réseaux de contraintes* d'entités temporelles liées par des relations temporelles granulaires dans le contexte des granularités qualitatives. Plus précisément, un réseau de contraintes est cohérent

Opérateur	Règle
\circ	$\exists C \in \mathcal{E} : A R^g C \wedge C S^g B \Leftrightarrow A (R \circ S)^g B$
\uparrow	$g \preceq h \wedge A R^g B \Rightarrow A (\uparrow R)^h B$
\downarrow	$g \sqsubseteq h \wedge A R^h B \Rightarrow A (\downarrow R)^g B$
\cap	$A R^g B \wedge A S^g B \Leftrightarrow A (R \cap S)^g B$
$\bar{\cdot}$	$A R^g B \Leftrightarrow B \bar{R}^g A$

TABLE 3 – Règles pour déduire de nouvelles relations ; A, B sont des entités temporelles, g, h sont des granularités, et $R, S \subseteq \mathcal{R}$ sont des relations générales.

si on peut trouver une instanciation de ses variables qui satisfait toutes les contraintes. Dans le contexte des algèbres temporelles, ce problème apparaît, par exemple, dans la planification avec des contraintes temporelles [Allen, 1991] ou dans la reconnaissance de plans [Song, 1994].

Nous commençons par définir ce qu'est un réseau de contraintes dans notre contexte, et nous donnons un algorithme polynomial qui calcule la clôture algébrique d'un tel réseau, pouvant être utilisée pour vérifier sa cohérence. Dans ce qui suit, nous supposons que $T \preceq g$ pour toute granularité g ; c'est-à-dire que nous ne nous intéressons pas aux granularités avec des trous (voir la proposition 8), ainsi toutes les granularités g, h vérifient $g \preceq h \iff g \sqsubseteq h$.

5.1 Réseau de contraintes, clôture algébrique

Un réseau de contraintes dans le cadre de notre approche consiste en un ensemble de variables, représentant des entités temporelles $\mathcal{E} = \{E_1, \dots, E_n\}$ et des granularités $\mathcal{G} = \{g_1, \dots, g_m\}$, et en un ensemble de contraintes entre ces variables, de la forme « $E_i R^{g_k} E_j$ » (avec $R \subseteq \mathcal{R}$), et « $g_i \preceq g_j$ ». Si la relation entre deux granularités est inconnue, elle peut être omise ; mais chaque granularité est nécessairement plus grossière (et donc moins fine) que le domaine temporel. Ainsi les granularités forment un treillis. Un réseau de contraintes est un *scénario* s'il y a une relation élémentaire comme contrainte entre chaque couple de variables, à chaque granularité. Une *solution* d'un réseau de contraintes est une instanciation de toutes les variables qui satisfait toutes les contraintes, c'est-à-dire une affectation de granularités aux g_i telle que toutes les relations \preceq soient satisfaites, et une affectation d'entités aux E_i telle que toutes les relations granulaires entre elles soient satisfaites. Un réseau de contraintes est dit *cohérent* s'il possède au moins une solution.

Notons que ces réseaux de contraintes utilisent des *granularités qualitatives* : aucune information quantitative n'est utilisée — il n'est donc pas possible de vérifier la cohérence de réseaux de contraintes pour un

ensemble spécifique de granularités, tel que le calendrier grégorien, bien qu'une incohérence dans le cadre des granularités qualitatives implique une incohérence dans un cadre où les granularités seraient instanciées. Les granularités qualitatives sont, en particulier, destinées à être utilisées lorsque les granularités mises en jeu ne sont pas complètement connues (par exemple, dans le cadre du langage naturel) ou que cette connaissance est imprécise, et lorsqu'il est nécessaire de relaxer ou d'approximer le cadre quantitatif.

Sans granularités, la clôture algébrique d'un réseau de contraintes est un réseau de contraintes qui a exactement le même ensemble de solutions et qui satisfait la propriété suivante : $\forall A, B, C \in \mathcal{E}, R_{AC} \subseteq R_{AB} \circ R_{BC}$, où R_{XY} est l'ensemble des relations élémentaires entre les entités X et Y . Nous généralisons cette notion en disant qu'un réseau de contraintes granulaires est *algébriquement clos* s'il satisfait $\forall A, B, C \in \mathcal{E}, \forall g \in \mathcal{G}, R_{AC}^g \subseteq R_{AB}^g \circ R_{BC}^g$, et si pour tout $A, B \in \mathcal{E}$ et pour tout $g, h \in \mathcal{G}$ tel que $g \preceq h$, $R_{AB}^h \subseteq \uparrow R_{AB}^g$ et $R_{AB}^g \subseteq \downarrow R_{AB}^h$. Autrement dit, appliquer n'importe quel opérateur ne fournit aucune information supplémentaire.

5.2 Cohérence et clôture algébrique

Vérifier la cohérence d'un réseau de contraintes est un problème NP-complet pour l'algèbre des intervalles et l'algèbre des points et des intervalles [Krokhin et Jonsson, 2002]. Dans ce contexte, une approche classique pour vérifier la cohérence est d'explorer tous les scénarios d'un réseau de contraintes, en élaguant de nombreux cas incohérents grâce à la clôture algébrique, puisqu'un scénario est cohérent si et seulement si sa clôture algébrique ne contient pas d'ensemble vide comme relation entre deux entités. Cette équivalence est aussi vraie pour certains réseaux de contraintes qui ne sont pas des scénarios, par exemple, ceux associés à l'algèbre des points ou à la sous-classe ORD-Horn de l'algèbre des intervalles [Nebel et Bürckert, 1995]. En général, la clôture algébrique permet de détecter certaines incohérences, mais pas toutes.

De plus, pour certaines algèbres, calculer la clôture

algébrique ne suffit pas toujours à vérifier la cohérence d'un scénario [Renz et Ligozat, 2005].

Ce n'est heureusement pas le cas avec notre algèbre : bien que le problème de vérification de la cohérence soit NP-complet — puisqu'avec une seule granularité, notre algèbre correspond à celle des points et intervalles — on peut malgré tout s'appuyer sur la proposition suivante.

Proposition 9. *Un scénario est cohérent si et seulement si sa clôture algébrique ne contient pas l'ensemble vide comme relation entre deux entités à une des granularités.*

Démonstration. L'idée principale est que les bornes des entités sont totalement ordonnées à chaque granularité, puisqu'il n'y a aucun ensemble vide dans la clôture algébrique. Nous commençons par exprimer les contraintes du scénario en utilisant les relations granulaires des points. Ensuite, nous utilisons l'algorithme suivant pour instancier les variables : premièrement, on instancie chacune des bornes des entités temporelles de sorte que les contraintes du domaine temporel soient vérifiées. Deuxièmement, pour chaque granularité, on construit ses granules comme des intervalles $[a, b[$ tels que toutes les bornes des entités qui sont égales à cette granularité, et uniquement elles, appartiennent au même granule, où a est la plus petite de ces bornes et b est la plus petite des bornes qui sont après à cette granularité. Puisque la table de conversion est respectée et puisque l'algorithme a construit un alignement correct, on peut prouver que les granularités ainsi générées respectent bien la relation \preceq . \square

Grâce à cette propriété, la clôture algébrique peut, en particulier, être utilisée dans une recherche vérifiant la cohérence d'un réseau de contraintes (exhibant un scénario cohérent) [Renz et Ligozat, 2005]. Nous pouvons, par exemple, utiliser la procédure de Ladkin et Reinefeld [1992], en remplaçant leur algorithme de cohérence de chemin par l'algorithme de clôture algébrique présenté la section suivante.

5.3 Algorithme de clôture algébrique

L'algorithme 1 calcule la clôture algébrique d'un réseau de contraintes granulaires, et peut détecter des incohérences. Il prend en entrée une liste L de tuples (R, A, B, g) (les contraintes), où A et B sont des entités, g est une granularité, et R est l'ensemble des relations élémentaires autorisées entre A et B à la granularité g . La connaissance courante sur la relation entre A et B à la granularité g est enregistrée dans la variable R_{AB}^g ; initialement, chacune de ces variables est initialisée à \mathcal{R} (l'ensemble de toutes les relations élémentaires), et à la fin, R_{AB}^g contient la relation entre

A et B à la granularité g de la clôture algébrique du réseau de contraintes initial.

L'algorithme affine la variable R_{AB}^g courante avec les contraintes dans L , et propage ces nouvelles connaissances en appelant les procédures **convertir** et **composer**. La première procédure convertit la relation courante vers chacune des granularités alignées avec la granularité de cette relation, tandis que la seconde compose la relation courante avec chacune des autres relations de la même granularité. Les contraintes déduites sont ensuite ajoutées à L . Notons que cet algorithme peut aussi être utilisé avec l'algèbre des relations granulaires entre points.

Algorithme 1 : Calcul de la clôture algébrique d'un réseau de contraintes temporelles granulaires.

tant que L n'est pas vide **faire**

$(R, A, B, g) \leftarrow \text{pop}(L)$
 $r \leftarrow R \cap R_{AB}^g$
si $r \neq R_{AB}^g$ **alors**
 si $r = \emptyset$ **alors**
 └ Il y a une incohérence
 $R_{AB}^g \leftarrow r$
 ajouter $(\overline{R_{AB}^g}, B, A, g)$ à L
 composer (A, B, g, L)
 convertir (A, B, g, L)

Procédure **convertir** (A, B, g, L)

pour chaque $h \in \mathcal{G}$ tel que $g \preceq h$ **faire**
 └ ajouter $(\uparrow R_{AB}^g, A, B, h)$ à L
pour chaque $h \in \mathcal{G}$ tel que $h \preceq g$ **faire**
 └ ajouter $(\downarrow R_{AB}^g, A, B, h)$ à L

Procédure **composer** (A, B, g, L)

pour chaque $C \in \mathcal{E} \setminus \{A, B\}$ **faire**
 └ ajouter $(R_{AB}^g \circ R_{BC}^g, A, C, g)$ à L

Montrons maintenant que l'algorithme 1 est polynomial. S'il y a n entités temporelles, le nombre total de relations à chaque granularité est $n(n-1)$. L'idée est qu'une relation R_{AB}^g ne peut être modifiée plus de 26 fois, puisque sa taille ne fait que décroître. Ainsi, les procédures **composer** et **convertir** ne seront appelées, dans le pire des cas, que $26 \cdot mn(n-1)$ fois. La première procédure effectue $n-2$ compositions, et la deuxième, au plus $m-1$ conversions. Par conséquent, le nombre total d'opérations est borné par $26 \cdot mn(n-1)(n-2+m-1)$. Ainsi, l'algorithme 1 est en $O(mn^2(m+n))$. Si m est fixé, alors l'algorithme est en $O(n^3)$, ce qui correspond à la complexité de l'algorithme de clôture algébrique pour les algèbres temporelles sans granularités.

5.4 Un exemple de raisonnement

Dans cette section, nous montrons comment déduire la relation entre A et C à partir de « A est chevauchée par B , mais elles sont indiscernables sous un point de vue plus grossier », et « B démarre C , et elles ne sont pas égales sous ce même point de vue ». Les contraintes correspondantes sont $g \preceq h$, $A (\bar{o})^g B$, $A (=)^h B$, $B (s)^g C$, et $B (\mathcal{R} \setminus \{e, =\})^h C$. Par conversion ascendante de $B (s)^g C$ et intersection, on trouve $B (\cdot s)^h C$. Par composition de $A (=)^h B$ et $B (\cdot s)^h C$, on déduit $A (\cdot s)^h C$. Enfin, par conversion descendante, on conclut que $A (mo \cdot s \cdot dd \cdot bb)^g C$. Ensuite, par composition entre $A (\bar{o})^g B$ et $B (s)^g C$, on déduit que $A (\bar{o} d f)^g C$. Finalement, par intersection, on trouve $A (d)^g C$. Supposons maintenant qu'on sache également que $A (f)^g C$; on déduirait $A (\emptyset)^g C$, et par conséquent on aurait prouvé que le réseau est incohérent.

6 Travaux connexes et discussion

Contrairement à la table de conversion des relations temporelles granulaires entre intervalles présentée par Euzenat [1995], nous n'avons pas besoin d'informations quantitatives, c'est-à-dire des durées des entités temporelles, pour savoir si on peut appliquer les opérateurs de conversion vers une granularité plus fine ou plus grossière : on peut toujours les utiliser. Puisque par changement de granularité, un intervalle peut devenir un point et inversement, dans l'extension granulaire de l'algèbre des intervalles, il n'est pas possible de raisonner avec des granularités très grossières. Notre approche a une plus grande expressivité, permettant d'avoir un domaine temporel dense et des granularités non nécessairement alignées. La table de conversion de notre formalisme permet une fusion de relations temporelles à des précisions différentes qui est plus flexible que la table de conversion originale d'Euzenat. Par exemple, la relation « d'égalité » combinée à la relation « avant » à une granularité plus fine est incohérente avec la table de conversion d'Euzenat, mais elle ne l'est pas avec notre table de conversion puisque l'égalité peut être une égalité entre points. La combinaison de relations entre points et intervalles et relations entre intervalles à différentes granularités réduit l'ambiguïté lors du raisonnement (voir section 4.3). Nous avons la possibilité de ne pas spécifier le type des entités temporelles (point ou intervalle) et de déduire ensuite ce type. De plus, nos opérateurs de conversion vérifient la liste des propriétés désirables que n'importe quel système d'opérateurs de conversion granulaire devrait vérifier selon Euzenat [1995].

Il existe d'autres approches sur le temps granulaire

avec des contraintes symboliques. Celle de Badaloni et Berati [1994] combine des contraintes numériques et symboliques, mais seules les contraintes numériques sont converties lors du changement de granularité. Les intervalles sont retirés du réseau de contraintes s'ils deviennent des points à une granularité plus grossière. De plus, le réseau peut devenir incohérent si les granularités associées sont très grossières.

Becher *et al.* [2000] offrent un formalisme dans lequel les granularités sont définies implicitement. En effet, les relations du formalisme décrivent, en particulier, que la granularité de la première entité est plus fine que (resp. plus grossière que, égale à) la granularité de la deuxième entité, sans préciser quelles sont ces granularités.

Bittner [2002] présente une approche basée sur le concept d'approximation. Il fournit une table de conversion, mais elle est moins précise et moins générale que celle introduite dans cet article. Plus précisément, à la granularité la plus grossière, le type des entités n'est pas défini, et à la granularité la plus fine, les entités ne sont que des intervalles. Une autre table de conversion est donnée pour les périodes non convexes. Cependant, ces résultats sont limités à deux granularités.

Pour terminer, Bettini *et al.* [2000] offrent une approche granulaire avec des contraintes numériques ; et des formalismes concernant le temps ancré avec des granularités, où les entités temporelles sont précisément localisées, ont également été proposés et étudiés [Franceschet et Montanari, 2001; Bettini *et al.*, 2000; Montanari, 1996].

7 Conclusion et perspectives

L'ajout des granularités aux algèbres temporelles permet de combiner des informations avec des précisions différentes, de modéliser des relations imprécises et de raisonner sur celles-ci. Grâce à notre généralisation de la table de conversion granulaire des relations d'intervalles d'Euzenat [1995], dans laquelle les relations entre points et intervalles sont permises, les conversions granulaires à une granularité plus fine ou plus grossière peuvent toujours être appliquées. En outre, notre algèbre est plus expressive que celle des relations granulaires d'intervalles, puisque de nouvelles relations intuitives sont modélisables, et les déductions sont plus flexibles et plus précises : les informations qualitatives des différentes granularités sont pleinement utilisées. Avec notre formalisme, nous pouvons vérifier la cohérence d'un ensemble de relations entre des points et des intervalles définies sur plusieurs échelles temporelles et utilisant un concept de proximité qualitative comme dans le langage naturel.

Par ailleurs, notre travail complète théoriquement celui d'Euzenat en exposant des conditions suffisantes pour utiliser les tables de conversion granulaire, en s'appuyant sur la définition des granularités temporelles de Bettini *et al.* [2000].

Ce travail ouvre plusieurs voies de recherche. Actuellement, nous cherchons une sous-classe de notre algèbre pour laquelle la vérification de la cohérence des réseaux de contraintes associés est polynomiale. Nous projetons, dans un futur proche, d'analyser si la clôture algébrique implique la cohérence de chemin quand le domaine temporel est dense. Nous avons également l'intention de généraliser ces idées au raisonnement spatial. De plus, dans un cadre plus général, nous prévoyons d'étudier les conditions sous lesquelles l'ajout des granularités qualitatives à une algèbre des relations préserve la complexité polynomiale de la vérification de la cohérence.

Références

- ALLEN, J. F. (1983). Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843.
- ALLEN, J. F. (1991). Planning as temporal reasoning. *KR*, 91:3–14.
- BADALONI, S. et BERATI, M. (1994). Dealing with time granularity in a temporal planning system. In *Temporal Logic*, pages 101–116. Springer.
- BECHER, G., CLÉRIN-DEBART, F. et ENJALBERT, P. (2000). A qualitative model for time granularity. *Computational intelligence*, 16(2):137–168.
- BETTINI, C., JAJODIA, S. G. et WANG, S. X. (2000). *Time Granularities in Databases, Data Mining and Temporal Reasoning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1st édition.
- BITTNER, T. (2002). Approximate qualitative temporal reasoning. *Annals of Mathematics and Artificial Intelligence*, 36(1–2):39–80.
- EUZENAT, J. (1995). An algebraic approach to granularity in time representation. In *Proc. 2nd IEEE international workshop on temporal representation and reasoning (TIME), Melbourne (FL US)*, pages 147–154, Regina (CA). University of Regina.
- EUZENAT, J. et MONTANARI, A. (2005). Time granularity. In FISHER, M., GABBAY, D. et VILA, L., éditeurs : *Handbook of temporal reasoning in artificial intelligence*, pages 59–118. Elsevier, Amsterdam (NL).
- FRANCESCHET, M. et MONTANARI, A. (2001). *Dividing and conquering the layered land*. Thèse de doctorat, Department of Mathematics and Computer Science, University of Udine, Italy.
- HIRSCH, R. (1996). Relation algebras of intervals. *Artif. Intell.*, 83:267–295.
- KROKHIN, A. et JONSSON, P. (2002). Extending the point algebra into the qualitative algebra. In *Temporal Representation and Reasoning, International Symposium on*, pages 28–28. IEEE Computer Society.
- LADKIN, P. B. et MADDUX, R. D. (1994). On binary constraint problems. *Journal of the ACM*, 41:435–469.
- LADKIN, P. B. et REINEFELD, A. (1992). Effective solution of qualitative interval constraint problems. *Artif. Intell.*, 57(1):105–124.
- MEIRI, I. (1996). Combining qualitative and quantitative constraints in temporal reasoning. *Artif. Intell.*, 87(1–2):343–385.
- MONTANARI, A. (1996). *Metric and layered temporal logic for time granularity*. ILLC.
- NEBEL, B. et BÜRCKERT, H.-J. (1995). Reasoning about temporal relations : a maximal tractable subclass of allen's interval algebra. *Journal of the ACM (JACM)*, 42(1):43–66.
- RENZ, J. et LIGOZAT, G. (2005). Weak composition for qualitative spatial and temporal reasoning. In *Principles and Practice of Constraint Programming-CP 2005*, pages 534–548. Springer.
- SCHOCKAERT, S., COCK, M. D. et KERRE, E. E. (2006). Imprecise temporal interval relations. In *In Proceedings of the 6th International Workshop on Fuzzy Logic and Applications , LNAI 3849*, pages 108–113. Springer.
- SONG, F. (1994). Extending temporal reasoning with hierarchical constraints. In *TIME*, pages 21–28.
- VILAIN, M., KAUTZ, H. et BEEK, P. (1986). Constraint propagation algorithms for temporal reasoning. In *Readings in Qualitative Reasoning about Physical Systems*, pages 377–382. Morgan Kaufmann.
- VILAIN, M. B. (1982). A system for reasoning about time. In *Proceedings of the National Conference on Artificial Intelligence. Pittsburgh, PA, August 18–20, 1982.*, pages 197–201.

Forçage d’extension en argumentation abstraite par optimisation booléenne

Sylvie Coste-Marquis Sébastien Konieczny Jean-Guy Mailly Pierre Marquis

CRIL, Université d’Artois - CNRS

Lens, France

{coste,konieczny,mailly,marquis}@cril.fr

Résumé

La dynamique des systèmes d’argumentation constitue un sujet de recherche très développé actuellement. En particulier, le fait de modifier un système d’argumentation de façon à ce qu’un ensemble d’arguments E donné devienne une extension (ou soit inclus dans une extension), appelé problème du forçage (*enforcement*) de E , a reçu une certaine attention ces dernières années. Dans cet article, nous définissons une nouvelle famille d’opérateurs de forçage pour lesquels le forçage peut être réalisé en ajoutant de nouveaux arguments et de nouvelles attaques au système initial (comme c’est le cas dans les approches existantes), mais aussi en remettant en question les attaques (et non-attaques) de ce système. Cette famille d’opérateurs de forçage inclut les opérateurs existants, mais aussi de nouveaux opérateurs, pour lesquels le succès de l’opération de forçage est garanti. Nous montrons également comment le problème du forçage peut être modélisé et résolu via une traduction en problème d’optimisation booléenne. Une étude expérimentale conclut l’article et montre que notre approche se révèle efficace en pratique.

1 Introduction

Le travail de Dung sur l’argumentation abstraite [10] est à l’origine d’un cadre simple et puissant pour représenter et raisonner sur des arguments. Dans ce cadre, les arguments sont associés aux nœuds d’un graphe dirigé, appelé système d’argumentation (AF), et les arcs entre les nœuds encodent les attaques entre arguments. Plusieurs sémantiques d’acceptabilité ont été définies dans le but de déterminer des ensembles d’arguments qui peuvent être acceptés conjointement.

Les systèmes d’argumentation sont utiles pour modéliser et résoudre de nombreux problèmes, par exemple pour représenter et raisonner au sujet de dé-

bats dans un système multiagent. Quel que soit le problème considéré, la dynamique des systèmes d’argumentation, c’est-à-dire l’évolution d’un AF pour intégrer une nouvelle information, est une question importante. De ce fait, ce sujet a été largement étudié au cours des dernières années [7, 4, 6, 5, 8, 9].

Dans cet article, nous nous intéressons en particulier au problème du forçage : la question centrale est de déterminer s’il est possible de changer un AF pour assurer qu’un ensemble d’arguments particulier soit une extension, ou soit au moins inclus dans une extension. Ce problème a été étudié dans [2, 1]. Dans ces travaux, diverses contraintes concernent les changements qui sont autorisés : forcer un ensemble d’arguments peut être effectué via l’ajout de nouveaux arguments et de nouvelles attaques entre ces nouveaux arguments et les arguments du système. Cependant, aucun changement parmi les attaques du système de départ n’est permis.

De tels opérateurs de forçage peuvent être utiles pour modéliser un débat argumentatif entre plusieurs agents, car l’arrivée de nouveaux arguments dans le débat peut remettre en question les extensions existantes, et un agent peut alors souhaiter déterminer quels arguments et attaques doivent être ajoutés pour forcer l’ensemble d’arguments qu’il préfère. Cependant, dans de nombreux autres scénarios, il est impossible aux agents d’utiliser de nouveaux arguments pour justifier le changement, et il est alors utile de remettre en question les attaques entre les arguments existants [8, 9].

Dans cet article, nous définissons une nouvelle famille d’opérateurs de forçage, pour lesquels le forçage peut être réalisé en ajoutant de nouveaux arguments (et attaques) au système F donné (comme dans les approches précédentes), mais aussi en remet-

tant en question certaines attaques et (non-attaques) de F . Cette famille inclut les opérateurs de forçage précédents comme cas particuliers, mais aussi de nouveaux opérateurs pour lesquels le succès de l’opération de forçage est garanti. Nous montrons comment le problème de forçage pour les opérateurs de cette famille peut être modélisé comme un problème d’optimisation pseudo-booléenne. D’intenses expérimentations montrent que la méthode est applicable en pratique, et passe à l’échelle.

L’article est organisé de la façon suivante. Les notions de base d’argumentation abstraite sont rappelées en section 2. La section 3 présente les définitions des principales familles de forçage définies dans les travaux précédents par Baumann et Brewka, met en évidence certaines de leurs limites du point de vue de l’impossibilité, et ensuite définit une nouvelle famille d’opérateurs de forçage pour lesquels le succès est garanti. En section 4, nous expliquons comment réaliser le forçage en résolvant des problèmes d’optimisation. Avant la conclusion, la section 5 présente la méthode utilisée pour implémenter les opérateurs de forçage, et donne des résultats expérimentaux.

2 Préliminaires formels

Les définitions suivantes viennent de [10].

Définition 1. *Un système d’argumentation abstrait (AF) F est un graphe dirigé $\langle A, R \rangle$ où A est un ensemble d’entités atomiques appelées arguments et $R \subseteq A \times A$ est la relation d’attaque.*

La signification intuitive de la relation d’attaque est que $(a_i, a_j) \in R$ si, lorsque a_i est accepté par l’agent, alors a_j doit être rejeté. On dit qu’un ensemble d’arguments $E \subseteq A$ attaque un argument a_i si et seulement si $\exists a_j \in E$ tel que $(a_j, a_i) \in R$. Un argument a_i (respectivement un ensemble d’arguments E) défend l’argument a_j contre a_k tel que $(a_k, a_j) \in R$ si a_i (respectivement E) attaque a_k .

Dung a défini plusieurs *sémantiques d’acceptabilité*, qui visent à définir des *extensions* : une extension est un ensemble d’arguments qui peuvent être acceptés conjointement par l’agent. Quelle que soit la sémantique σ , $Ext_\sigma(F)$ est l’ensemble des σ -extensions du système F . Les différentes sémantiques reflètent certaines propriétés qui doivent être satisfaites par les extensions. Par exemple, $E \subseteq A$ est un ensemble sans conflit de $F = \langle A, R \rangle$ si et seulement s’il n’y a pas d’arguments $a_i, a_j \in E$ tels que $(a_i, a_j) \in R$. Ensuite, $E \subseteq A$ est une extension complète de $F = \langle A, R \rangle$ si et seulement si E est sans conflit et E contient chaque argument $a_k \in A$ qui est défendu par E . L’extension

de base est l’élément minimal (selon \subseteq) parmi les extensions complètes. $E \subseteq A$ est une extension stable de $F = \langle A, R \rangle$ si et seulement si E est sans conflit et E attaque tout argument $a_k \in A \setminus E$.

3 Forçage d’extension

Forcer un ensemble d’arguments E est défini dans [2] comme un changement d’un AF F vers un autre F' de façon à ce que E soit une extension de F' ou soit au moins inclus dans une extension de F' . Plusieurs méthodes de forçages sont présentées, basées sur la notion d’*expansion* d’un AF. Une expansion est l’ajout de nouveaux arguments et nouvelles attaques à un AF, respectant diverses contraintes. Le forçage de E dans F est alors défini comme une expansion de F telle que E en est une extension. Trois sortes d’expansions sont considérées :

- L’expansion *normale* : de nouveaux arguments sont ajoutés, ainsi que de nouvelles attaques telles qu’au moins un des arguments considérés par chaque nouvelle attaque est un des nouveaux arguments (il n’y a pas de changement dans les attaques entre les arguments déjà présents).
- L’expansion *faible* est une expansion normale telle qu’aucune attaque n’est dirigée d’un nouvel argument vers un ancien argument. Les nouveaux arguments sont alors appelés *arguments faibles*.
- L’expansion *forte* est une expansion normale telle qu’aucune attaque n’est dirigée d’un ancien argument vers un nouveau. Les nouveaux arguments sont alors appelés *arguments forts*.

En plus de la nature de l’expansion, deux paramètres additionnels interviennent dans la définition d’un opérateur de forçage. Tout d’abord, le forçage peut être *strict* lorsque l’on attend que l’ensemble d’arguments E soit exactement une extension du système résultant du forçage, ou *non strict* lorsqu’il suffit que l’ensemble E soit inclus dans une extension du système. De plus, le forçage peut être *conservatif* si la sémantique reste la même ou *libéral* dans le cas contraire. Dans la suite, nous définissons des opérateurs de forçage qui peuvent être utilisés aussi bien pour le forçage conservatif que libéral, puisque la sémantique associée au système d’entrée n’est pas spécifiée dans la définition des opérateurs. Cependant pour des raisons de simplicité, nous nous concentrons sur des cas de forçage conservatif.

Définition 2. *Soit $F = \langle A, R \rangle$ un AF, σ une sémantique d’acceptabilité, $E \subseteq A$ un ensemble d’arguments, et A' un ensemble d’arguments tel que $A \cap A' = \emptyset$. L’opérateur de forçage normal (respectivement normal strict) $+_\sigma^N$ (resp. $+_{\sigma,s}^N$) est défini comme une fonction*

qui associe à F et E un AF $F' = \langle A \cup A', R \cup R_{A'} \rangle$, avec $R_{A'}$ un ensemble d’attaques (a_i, a_j) telles que $a_i \in A'$ ou $a_j \in A'$, et tel que E est inclus dans (resp. est exactement) une extension de F' . De plus,

- si $R_{A'} \cap (A' \times A) = \emptyset$, alors $+_{\sigma}^{N,W}$ (resp. $+_{\sigma,s}^{N,W}$) est l’opérateur de forçage faible (resp. strict faible) ;
- si $R_{A'} \cap (A \times A') = \emptyset$, alors $+_{\sigma}^{N,S}$ (resp. $+_{\sigma,s}^{N,S}$) est l’opérateur de forçage fort (resp. strict fort).

Nous illustrons ci-dessous l’approche de forçage fort.

Exemple 1. Soit F le système donné à la figure 1(a). Son ensemble d’extensions stables est $Ext_{st}(F) = \{\{a_1, a_4\}\}$. L’extension que l’on souhaite forcer est $E = \{a_2, a_3\}$. Un forçage normal possible est présenté à la figure 1(b) : les extensions stables de F_1 sont $Ext_{st}(F_1) = \{\{a_2, a_3, b\}, \{a_1, a_4\}\}$, et donc l’ensemble E est forcé. F_2 , présenté à la figure 1(c), est un résultat possible de forçage fort, étant donné que $Ext_{st}(F_2) = \{\{a_2, a_3, b\}\}$. Le forçage faible n’est en

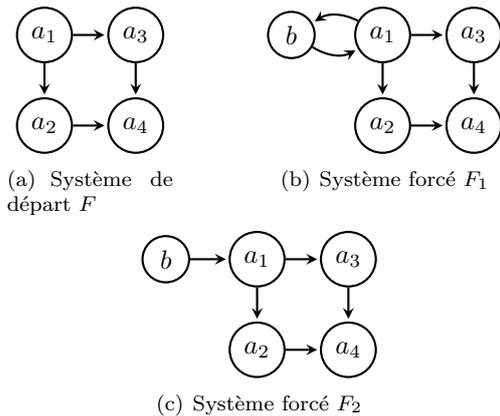


FIGURE 1 – Processus de forçage normal et fort

revanche pas possible dans ce cas, puisqu’il faut rejeter a_1 pour obtenir a_2 et a_3 dans une extension, ce qui n’est pas réalisable en ajoutant des arguments faibles.

Comme illustré par l’exemple précédent dans le cas du forçage faible, il est important de noter que quel que soit l’opérateur de forçage considéré, l’opération peut échouer. Par exemple, considérons l’exemple simple du forçage de $E = \{a_1, a_2\}$ dans le système $F = \langle A, R \rangle$ tel que $(a_1, a_2) \in R$. Il est évident que le forçage de E n’est possible avec aucun des opérateurs de forçage décrits précédemment. Le théorème 2 et le théorème 3 de [2] présentent divers résultats d’impossibilité au sujet des opérateurs de forçage strict. Il est toutefois intéressant de noter le résultat qui indique que pour tout système F , il est possible de forcer tout ensemble d’arguments E qui est sans conflit dans F , via un opérateur de forçage fort non strict, et qui garantit aussi

que l’ajout d’un unique argument est suffisant [2]. Cela signifie que le forçage fort non strict peut être réalisé en ajoutant seulement un seul argument, ce qui est particulièrement utile pour implémenter les encodages correspondants (voir section 4).

Notons que la présence de conflits dans l’ensemble E est une condition suffisante, mais pas nécessaire, pour provoquer l’échec du forçage. Dans le but de rendre ceci plus formel, introduisons la notion d’ensemble d’arguments non trivial pour une sémantique donnée :

Définition 3. Soit $F = \langle A, R \rangle$ un système d’argumentation, et σ une sémantique. $E \subseteq A$ est un ensemble d’arguments non trivial pour σ dans F si et seulement si E est sans conflit dans F et $E \notin Ext_{\sigma}(F)$.

Nous supposons que l’ensemble E à forcer est non trivial pour σ , afin d’éviter les cas évidents pour lesquels le forçage est d’emblée satisfait car E est une σ -extension de F , ou impossible en raison des conflits. Cependant, cette hypothèse n’est pas suffisante pour éviter l’échec du forçage pour toute sémantique :

Proposition 1. Pour tout $F = \langle A, R \rangle$ et $E \subseteq A$ un ensemble d’arguments non trivial pour la sémantique stable dans F , il n’y a aucun forçage strict de E dans F pour la sémantique stable.

Proposition 2. Pour tout $F = \langle A, R \rangle$, et $E \subseteq A$ un ensemble d’arguments non trivial pour la sémantique complète dans F ,

1. si E ne se défend pas contre tout attaquant, alors il n’y a aucun forçage strict de E dans F pour la sémantique stable ;
2. sinon, si E défend un argument $a_i \in A \setminus E$, alors
 - (a) il n’y a pas de forçage strict faible de E dans F pour la sémantique complète.
 - (b) si les cycles de longueur impaire ne sont pas autorisés, alors il n’y a aucun forçage strict fort de E dans F pour la sémantique complète.

Proposition 3. Pour tout $F = \langle A, R \rangle$ et $E \subseteq A$ un ensemble d’arguments non trivial pour la sémantique de base dans F , si $Ext_{gr}(F) = \{\emptyset\}$, alors il n’y a pas de forçage strict de E dans F pour la sémantique de base.

Forçage à arguments fixés. Dans les approches précédentes pour le forçage d’un ensemble d’arguments, il est supposé que de nouveaux arguments peuvent être ajoutés, et que les interactions entre les arguments pré-existants ne changent pas. Cette méthode est particulièrement sensée lorsque le forçage est supposé être

le résultat d’un dialogue : étant donné un système d’argumentation qui représente l’état du dialogue, un agent ajoute de nouveaux arguments lorsqu’il veut convaincre un autre agent d’accepter un ensemble d’arguments particulier. Le fait d’interdire tout changement entre les attaques du système de départ est la raison des résultats d’impossibilité donnés ci-dessus. Il est intéressant de noter que le cas « contraire », c’est-à-dire si l’on considère des situations pour lesquelles on ne peut changer l’ensemble d’arguments, mais il est possible de faire évoluer la relation d’attaque, a aussi du sens. Par exemple, quand l’agent constate qu’un ensemble d’arguments est une extension dans le résultat du processus argumentation, mais ne correspond pas au résultat du système personnel de l’agent. Dans de tels cas, sans connaissance de nouveaux arguments, l’agent doit changer ses croyances au sujet de la relation d’attaque pour être cohérent avec l’ensemble d’arguments observé.

Définition 4. Soit $F = \langle A, R \rangle$ un AF, σ une sémantique d’acceptabilité, et $E \subseteq A$ un ensemble d’arguments. L’opérateur de forçage à arguments fixés (resp. strict à arguments fixés) $+_{\sigma}^A$ (resp. $+_{\sigma,s}^A$) est défini comme une fonction qui associe à F et E à un AF $F' = \langle A, R' \rangle$, avec $R' \subseteq A \times A$, et tel que E est inclus dans (resp. est exactement) une extension de F' .

L’opérateur de forçage à arguments fixés garantit le succès de l’opération, même dans le cas strict :

Proposition 4. Soit $F = \langle A, R \rangle$ un AF, σ une sémantique d’acceptabilité et $E \subseteq A$ un ensemble d’arguments. Il existe un forçage strict F' de E dans F .

Bien sûr, les deux idées (ajouter des arguments et changer les attaques) peuvent être combinées :

Définition 5. Soit $F = \langle A, R \rangle$ un AF, σ une sémantique d’acceptabilité, et $E \subseteq A$ un ensemble d’arguments. L’opérateur de forçage général (resp. strict général) $+_{\sigma}$ (resp. $+_{\sigma,s}$) est défini comme une fonction qui associe à F et E un AF $F' = \langle A \cup A', R' \rangle$, avec $R' \subseteq A \times A'$, et tel que E est inclus dans (resp. est exactement) une extension de F' .

Exemple 2. Considérons à nouveau l’AF F décrit à la figure 1(a). Nous avons donné un exemple de forçage non strict fort, mais comme montré par la proposition 1, il est impossible de réaliser un forçage strict sous la sémantique stable en utilisant les approches de Baumann (normal, fort et faible). Utilisons à présent le forçage à arguments fixés pour obtenir un forçage strict de l’ensemble d’arguments $E = \{\{a_2, a_3\}\}$. Un résultat possible est le système F_3 décrit à la figure 2(a), dont les extensions stables sont $Ext_{st}(F_3) = \{\{a_1, a_4\}, \{a_2, a_3\}\}$, et donc E est

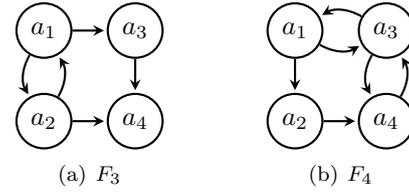


FIGURE 2 – Deux résultats possibles du forçage à arguments fixés

forcé en tant qu’extension stable du résultat. Un autre résultat possible est F_4 donné à la figure 2(b), dont les extensions stables sont les mêmes : $Ext_{st}(F_4) = \{\{a_1, a_4\}, \{a_2, a_3\}\}$.

Dans l’exemple précédent, les deux options présentées pour le forçage, F_3 et F_4 , ne sont pas équivalentes du point de vue du nombre d’attaques qu’il a été nécessaire d’ajouter pour parvenir au résultat. En effet, par rapport au système de départ F , F_3 ajoute une unique attaque (a_2, a_1) , tandis que F_4 ajoute deux attaques (a_3, a_1) et (a_4, a_3) . Cela nous conduit à nous poser la question du changement minimal dans le processus de forçage.

Changement minimal. Comme expliqué à la proposition 4, la possibilité de forcer un ensemble d’arguments est assurée lorsque des changements de la relation d’attaque sont autorisés. D’un point de vue pratique, cela offre une garantie de succès, ce qui est une propriété importante pour un tel opérateur. Une autre propriété attendue est le changement minimal, emprunté à la théorie des changements de croyances. Les processus de forçage peuvent conduire à plusieurs résultats, et donc les opérateurs de forçage définis précédemment doivent sélectionner un des systèmes possibles comme résultat. [1] a déjà étudié une telle notion de proximité pour les approches de forçage basées sur l’expansion normale. Il définit le changement minimal comme la minimisation du nombre d’attaques qui sont ajoutées au système durant le processus de forçage. Nous généralisons cette notion de changement minimal, en utilisant la distance Hamming pour mesurer à quel point deux systèmes sont différents.

Définition 6. La distance de Hamming d_h entre deux systèmes d’argumentation $F = \langle A, R \rangle$ et $F' = \langle A', R' \rangle$ est définie par :

$$d_h(F, F') = |(R \setminus R') \cup (R' \setminus R)|$$

Pour tout opérateur de forçage $+$, l’opérateur minimisant associé $+_{\min}$ est tel que le système résultant F' minimise la distance de Hamming avec le système de départ F parmi l’ensemble des systèmes qui sont un résultat possible de $+$.

4 Le forçage via la satisfaisabilité et l’optimisation

Une première observation au sujet du calcul du forçage d’un ensemble d’arguments est que lorsque l’on limite le nombre de changements autorisés dans la relation d’attaque, le forçage est difficile d’un point de vue calculatoire dans le cas général :

Proposition 5. *Soit $F = \langle A, R \rangle$ un AF, $E \subseteq A$, et k un entier. Déterminer s’il est possible de forcer E dans F pour la sémantique stable avec au plus k changements (ajouts ou retraits) d’attaques est NP-difficile.*

La proposition 5 assure que (sauf si $P = NP$) il n’y a pas d’algorithme en temps polynomial qui réalise le forçage avec changement minimal dans le cas général. Pour cette raison, il est tout à fait sensé de traiter les problème de forçage (resp. de forçage avec changement minimal) en utilisant des algorithmes développés pour résoudre (resp. optimiser) des problèmes NP-difficiles. C’est pourquoi nous procédons de la façon suivante : nous réduisons l’enforcement à un problème de satisfaisabilité propositionnelle, et le forçage avec changement minimal à un problème d’optimisation pseudo-booléenne.

Le forçage via la satisfaisabilité booléenne. Notre approche par traduction est basée sur la possibilité d’associer à un AF F et une sémantique σ une formule propositionnelle dont les modèles correspondent exactement aux σ -extensions de F .

Définition 7. *Étant donné F un AF et σ une sémantique, Φ_σ^F est une formule propositionnelle construite sur l’ensemble des variables booléennes $\{x_a \mid a \in A\}$, telle que $\{x_{a_1}, \dots, x_{a_k}\}$ est un modèle de Φ_σ^F si et seulement si $\{a_1, \dots, a_k\}$ est une σ -extension de F .*

Dans la suite, dans un souci de simplification, et comme aucune ambiguïté n’est possible, nous écrivons les formules en utilisant le symbole a_i au lieu de x_{a_i} . Nous nous focalisons sur l’encodage Φ_{st} de la sémantique stable, tel que présenté dans [3]. Étant donné $F = \langle A, R \rangle$, Φ_{st} est défini par

$$\Phi_{st} = \bigwedge_{a_i \in A} [a_i \Leftrightarrow (\bigwedge_{a_j : (a_j, a_i) \in R} \neg a_j)]$$

Ensuite, vérifier si un ensemble d’arguments E est une extension stable de F est équivalent à vérifier si la formule $\Phi_{st,s}^E = \Phi_{st} \wedge (\bigwedge_{a_k \in E} a_k) \wedge (\bigwedge_{a_l \notin E} \neg a_l)$ est cohérente. Pour réaliser un forçage non strict, il est nécessaire d’avoir un moyen de s’assurer que E est inclus dans une extension. Il suffit pour cela de retirer la conjonction $(\bigwedge_{a_l \notin E} \neg a_l)$ de la formule, $\Phi_{st,s}^E$, ce qui

donne précisément la formule Φ_{st}^E dont nous avons besoin.

Dans le but d’établir un lien entre la sémantique et la structure du système d’argumentation dans les modèles de la formule, nous introduisons les variables booléennes att_{a_i, a_j} qui signifient qu’il y a une attaque de a_i vers a_j dans le système F considéré. Les formules données précédemment peuvent être généralisées en :

$$\Phi_{st}^{A,E} = \bigwedge_{a_i \in A} [a_i \Leftrightarrow (\bigwedge_{a_j \in A} (att_{a_j, a_i} \Rightarrow \neg a_j))] \wedge (\bigwedge_{a_k \in E} a_k)$$

et

$$\Phi_{st,s}^{A,E} = \bigwedge_{a_i \in A} [a_i \Leftrightarrow (\bigwedge_{a_j \in A} (att_{a_j, a_i} \Rightarrow \neg a_j))] \wedge (\bigwedge_{a_k \in E} a_k) \wedge (\bigwedge_{a_l \notin E} \neg a_l)$$

Assez clairement, propager les valeurs de vérité des variables att_{a_i, a_j} dans ces formules est suffisant pour obtenir la formule $\Phi_{st,s}^E$ et sa contrepartie non stricte. Cette formule est la base de nos encodages propositionnels des opérateurs de forçage d’extension. Il nous reste à introduire deux fonctions qui permettent de « décoder » de telles formules pour obtenir des systèmes d’argumentation :

- $Proj_{att}^A(\Phi) = \{m \cap \{att_{a_i, a_j} \mid a_i, a_j \in A\} \mid m \models \Phi\}$ est l’ensemble des modèles de la formule Φ projetés sur les variables att_{a_i, a_j} .
- $arg^A(m) = \langle A, R \rangle$ tel que $(a_i, a_j) \in R$ si et seulement si $att_{a_i, a_j} \in m$, avec m un modèle projeté sur les variables att_{a_i, a_j} , est le système correspondant à l’affectation des variables att_{a_i, a_j} . Ensuite, avec M un ensemble de tels modèles, $arg^A(M) = \{arg^A(m) \mid m \in M\}$.

Nous avons aussi besoin d’un encodage pour la structure d’un AF $F = \langle A, R \rangle$:

$$struct_{A'}(F) = (\bigwedge_{(a_i, a_j) \in R} att_{a_i, a_j}) \wedge (\bigwedge_{(a_i, a_j) \notin R} \neg att_{a_i, a_j})$$

où $a_i, a_j \in A \cup A'$. $struct(F)$ est une notation pour $struct_\emptyset(F)$.

Finalement, $\delta : \{F_1, \dots, F_k\} \mapsto F_j$ tel que $F_j \in \{F_1, \dots, F_k\}$ est une règle de *tie-break* qui sélectionne un unique système parmi un ensemble de systèmes.

Sur cette base, tout opérateur de forçage défini dans la section précédente peut être encodé comme un problème de satisfaisabilité d’une formule propositionnelle. En effet, par construction, chaque modèle de la formule $\Phi_\sigma^{A \cup A', E}$, lorsqu’il est projeté sur les variables att_{a_i, a_j} donne un AF qui est un forçage normal de E . Il nous suffit d’ajouter les bonnes contraintes pour assurer que le forçage est fort ou faible. Dans le but d’éviter l’introduction de nouveaux arguments, et obtenir les

opérateurs à arguments fixés, il nous suffit de considérer la formule $\Phi_{\sigma}^{A,E}$ dans l’encodage de l’opérateur. De façon similaire, les formules $\Phi_{\sigma,s}^{A \cup A',E}$ et $\Phi_{\sigma,s}^{A,E}$ peuvent être utilisées pour définir les contreparties strictes des opérateurs de forçage.

Définition 8. *Pour tout AF $F = \langle A, R \rangle$, tout ensemble d’arguments $E \subseteq A$, toute sémantique σ , et $X = \sigma$ ou $X = \sigma, s$:*

$$F +_X^N E = \delta(\arg^{A \cup A'}(Proj_{att}^{A \cup A'}(\Phi_X^{A \cup A',E} \wedge struct(F))))$$

$$F +_X^{N,W} E = \delta(\arg^{A \cup A'}(Proj_{att}^{A \cup A'}(\Phi_X^{A \cup A',E} \wedge struct(F) \wedge (\bigwedge_{(a_i, a_j) \in A' \times A} \neg att_{a_i, a_j}))))$$

$$F +_X^{N,S} E = \delta(\arg^{A \cup A'}(Proj_{att}^{A \cup A'}(\Phi_X^{A \cup A',E} \wedge struct(F) \wedge (\bigwedge_{(a_i, a_j) \in A \times A'} \neg att_{a_i, a_j}))))$$

$$F +_X^A E = \delta(\arg^A(Proj_{att}^A(\Phi_X^{A,E})))$$

$$F +_X E = \delta(\arg^{A \cup A'}(Proj_{att}^{A \cup A'}(\Phi_X^{A \cup A',E}))).$$

Pour chacun de ces opérateurs de forçage $+$, tout AF F et tout ensemble d’arguments E , $Enc(F + E)$ représente l’encodage propositionnel correspondant. Par exemple, $Enc(F +_X^N E)$ est la formule propositionnelle $\Phi_{\sigma}^{A \cup A',E} \wedge struct(F)$. Il suffit d’utiliser n’importe quel prouveur SAT pour trouver un modèle de $Enc(F + E)$ et ensuite décoder les valeurs de vérité des variables att_{a_i, a_j} pour déterminer un forçage de E dans F .

Le forçage avec changement minimal via l’optimisation pseudo-booléenne.

Comme expliqué précédemment, [1] considère un principe de changement minimal dans le processus de forçage. Dans ce travail, la minimalité fait référence à la minimalité du nombre d’attaques à ajouter au système pour réaliser une expansion normale. Un moyen possible d’assurer le changement minimal est de définir une règle de *tie-break* δ qui sélectionne un des systèmes d’argumentation résultant du forçage qui minimise cette mesure. Dans le but de tirer avantage de logiciels d’optimisation disponibles, une approche alternative consiste à encoder le critère de minimalité en une fonction objectif pseudo-booléenne :

$$newAtt(A \cup A') = \sum_{(a_i, a_j) \in ((A \cup A') \times (A \cup A')) \setminus (A \times A)} att_{a_i, a_j}$$

Pour le forçage fort ou faible, cette représentation de la fonction objectif peut être simplifiée, puisque les variables att_{a_i, a_j} qui correspondent à des attaques interdites ont nécessairement la valeur *faux*.

Le changement minimal pour le forçage à arguments fixés et le forçage général n’est pas aisé à encoder en utilisant les variables booléennes disponibles. Afin d’obtenir l’encodage voulu, nous considérons des variables additionnelles pour représenter l’état du système avant le forçage. On minimise ensuite le nombre

de différences entre les valeurs de vérité de ces variables et celles qui leur correspondent dans le nouveau système. Formellement, pour tout couple d’arguments $(a_i, a_j) \in (A \cup A') \times (A \cup A')$, la variable $prev_{a_i, a_j}$ a la valeur *vrai* si et seulement si $(a_i, a_j) \in R$. Ainsi, $prev_{a_i, a_j} \oplus att_{a_i, a_j}$, où \oplus désigne le connecteur « ou exclusif » habituel, donne l’information au sujet du changement de l’attaque (a_i, a_j) : s’il y avait précédemment une attaque de a_i à a_j , et que cette attaque n’est plus présente après le forçage, $prev_{a_i, a_j} \oplus att_{a_i, a_j}$ est *vrai*. C’est aussi *vrai* s’il n’y avait pas attaque avant le forçage, et qu’il y en a une après le forçage. L’encodage de la structure du système F doit aussi être revu pour tenir compte des variables $prev_{a_i, a_j}$:

$$struct_{A'}^{prev} = (struct_{A'}(F))|_{att_{a_i, a_j} \leftarrow prev_{a_i, a_j}}$$

Minimiser les différences sur la relation d’attaque est alors équivalent à minimiser la fonction objectif

$$attChange(A \cup A') = \sum_{a \in (A \cup A'), b \in (A \cup A')} prev_{a_i, a_j} \oplus att_{a_i, a_j}$$

Clairement, cette somme compte 1 pour toute attaque (a_i, a_j) dans le résultat qui concerne un argument de l’ensemble A' , puisque $prev_{a_i, a_j}$ est toujours *faux* si $a_i \in A'$ ou $a_j \in A'$. Ainsi, cette approche peut aussi être utilisée pour le forçage général.

Résumons à présent les définitions des opérateurs de forçage avec changement minimal :

Définition 9. *Pour tout AF $F = \langle A, R \rangle$, tout ensemble d’arguments $E \subseteq A$,*

- si $+_{\min}$ est un opérateur de forçage normal, fort ou faible (ou sa contrepartie stricte) avec changement minimal, alors forcer E dans F est équivalent à satisfaire $Enc(F + E)$ tout en minimisant $newAtt(A \cup A')$;
- si $+_{\min}$ est un opérateur de forçage à arguments fixés ou général (ou sa contrepartie stricte) avec changement minimal, alors forcer E dans F est équivalent à satisfaire $Enc(F + E) \wedge_{\min} struct_{A'}^{prev}(F)$ tout en minimisant $attChange(A \cup A')$.

Nous remarquons que la seconde méthode, qui fait intervenir la fonction objectif $attChange(A \cup A')$, est suffisante pour chaque opérateur de forçage. Cependant, comme cette méthode requiert l’ajout de variables booléennes $prev_{a_i, a_j}$, nous ne l’utilisons pas si ce n’est pas réellement nécessaire, pour éviter toute perte d’efficacité calculatoire.

Le cadre formel adapté à notre problème d’optimisation est le problème d’optimisation pseudo-booléenne (PB), qui est une extension du problème de satisfaisabilité booléenne.

Définition 10. *Étant donné un ensemble de variables booléennes $V = \{x_1, \dots, x_n\}$ et une fonction $\mathcal{O} : \{0, 1\}^n \rightarrow \mathbb{R}$, un problème PB-Opt $\mathcal{P} = (\mathcal{C} = \{c_1, \dots, c_m\}, \mathcal{O})$ sur V est la recherche d’une affectation de chaque variable dans V telle que les contraintes*

$$\begin{aligned} c_1 : & w_1^1 x_1 + \dots + w_n^1 x_n \geq k^1 \\ & \vdots \\ c_m : & w_1^m x_1 + \dots + w_n^m x_n \geq k^m \end{aligned}$$

sont satisfaites, et la fonction objectif \mathcal{O} atteint sa valeur optimale.

Dans notre cas, la valeur optimale de la fonction objectif est sa valeur minimale. Il est bien connu que toute formule propositionnelle peut être transformée en une formule équivalente sous forme normale conjonctive (CNF), et toute clause d’une formule CNF peut être ré-écrite comme une contrainte PB : la clause $x_1 \vee x_2 \vee \dots \vee x_n$ est satisfaite si et seulement si la contrainte PB $x_1 + x_2 + \dots + x_n \geq 1$ est satisfaite. Ainsi, les problèmes d’optimisation décrits précédemment peuvent être aisément traduits dans le formalisme PB.

5 Résultats expérimentaux

Dans notre étude expérimentale, nous nous sommes focalisés sur le problème de forçage avec changement minimal. Nous avons implémenté la famille d’opérateurs de forçage décrits dans cet article, en utilisant l’outil `Cplex` [13] comme moteur d’optimisation sous-jacent. Nous présentons uniquement les résultats obtenus pour trois approches : le forçage non strict fort de [2], et les versions stricte et non stricte de notre opérateur de forçage à arguments fixés. Dans chaque cas, nous avons utilisé la sémantique stable.

Nous étudions une classe de systèmes d’argumentation aléatoires [11, 12]. Étant donné un ensemble de n arguments, chaque attaque entre deux arguments est générée en utilisant une probabilité fixée p . Dans nos expérimentations, n varie de 200 jusque 500 arguments. Pour chaque n , les graphes sont divisés en quatre familles, correspondant à quatre valeurs de p distinctes. Nous avons utilisé les familles de systèmes de [11], où $p \in \{0.4, 0.65, 0.9\}$. Nous avons aussi généré des systèmes avec une probabilité $p = 0.1$. Il apparaît dans nos expérimentations que le choix de p ne change pas de façon significative les performances de l’algorithme de forçage par traduction, donc nous reportons uniquement les résultats pour $p = 0.1$.

Nous avons calculé le forçage avec changement minimal d’ensembles d’arguments E dans des systèmes d’argumentation F contenant n arguments, avec $n \in$

$\{200, 300, 400, 500\}$. Pour chaque système F avec n arguments, nous avons considéré des ensembles d’arguments E à forcer tels que $|E| = m$, m variant entre 1 et $\frac{35}{100}n$. Pour chaque couple de valeurs (n, m) , nous avons généré 10 requêtes de forçage¹. Sur la figure 3, l’ordonnée de chaque point des courbes correspond au temps de calcul moyen pour tous les couples (F, E) qui ont été considérés, où le nombre n d’arguments de F est indiqué en abscisse.

Le premier résultat intéressant qui ressort de nos expérimentations est que le forçage est traitable en pratique sur de tels systèmes d’argumentation générés aléatoirement, ce qui n’était pas évident, étant donné que le forçage est NP-difficile. Comme illustré à la figure 3, le temps de calcul augmente de façon raisonnable avec le nombre d’arguments n , jusqu’à une valeur moyenne de 13.76 secondes (pour un écart-type de 0.48) obtenue avec des systèmes d’argumentation à 500 arguments, quand le forçage strict à argument fixés est considéré (courbe +), et jusqu’à une moyenne de 16.61 secondes (pour un écart-type de 5.31) quand le forçage fort est considéré (courbe \times).

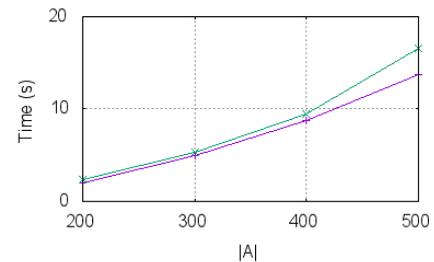
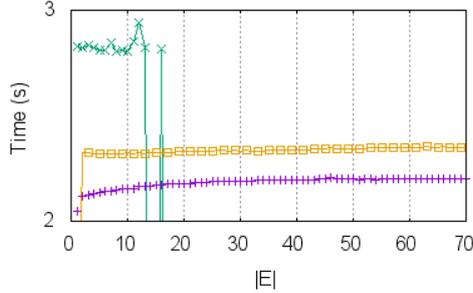


FIGURE 3 – Temps moyen pour le forçage fort (courbe \times) et strict à arguments fixés (courbe +), n variant de 200 à 500

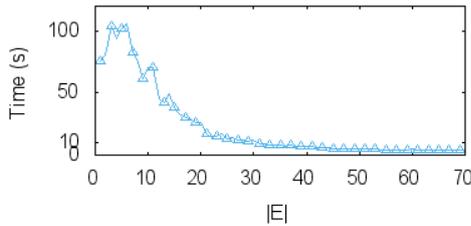
Ensuite, nous avons comparé les trois différentes approches sur des familles de systèmes d’argumentation à 200 arguments, laissant le cardinal de E varier de 1 à 70. Le but de cette comparaison est d’étudier l’impact du cardinal de E sur le comportement des opérateurs de forçage. Nous n’avons pas retiré les ensembles d’arguments triviaux des expérimentations, puisque notre approche peut retirer des attaques, rendant sans conflit un ensemble au départ conflictuel. Cela nous permet d’illustrer le taux d’échec du forçage fort, qui est (sans surprise) élevé, puisque le forçage est impossible dès que la requête de forçage est un ensemble conflictuel dans le système de départ. En effet, avec une probabilité p qu’une attaque apparaisse entre deux arguments, la probabilité qu’un ensemble d’arguments E de cardinal m soit sans conflit est $(1 - p)^m$. Donc,

1. Nous appelons « requête de forçage » l’ensemble d’arguments E dont on attend qu’il soit une extension.

plus le cardinal de la requête de forçage est élevé, plus la probabilité que le forçage soit possible est faible. En particulier, dans nos expérimentations, le forçage fort échoue toujours lorsque $m > 20$. Clairement, le taux d’échec du forçage fort croît exponentiellement en fonction de m .



(a) Forçage strict à arguments fixés (courbe +), et succès (courbe \times) et échecs (courbe \square) pour le forçage fort



(b) Forçage non strict à argument fixés

FIGURE 4 – Temps moyens, $n = 200$, m variant de 1 à 70

Nous avons comparé le temps de calcul pour le forçage pour les trois approches (voir la figure 4). La courbe \times représente le temps moyen pour réaliser le forçage fort, tandis que la courbe \square correspond au temps nécessaire à l’algorithme pour reporter l’échec quand le forçage est impossible. Pour le forçage fort aussi bien que pour le forçage à arguments fixés (courbe +), il apparaît que le temps requis pour obtenir le résultat est le même quel que soit le cardinal de la requête de forçage et la probabilité d’attaques dans le système : entre 2 et 3 secondes. Le cardinal a plus d’influence sur l’opérateur de forçage non strict à arguments fixés. En effet, plus la requête de forçage est petite, plus le résultat est difficile à obtenir. Quand le cardinal croît, le temps de calcul décroît jusqu’à quelques secondes.

Enfin, nous avons mesuré les efforts requis du point de vue du changement (c’est-à-dire le nombre d’attaques à ajouter ou retirer) pour forcer E dans le système d’argumentation (voir figure 5). Clairement, l’effort requis croît avec le cardinal de la requête de for-

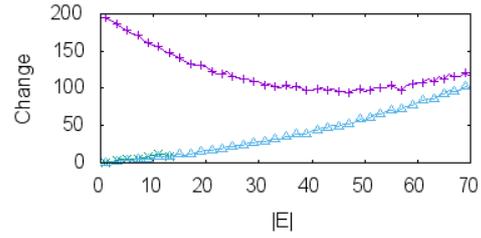


FIGURE 5 – Changement moyen pour le forçage fort (courbe \times), strict à arguments fixés (courbe +) et non strict à arguments fixés (courbe Δ)

çage pour l’opérateur fort (courbe \times) et non strict à arguments fixés (courbes Δ). La courbe pour le forçage fort s’arrête bien avant les autres (après 14 arguments) en raison du haut taux d’échec que nous avons mentionné. Jusqu’à ce point, comme on peut l’observer, la courbe \times est presque identique à la courbe Δ . Le forçage strict à arguments fixés (courbe +) requiert plus de changements de la relation d’attaque. Comme l’ensemble forcé doit être exactement une extension stable, même un petit ensemble d’arguments requiert l’addition d’un grand nombre d’attaques. Par exemple, avec $E = \{a_i\}$, il est nécessaire que a_i attaque chaque autre argument pour assurer que E est une extension stable. Quand le cardinal m de E croît, les efforts requis par les versions stricte et non stricte de l’opérateur à arguments fixés convergent.

6 Conclusion

Dans cet article, nous avons étudié le problème de forçage d’un ensemble d’arguments en tant qu’une extension d’un système d’argumentation. Notre contribution est multiple. Premièrement, nous avons indiqué plusieurs cas pour lesquels les approches de forçage existantes peuvent échouer, même si l’ensemble à forcer est sans conflit. Pour pallier cette faiblesse, nous avons défini de nouvelles méthodes de forçage, pour lesquelles le succès du processus est garanti. Pour chacune de ces méthodes, nous avons donné des encodages booléens qui permettent de tirer parti des logiciels de satisfaction et d’optimisation afin de procéder au forçage. Nous avons utilisé un outil d’optimisation connu pour implémenter une bibliothèque d’opérateurs de forçage, et nous avons expérimenté certains d’entre eux sur une large classe d’instances. Les expérimentations ont montré que notre approche est applicable en pratique et supporte le passage à l’échelle.

Il est intéressant de noter que d’autres types d’opérations de changement sont des forçages d’extension spécifiques. Par exemple, l’explication crédule étudiée dans [5] est le forçage d’un singleton. De façon simi-

laire, certains types de changements dirigés par les buts de [14] et certains opérateurs de révision de [8, 9] sont des opérateurs de forçage. Ainsi, notre approche de forçage est aussi utile pour calculer le résultat de tels opérateurs de changement dans les systèmes d’argumentation.

Ce travail ouvre plusieurs perspectives pour de futures recherches. Pour autant que nous le sachions, aucun des travaux existants au sujet du changement dans les systèmes d’argumentation n’a conduit à l’implémentation de logiciels (efficaces). Cependant, l’implémentation de systèmes logiciels pour l’argumentation est actuellement un sujet actif (dans la même veine, voir l’organisation d’une compétition de prouveurs d’argumentation [15]). La création de notre logiciel de forçage est issue de la même volonté de fournir des outils de raisonnement argumentatif, ce qui est aujourd’hui une étape nécessaire pour faire progresser ce domaine. Nous prévoyons ainsi d’encoder et d’implémenter des opérateurs de forçage pour d’autres sémantiques. Cette tâche est directe pour d’autres sémantiques pour lesquelles le calcul d’une extension est un problème (au pire) NP-complet. En effet, de telles sémantiques σ peuvent être encodées par des formules propositionnelles Φ_σ similaires à la formule présentée ici pour la sémantique stable. Les sémantiques dont la complexité est plus élevée, comme par exemple la sémantique préférée, nécessitent un travail plus approfondi, étant donné qu’il faut passer par un cadre formel différent (comme les formules booléennes quantifiées par exemple), ou accepter une croissance exponentielle de la taille des encodages. D’autres extensions de nos approches peuvent être envisagées, comme la prise en compte du changement minimal du statut des arguments ou l’ajout de contraintes d’intégrité.

Remerciements

Ce travail a bénéficié d’une aide de l’Agence Nationale de la Recherche portant la référence ANR-13-BS02-0004 dans le cadre du projet AMANDE.

Références

- [1] Ringo Baumann. What does it take to enforce an argument? Minimal change in abstract argumentation. In *ECAI’12*, pages 127–132, 2012.
- [2] Ringo Baumann and Gerhard Brewka. Expanding argumentation frameworks : Enforcing and monotonicity results. In *COMMA’10*, pages 75–86, 2010.
- [3] Philippe Besnard and Sylvie Doutre. Checking the acceptability of a set of arguments. In *NMR’04*, pages 59–64, 2004.
- [4] Pierre Bisquert, Claudette Cayrol, Florence Dupin de Saint Cyr, and Marie-Christine Lagasquie-Schiex. Change in argumentation systems : exploring the interest of removing an argument. In *SUM’11*, pages 275–288, 2011.
- [5] Richard Booth, Dov Gabbay, Souhila Kaci, Tjitze Rienstra, and Leendert van der Torre. Abduction and dialogical proof in argumentation and logic programming. In *ECAI’14*, pages 117–122, 2014.
- [6] Richard Booth, Souhila Kaci, Tjitze Rienstra, and Leendert van der Torre. A logical theory about dynamics in abstract argumentation. In *SUM’13*, pages 148–161, 2013.
- [7] Claudette Cayrol, Florence Dupin de Saint Cyr, and Marie-Christine Lagasquie-Schiex. Change in abstract argumentation frameworks : Adding an argument. *J. Artif. Intell. Res.*, 38 :49–84, 2010.
- [8] Sylvie Coste-Marquis, Sébastien Konieczny, Jean-Guy Mailly, and Pierre Marquis. On the revision of argumentation systems : Minimal change of arguments statuses. In *KR’14*, pages 52–61, 2014.
- [9] Sylvie Coste-Marquis, Sébastien Konieczny, Jean-Guy Mailly, and Pierre Marquis. A translation-based approach for revision of argumentation frameworks. In *JELIA’14*, pages 397–411, 2014.
- [10] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artif. Intell.*, 77(2) :321–357, 1995.
- [11] Wolfgang Dvorák, Matti Järvisalo, Johannes Peter Wallner, and Stefan Woltran, 2011. see <http://www.dbai.tuwien.ac.at/research/project/argumentation/dynpartix/examples/>.
- [12] Wolfgang Dvorák, Matti Järvisalo, Johannes Peter Wallner, and Stefan Woltran. Complexity-sensitive decision procedures for abstract argumentation. *Artif. Intell.*, 206 :53–78, 2014.
- [13] IBM. IBM ILOG CPLEX Optimization Studio : Optimization model development toolkit for mathematical and constraint programming, 2014.
- [14] Dionysios Kontarinis, Elise Bonzon, Nicolas Mauudet, Alan Perotti, Leon van der Torre, and Serena Villata. Rewriting rules for the computation of goal-oriented changes in an argumentation system. In *CLIMA XIV*, pages 51–68, 2013.
- [15] Matthias Thimm and Serena Villata. First International Competition on Computational Models of Argumentation (ICCM’15), 2015. see <http://argumentationcompetition.org/2015/>.

Révision de bases de croyances : complexité de la vérification de modèle*

Nadia Creignou¹Raida Ktari^{1,2}Odile Papini²¹ Aix-Marseille Université, CNRS, LIF UMR 7279, 13288 Marseille, France² Aix-Marseille Université, CNRS, LSIS UMR 7296, 13288 Marseille, France

{nadia.creignou, raida.ktari, odile.papini}@univ-amu.fr

Résumé

Cet article porte sur la complexité du problème de vérification de modèle pour des opérateurs de révision de bases de croyances. Nous poursuivons l'étude initiée par Liberatore & Schaerf et introduisons deux nouveaux opérateurs de révision de bases de croyances reposant sur les sous-bases de croyances cohérentes de cardinalité maximale. Nous établissons la complexité de ce problème pour divers opérateurs dans le cas de la logique propositionnelle et dans le cas du fragment de Horn.

Abstract

This paper deals with the complexity of model checking for belief bases revision. We extend the study initiated by Liberatore & Schaerf and introduce two new belief base revision operators stemming from consistent subbases maximal with respect to cardinality. We establish the complexity of this problem for various operators within the framework of propositional logic as well as in the Horn fragment.

1 Introduction

La révision de croyances est une problématique importante en intelligence artificielle. Dans de nombreuses situations un agent dispose d'informations incomplètes, imparfaites ou incertaines et il émet des hypothèses qui peuvent être contredites lors de l'arrivée d'une nouvelle information considérée comme plus fiable. L'opération de révision consiste alors à restaurer la cohérence en conservant la nouvelle information tout en modifiant le moins possible les croyances initiales de l'agent. Ces principes ont été formalisés en

termes de postulats (*postulats AGM*) [1] et de nombreux opérateurs ont été proposés dans la littérature, qui sont classés selon deux points de vue, sémantique [11] et syntaxique.

D'un point de vue syntaxique, les croyances de l'agent sont représentées par des bases de croyances, c'est-à-dire des ensembles finis de formules propositionnelles. La plupart des approches reposent sur la construction de sous-bases cohérentes maximales selon différents critères [9, 20, 17, 3, 12, 10, 2]. Hansson a montré qu'il semble naturel de réviser des bases de croyances définies explicitement et d'étendre ensuite ces opérations à des ensembles de croyances en considérant la clôture déductive du résultat de la révision [10]. C'est le point de vue que nous adoptons dans cet article.

Concernant la complexité de la révision de croyances, les premiers travaux [14, 6, 15, 4] ont porté sur le problème d'inférence. A partir d'un ensemble de croyances, d'une nouvelle information et d'une requête (une formule propositionnelle), le problème d'inférence consiste à décider si la requête est une conséquence logique de l'ensemble de croyances révisé par la nouvelle information. Liberatore et Schaerf [13] ont ensuite considéré un autre problème, qu'ils considèrent comme une tâche plus basique, à savoir la vérification de modèle. Etant donné un ensemble de croyances et une nouvelle information, il s'agit de décider si une interprétation est un modèle de l'ensemble de croyances révisé par la nouvelle information. Bien que les problèmes d'inférence et de vérification de modèle soient liés, les résultats de complexité pour le problème de vérification de modèle ne s'obtiennent pas automatiquement à partir de ceux du problème d'inférence [13].

Liberatore & Shaerf [13] se sont concentrés sur les opérateurs syntaxiques reposant sur des sous-bases co-

*Ce travail a bénéficié de l'aide de l'Agence Nationale de la Recherche, projet ASPIQ portant la référence ANR-12-BS02-0003 et projet AGGREG portant la référence ANR-14-CE25-0017-01.

hérentes maximales selon l'inclusion ensembliste, en particulier l'opérateur de Ginsberg [9] et l'opérateur Widtjo [20]. Dans cet article, nous définissons un nouvel opérateur RSRW reposant sur des sous-bases cohérentes de cardinalité maximale, et sa généralisation PRSRW à des bases de croyances stratifiées. Nous présentons tous ces opérateurs dans un cadre formel unifié, puis nous étudions la complexité du problème de vérification de modèle pour chacun d'entre eux, dans le cas propositionnel ainsi que dans le cas du fragment de Horn. Nous obtenons ainsi une vue synthétique de l'impact des différentes stratégies et des différents critères de maximalité sur la complexité du problème.

2 Préliminaires

2.1 Notations

Nous nous plaçons dans le cadre de la logique propositionnelle. Nous rappelons qu'un littéral est un atome (littéral positif) ou la négation d'un atome (littéral négatif). Pour tout ensemble d'atomes A , on note $\text{Lit}(A)$ l'ensemble des littéraux construits à partir de A . Une formule CNF est une conjonction de clauses. Une clause est une disjonction de littéraux. Une clause est dite de Horn si elle comporte au plus un littéral positif. Une formule de Horn est une formule formée par la conjonction de clauses de Horn. Soit φ une formule, nous notons $\text{Mod}(\varphi)$ l'ensemble des modèles de φ . Une formule ψ est une conséquence logique de φ , noté $\varphi \models \psi$, si $\text{Mod}(\varphi) \subseteq \text{Mod}(\psi)$, et les deux formules sont équivalentes, noté $\varphi \equiv \psi$, si $\text{Mod}(\varphi) = \text{Mod}(\psi)$. Nous notons $Cn(\varphi)$ l'ensemble des conséquences logiques de φ . Une théorie T est telle que $T = Cn(T)$, c'est-à-dire que c est un ensemble de formules déductivement clos.

Dans la suite, nous manipulons des bases de croyances, c'est-à-dire des ensembles finis de formules propositionnelles. Si $B = \{\varphi_1, \varphi_2, \dots, \varphi_n\}$ est un tel ensemble, on note $\bigwedge B$ la conjonction de ses formules $\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n$. Ainsi, étant donné une famille d'ensembles finis de formules $\mathcal{W} = \{B_1, \dots, B_p\}$, la notation $\bigvee_{i=1}^p \bigwedge B_i$ sera utilisée pour représenter $\bigvee_{i=1}^p \bigwedge_{\varphi \in B_i} \varphi$.

2.2 Classes de complexité

Les classes P et NP sont les classes des problèmes décidables en temps polynomial déterministe, resp. non déterministe; la classe coNP est le complémentaire de la classe NP [16]. Un oracle pour une classe \mathcal{A} est un algorithme qui peut donner instantanément la réponse à un problème de la classe \mathcal{A} . Pour une classe de complexité \mathcal{C} , on notera $\mathcal{C}^{\mathcal{A}}$ la classe des problèmes qui

peuvent être résolus par un algorithme de la classe \mathcal{C} à condition que l'on puisse faire appel à un oracle pour \mathcal{A} . Stockmeyer définit inductivement la hiérarchie polynomiale, dont le premier niveau est constitué des classes P, NP et coNP [18]. Dans cette hiérarchie nous utiliserons en particulier des classes du second niveau, à savoir $\Sigma_2\text{P} = \text{NP}^{\text{NP}}$, $\Delta_2\text{P} = \text{P}^{\text{NP}}$. Nous utiliserons également la classe $\Theta_2\text{P} = \text{P}^{\text{NP}[\log(n)]}$ introduite par Wagner dans [19] et qui constitue une sous classe de $\Delta_2\text{P}$; les problèmes de $\Theta_2\text{P}$ sont ceux de $\Delta_2\text{P}$ qui peuvent être résolus en temps polynomial avec seulement un nombre logarithmique d'appels à l'oracle de NP.

3 Révision de bases de croyances

Un opérateur de révision syntaxique $*$ prend en entrée une base de croyances B et une formule μ représentant une nouvelle information, et renvoie une nouvelle base de croyances $B * \mu$. Ces opérateurs sont basés sur la recherche de $\mathcal{W}(B, \mu)$, l'ensemble des sous-bases maximales de B cohérentes avec la nouvelle information μ . Ils exploitent ensuite cet ensemble pour définir la base de croyances révisée selon une certaine stratégie. Nous distinguons deux stratégies. La première considère toutes les sous-bases maximales aussi plausibles les unes que les autres. La seconde repose sur l'intersection des sous-bases maximales cohérentes, c'est-à-dire ne retient que les informations qui ne sont jamais remises en question.

Nous définissons deux opérateurs utilisant la cardinalité comme critère de maximalité, RSRG et RSRW. Dans de nombreuses applications le critère de cardinalité est utilisé car l'acquisition d'information est coûteuse. Par ailleurs, l'extension de RSRG aux ensembles de croyances vérifie tous les postulats AGM [2]. Ces deux opérateurs correspondent aux deux stratégies que nous considérons, l'une prudente (RSRW), l'autre plus permissive (RSRG).

Nous considérons l'ensemble des sous-bases cohérentes maximales par cardinalité, $\mathcal{W}_{card}(B, \mu) = \{B_1 \subseteq B \mid \bigwedge B_1 \not\models \neg\mu \text{ et pour tout } B_2 \subseteq B \text{ tel que } |B_1| < |B_2|, \bigwedge B_2 \models \neg\mu\}$.

Les deux stratégies précédentes donnent alors lieu à deux opérateurs RSRG [2] et RSRW¹ définis comme suit :

$$\begin{aligned} B *_{\text{RSRG}} \mu &= \bigvee_{B' \in \mathcal{W}_{card}(B, \mu)} \bigwedge (B' \cup \{\mu\}), \\ B *_{\text{RSRW}} \mu &= \bigwedge \bigcap_{B' \in \mathcal{W}_{card}(B, \mu)} B' \cup \{\mu\}. \end{aligned}$$

Une interprétation m est un modèle de la base de

1. La notation RSR provient de l'expression "Removed Sets Revision" qualifiant les opérateurs basés sur le retrait du plus petit nombre de formules de la base initiale [2]

croyances révisée, $m \models B * \mu$, si et seulement si m est un modèle de μ et

- d'au moins un ensemble de $\mathcal{W}_{card}(B, \mu)$, dans le cas de RSRG,
- de chacune des formules apparaissant dans toutes les sous-bases cohérentes maximales, c'est-à-dire dans tous les ensembles de $\mathcal{W}_{card}(B, \mu)$, dans le cas de RSRW.

Ces deux opérateurs sont analogues à l'opérateur de Ginsberg, $*_G$, [9] et l'opérateur Widtio, $*_{wid}$, [20]. Ces derniers utilisent l'inclusion ensembliste comme critère de maximalité et considèrent donc $\mathcal{W}_{\subseteq}(B, \mu) = \{B_1 \subseteq B \mid \bigwedge B_1 \not\models \neg\mu \text{ et pour tout } B_2 \text{ tel que } B_1 \subset B_2 \subseteq B, \bigwedge B_2 \models \neg\mu\}$, au lieu de \mathcal{W}_{card} . Ils sont définis par :

$$\begin{aligned} B *_G \mu &= \bigvee_{B' \in \mathcal{W}_{\subseteq}(B, \mu)} \bigwedge (B' \cup \{\mu\}), \\ B *_{wid} \mu &= \bigwedge \bigcap_{B' \in \mathcal{W}_{\subseteq}(B, \mu)} (B' \cup \{\mu\}). \end{aligned}$$

Les opérateurs RSRG et RSRW peuvent être étendus à des bases de croyances stratifiées. Une base de croyances stratifiée $B = (S_1, \dots, S_n)$ est donnée par une partition de la base de croyances en strates S_i ($1 \leq i \leq n$) représentant différentes priorités entre formules.

Etant donné $X \subseteq B$ un ensemble de formules, nous définissons $trace(X, B)$ par le n -uplet d'entiers formé par : $trace(X, B) = (|X \cap S_1|, \dots, |X \cap S_n|)$. L'ordre lexicographique usuel, noté \leq_{lex} , sur les traces fournit un nouveau critère de maximalité pour les sous-bases cohérentes. Nous définissons l'ensemble des sous-bases cohérentes maximales selon ce critère par $\mathcal{W}_{cardlex}(B, \mu) = \{B_1 \subseteq B \mid \bigwedge B_1 \not\models \neg\mu \text{ et pour tout } B_2 \subseteq B \text{ tel que } trace(B_1, B) <_{lex} trace(B_2, B), \bigwedge B_2 \models \neg\mu\}$.

Observons que tous les éléments de $\mathcal{W}_{cardlex}(B, \mu)$ ont la même trace, que nous noterons $Tracemax(B, \mu)$. De plus, on a $\mathcal{W}_{cardlex}(B, \mu) \subseteq \mathcal{W}_{\subseteq}(B, \mu)$. Les opérateurs PRSRG [2, 3] et PRSRW sont alors définis comme suit :

$$\begin{aligned} B *_{PRSRG} \mu &= \bigvee_{B' \in \mathcal{W}_{cardlex}(B, \mu)} \bigwedge (B' \cup \{\mu\}), \\ B *_{PRSRW} \mu &= \bigwedge \bigcap_{B' \in \mathcal{W}_{cardlex}(B, \mu)} (B' \cup \{\mu\}). \end{aligned}$$

Lorsqu'une base de croyances n'est pas stratifiée, la cardinalité et la trace coïncident et les opérateurs PRSRG et PRSRW coïncident alors respectivement avec les opérateurs RSRG et RSRW.

4 Complexité de la vérification de modèle

Nous nous intéressons à la complexité du problème de vérification de modèle défini comme suit :

Problème : MODEL-CHECKING($*$)

Instance : B une base de croyances, φ une formule, m une interprétation

Question : $m \models B * \varphi$?

Observons que tout modèle de $B *_{RSRG} \mu$ est un modèle de $B *_G \mu$ et à l'inverse tout modèle de $B *_{wid} \mu$ est un modèle de $B *_{RSRW} \mu$.

Nous étudions successivement la complexité de ce problème selon la stratégie qui est utilisée.

4.1 Complexité pour les opérateurs $*_G, *_{RSRG}$ et $*_{PRSRG}$

La complexité de ce problème a été étudiée par Liberatore et Schaerf dans [13] pour l'opérateur de Ginsberg.

Théorème 4.1 [13] MODEL-CHECKING($*_G$) est coNP-complet dans le cas général de la logique propositionnelle, et dans P dans le cas des formules de Horn.

Ainsi, pour cet opérateur, la restriction à des formules de Horn rend le problème de la vérification de modèle traitable efficacement. Nous montrons dans le théorème suivant que quand le critère utilisé pour la maximalité des sous-bases cohérentes est celui de la cardinalité, la complexité dans le cas propositionnel reste inchangée. En revanche, avec ce critère la restriction aux formules de Horn ne rend pas le problème plus facile à traiter.

Théorème 4.2 MODEL-CHECKING($*_{RSRG}$) et MODEL-CHECKING($*_{PRSRG}$) sont coNP-complets dans le cas général de la logique propositionnelle, aussi bien que dans le cas des formules de Horn.

Preuve : Soit (B, μ, m) où B est une base de croyances stratifiée, une instance de MODEL-CHECKING($*_{PRSRG}$). Si m n'est pas un modèle de μ , alors $m \not\models B *_{PRSRG} \mu$. Sinon, un seul sous-ensemble B' de B est candidat à être de trace maximale parmi tous les sous-ensembles de B cohérents avec μ ayant m comme modèle. Il s'agit de $B' = \{\alpha \in B \mid m \models \alpha\}$. Pour montrer que $m \not\models B *_{PRSRG} \mu$, il suffit de montrer que B' n'est pas de trace maximale. Pour cela il suffit de deviner un ensemble $B_0 \subseteq B$ et une interprétation m_0 tels que $m_0 \models \bigwedge (B_0 \cup \{\mu\})$ (et donc $B_0 \cup \{\mu\}$ est cohérent) et $trace(B', B) <_{lex} trace(B_0, B)$. Ces vérifications pouvant être faites en temps polynomial, cela prouve que MODEL-CHECKING($*_{PRSRG}$) (et *a fortiori* MODEL-CHECKING($*_{RSRG}$)) est dans coNP.

Montrons maintenant la coNP-difficulté de MODEL-CHECKING($*_{RSRG}$) (et *a fortiori* celle de MODEL-CHECKING($*_{PRSRG}$)) dans le cas des formules de Horn. Nous utilisons une réduction du problème MAX-INDEPENDENT-SET, défini ci-dessous et bien connu pour être NP-complet (voir par exemple [8]), au complémentaire de MODEL-CHECKING($*_{RSRG}$).

Problème : MAX-INDEPENDENT-SET

Instance : $G = (V, E)$ un graphe non orienté, k un entier.

Question : Existe-t-il dans G un stable de taille supérieure ou égale à k , c'est-à-dire $V' \subseteq V$ avec $|V'| \geq k$, tel que pour tout couple de sommets $\{x, y\} \in V'^2$, $\{x, y\} \notin E$?

Soit $G = (V, E)$ un graphe non orienté et k un entier. A chaque sommet de G nous associons une variable propositionnelle de même nom, et nous considérons $U = \{u_1, \dots, u_{k-1}\}$ un ensemble de $k-1$ nouvelles variables propositionnelles. Nous considérons alors B une base de croyances, μ une formule et m une interprétation définies comme suit :

- $B = \{(v_i) \mid v_i \in V\} \cup \{(u_j) \mid u_j \in U\} \cup \{(\neg v_i \vee \neg v_j) \mid \{v_i, v_j\} \in E\} \cup \{(\neg v_i \vee \neg u_j) \mid v_i \in V, u_j \in U\}$
- $\mu = s$ et $m = U \cup \{s\}$ où s est une nouvelle variable n'apparaissant pas dans B .

Montrons que G admet un stable de taille supérieure ou égale à k si et seulement si $m \models B *_{RSRG} \mu$.

Observons que toute sous-base cohérente de B est cohérente avec μ , de plus il existe toujours une sous-base cohérente de B de cardinalité maximale qui contient toutes les clauses binaires. En effet s'il manque une clause $(\neg v_i \vee \neg w_j)$ dans B , par maximalité cela signifie que B contient à la fois v_i et w_j . Puisque toutes les clauses binaires sont négatives, l'ensemble $B' = B \cap \{(\neg v_i \vee \neg w_j)\} \setminus \{v_i\}$ est cohérent, de même cardinalité que B et contient une clause binaire de plus. Finalement un tel ensemble ne peut contenir à la fois une clause unaire issue de V et une issue de U .

Supposons que G admet un stable W de taille supérieure ou égale à k . L'ensemble $B' = \{(v_i) \mid v_i \in W\} \cup \{(\neg v_i \vee \neg v_j) \mid \{v_i, v_j\} \in E\} \cup \{(\neg v_i \vee \neg u_j) \mid v_i \in V, u_j \in U\}$ est alors cohérent (satisfait par l'interprétation W). Puisque $|W| \geq k$, B' prouve que l'ensemble des clauses de B satisfaites par m n'est pas de cardinalité maximale, ce qui prouve que $m \not\models B *_{RSRG} \mu$.

Réciproquement, conformément aux observations faites ci-dessus, si G n'admet pas de stable de taille supérieure ou égale à k , alors l'ensemble $\{(u_j) \mid u_j \in U\} \cup \{(\neg v_i \vee \neg v_j) \mid \{v_i, v_j\} \in E\} \cup \{(\neg v_i \vee \neg u_j) \mid v_i \in V, u_j \in U\}$ est de cardinalité maximale. Or cet ensemble est l'ensemble des clauses de B satisfaites par m , et donc $m \models B *_{RSRG} \mu$. \blacksquare

4.2 Complexité pour les opérateurs $*_{\text{wid}}$, $*_{\text{RSRW}}$ et $*_{\text{PRSRW}}$

Nous étudions maintenant la complexité des opérateurs qui utilisent une stratégie plus prudente, à savoir les opérateurs Widdio, RSRW et PRSRW. Dans le cas propositionnel, la vérification de modèle pour ces opérateurs se situe au deuxième niveau de la hiérarchie polynomiale.

Théorème 4.3 [13] $\text{MODEL-CHECKING}(*_{\text{wid}})$ est $\Sigma_2\text{P}$ -complet dans le cas général de la logique propositionnelle.

A notre connaissance seul ce résultat était connu jusqu'à présent, et en particulier la complexité exacte de ce problème dans le cas des formules de Horn était encore une question ouverte. Nous répondons à cette question en montrant que la restriction aux formules de Horn fait descendre le problème d'un niveau dans la hiérarchie polynomiale.

Théorème 4.4 $\text{MODEL-CHECKING}(*_{\text{wid}})$ est NP-complet dans le cas des formules de Horn.

Preuve : Pour prouver que $m \models B *_{\text{wid}} \mu$, il faut montrer que pour tout $\alpha \in B$ tel que $m \not\models \alpha$, il existe $B'_\alpha \subseteq B$ tel que $B'_\alpha \cup \{\mu\}$ est cohérent et $B'_\alpha \cup \{\mu\} \cup \{\alpha\}$ est incohérent (un tel ensemble peut être complété jusqu'à devenir maximal par inclusion).

Toutes les vérifications de cohérence sont exécutables en temps polynomial quand les formules sont de Horn. Il s'agit donc d'un problème dans NP.

Dans un second temps, nous établissons la NP-difficulté par réduction à partir du problème PQ-ABDUCTION défini comme suit et prouvé être NP-complet pour les formules de Horn par Creignou et Zanuttini dans [5] :

Problème : PQ-ABDUCTION

Instance : φ une formule de Horn, $A = \{x_1, \dots, x_n\}$ un ensemble de variables tel que $A \subseteq \text{Var}(\varphi)$, $q \in \text{Var}(\varphi) \setminus A$ une variable.

Question : Existe-t-il un ensemble $E \subseteq \text{Lit}(A)$ tel que $\varphi \wedge \bigwedge E$ est satisfaisable et $\varphi \wedge \bigwedge E \wedge \neg q$ est insatisfaisable ?

Soit (φ, A, q) une instance du problème PQ-ABDUCTION. Supposons sans perte de généralité que la formule de Horn $\varphi \wedge q$ est satisfaisable. Considérons B un ensemble de clauses, μ une formule et m une interprétation, définis de la façon suivante :

- $B = \{(\varphi \wedge l) \vee \neg x_0 \mid l \in \text{Lit}(A)\} \cup \{\neg q \wedge x_0\}$.
- $\mu = x_0$.

- Choisissons pour m une interprétation telle $m \models \varphi \wedge x_0 \wedge q$.

Remarquons que puisque φ est une formule de Horn toutes les formules de B peuvent être écrites comme des formules de Horn.

Montrons qu'il existe $E \subseteq \text{Lit}(A)$ tel que $\varphi \wedge \bigwedge E$ est satisfaisable et $\varphi \wedge \bigwedge E \wedge \neg q$ est insatisfaisable si et seulement si $m \models B *_{\text{wid}} \mu$, autrement dit, pour tout $\alpha \in B$ tel que $m \not\models \alpha$, il existe $B'_\alpha \subseteq B$ tel que $B' \cup \{\mu\}$ est cohérent et $B'_\alpha \cup \{\mu\} \cup \{\alpha\}$ est incohérent.

Dans un premier temps considérons un ensemble $E \subseteq \text{Lit}(A)$ tel que $\varphi \wedge \bigwedge E$ est satisfaisable et $\varphi \wedge \bigwedge E \wedge \neg q$ est insatisfaisable. Commençons par examiner les formules α de B dont m n'est pas un modèle. Il y a deux cas possibles :

- Si $\alpha = (\neg q \wedge x_0)$, prenons $B'_\alpha = \{(\varphi \wedge l) \vee \neg x_0 \mid l \in E\}$. On a alors $\bigwedge(B'_\alpha \cup \{\mu\}) \equiv ((\varphi \wedge \bigwedge E) \vee \neg x_0) \wedge x_0$. Cette formule est satisfaisable puisque x_0 n'apparaît ni dans φ ni dans E . En revanche, $\bigwedge(B'_\alpha \cup \{\mu\} \cup \{\alpha\}) \equiv \varphi \wedge \bigwedge E \wedge x_0 \wedge \neg q$ est insatisfaisable.
- Si $\alpha = ((\varphi \wedge l) \vee \neg x_0)$ pour un certain l . Si $m \not\models \alpha$, alors $m \not\models l$ puisque $m \models \varphi$. Prenons alors $B'_\alpha = \{(\varphi \wedge \neg l) \vee \neg x_0\}$. Puisque $m \models \varphi$ et $m \not\models l$ (et donc $m \models \neg l$), l'unique formule de B'_α est satisfaite par m , et donc $B'_\alpha \cup \{\mu\}$ est cohérent. D'autre part, $B'_\alpha \cup \{\mu\} \cup \{\alpha\} = \{(\varphi \wedge \neg l) \vee \neg x_0\} \cup \{x_0\} \cup \{(\varphi \wedge l) \vee \neg x_0\}$ est incohérent.

On a donc montré que s'il existe un ensemble $E \subseteq \text{Lit}(A)$ tel que $\varphi \wedge \bigwedge E$ est satisfaisable et $\varphi \wedge \bigwedge E \wedge \neg q$ est insatisfaisable, alors $m \models B *_{\text{wid}} \mu$.

Réciproquement supposons que $m \models B *_{\text{wid}} \mu$. Autrement dit, supposons que pour tout $\alpha \in B$ tel que $m \not\models \alpha$ il existe un sous ensemble $B'_\alpha \subseteq B$ tel que $B'_\alpha \cup \{\mu\}$ est cohérent et $B'_\alpha \cup \{\mu\} \cup \{\alpha\}$ est incohérent. Considérons $\alpha = \neg q \wedge x_0$ et prenons $E = \{l \mid ((\varphi \wedge l) \vee \neg x_0) \in B'_\alpha\}$. D'une part, observons que $\varphi \wedge \bigwedge E \wedge x_0 \equiv \bigwedge(B'_\alpha \cup \{\mu\})$ et donc $\varphi \wedge \bigwedge E$ est satisfaisable. D'autre part, $\varphi \wedge \bigwedge E \wedge x_0 \wedge \neg q \equiv \bigwedge(B'_\alpha \cup \{\mu\} \cup \{\alpha\})$ et donc $\varphi \wedge \bigwedge E \wedge \neg q$ est insatisfaisable.

Enfin, observons que la réduction est calculable en temps polynomial. Le seul point critique est de choisir m qui soit un modèle de φ . Or φ est une formule de Horn. Il est donc possible de trouver un modèle de φ en temps polynomial. On a donc finalement bien démontré que $\text{MODEL-CHECKING}(*_{\text{wid}})$ est NP-complet dans le cas des formules de Horn. \blacksquare

Pour cette stratégie extrêmement prudente, contrairement à la stratégie précédente, considérer la cardinalité comme critère de maximalité rend le problème dans le cas propositionnel plus facile, puisqu'il se situe dans une sous-classe de $\Sigma_2\text{P}$. En effet, nous obtenons

les résultats suivants pour les problèmes $\text{MODEL-CHECKING}(*_{\text{RSRW}})$ et $\text{MODEL-CHECKING}(*_{\text{PRSRW}})$.

Théorème 4.5 *Dans le cas général de la logique propositionnelle,*

- $\text{MODEL-CHECKING}(*_{\text{RSRW}})$ est dans $\Theta_2\text{P}$ et est coNP-difficile,
- $\text{MODEL-CHECKING}(*_{\text{PRSRW}})$ est dans $\Delta_2\text{P}$ et est coNP-difficile.

Preuve : Commençons par établir l'appartenance de $\text{MODEL-CHECKING}(*_{\text{RSRW}})$ à la classe $\Theta_2\text{P}$. Soit B une base de croyances et μ une formule. On note k_{max} la cardinalité maximale des sous-ensembles de B cohérents avec μ . Pour décider si m est un modèle de la base révisée $B *_{\text{RSRW}} \mu$, il faut vérifier que pour toutes les formules α de B , dont l'interprétation m n'est pas un modèle, il existe au moins un sous-ensemble B'_α de $B \setminus \{\alpha\}$ cohérent avec μ et de cardinalité égale à k_{max} . On peut donc utiliser l'algorithme suivant :

1. Vérifier que $m \models \mu$
 - sinon, on a $m \not\models B *_{\text{RSRW}} \mu$.
 - si oui, continuer.
2. Calculer k_{max} .
3. — Pour tout $\alpha \in B$ telle que $m \not\models \alpha$, existe-t-il un sous-ensemble $B'_\alpha \subseteq B \setminus \{\alpha\}$ tel que : $B'_\alpha \cup \{\mu\}$ est cohérent et $|B'_\alpha| = k_{\text{max}}$?
 - si oui, $m \models B *_{\text{RSRW}} \mu$,
 - sinon, $m \not\models B *_{\text{RSRW}} \mu$.

Le calcul de k_{max} se fait par une recherche dichotomique classique en posant des questions du type "Existe-t-il un sous-ensemble de B cohérent avec μ et de taille supérieure ou égale à k ?", et donc se fait par un nombre logarithmique d'appels à un oracle NP.

La troisième étape est réalisée en considérant toutes les formules $\alpha \in B$ en parallèle, et donc requiert un seul appel à un oracle NP (il suffit de deviner au plus $|B|$ couples (B'_α, m_α) où B'_α est un sous-ensemble de $B \setminus \{\alpha\}$ avec $|B'_\alpha| = k_{\text{max}}$, et m_α est une interprétation telle que $m_\alpha \models \bigwedge(B'_\alpha \cup \{\mu\})$).

Au total nous obtenons un algorithme polynomial qui utilise un nombre logarithmique d'appels à un oracle NP, ce qui prouve bien que le problème $\text{MODEL-CHECKING}(*_{\text{RSRW}})$ est dans $\Theta_2\text{P}$.

Pour le problème $\text{MODEL-CHECKING}(*_{\text{PRSRW}})$, l'algorithme est similaire en considérant la trace au lieu de la cardinalité. Le calcul de la trace des sous-bases cohérentes maximale dans l'ordre lexicographique se fait en calculant successivement (et non en parallèle) les composantes de cette trace strate par strate, par des appels adaptatifs à un oracle NP. De ce fait le nombre d'appels à l'oracle NP dépend du nombre de

strates de la base de croyances, l'algorithme prouve alors l'appartenance du problème à la classe Δ_2P .

Pour les bornes inférieures il suffit de montrer que $\text{MODEL-CHECKING}(*_{RSRW})$ est coNP-difficile. Nous utilisons une réduction à partir du problème de l'insatisfaisabilité des formules propositionnelles. Etant donné une formule φ , considérons la base $B = \{\varphi \wedge r\}$, la formule $\mu = s$ et l'interprétation $m = \{s\}$ où r et s sont de nouvelles variables n'apparaissant pas dans φ . Il est clair que $B*_{RSRW}\mu = \varphi \wedge r \wedge s$ ou $B*_{RSRW}\mu = s$ selon que φ est satisfaisable ou non. Ainsi φ est insatisfaisable si et seulement si $m \models B*_{RSRW}\mu$. Cela prouve la coNP-difficulté de $\text{MODEL-CHECKING}(*_{RSRW})$. \blacksquare

Il est à noter que les bornes inférieure et supérieure de complexité obtenues pour $\text{MODEL-CHECKING}(*_{RSRW})$ et $\text{MODEL-CHECKING}(*_{PRSRW})$ ne coïncident pas et donc que la complexité exacte de ces problèmes reste ouverte. La complexité de ces problèmes quand ils sont restreints à des formules de Horn est elle aussi ouverte.

Le tableau suivant récapitule les résultats de complexité obtenus :

Op.	Log. prop.	Horn
Ginsberg	coNP-complet [13, Th. 1]	P [13, Th. 17]
Widtio	Σ_2P -complet [13, Th. 2]	NP-complet, Th. 4.4
RSRG	coNP-complet, Th. 4.2	coNP-complet, Th. 4.2
RSRW	dans Θ_2P , coNP-difficile, Th. 4.5	dans Θ_2P , Th. 4.5
PRSRG	coNP-complet, Th. 4.2	coNP-complet, Th. 4.2
PRSRW	dans Δ_2P , coNP-difficile, Th. 4.5	dans Δ_2P , Th. 4.5

5 Travaux connexes

Il est bien connu que la révision de bases de croyances et l'inférence non-monotone à partir d'une base de croyances incohérente sont les deux faces d'une même pièce [7]. En cas d'incohérence en présence d'une nouvelle information, deux attitudes sont possibles : soit réviser la base de croyances, soit définir une relation d'inférence non-monotone tolérant l'incohérence. Dans [4] Cayrol, Lagasquie-Schiex et T. Schiex présentent une étude comparative de certaines relations d'inférence syntaxique. Soit B une base de croyances, $<$ un pré-ordre total sur les formules de la base, et ϕ une formule propositionnelle, les relations d'inférence sont définies de façon synthétique par $(B, <) \mid \sim^{p,m} \phi$ où $p \in \{T, INCL, LEX\}$ représente le mécanisme

de sélection de sous bases maximales cohérentes, de sous-bases maximales cohérentes préférées pour l'inclusion ensembliste ou de sous-bases maximales cohérentes préférées pour l'ordre lexicographique respectivement et $m \in \{\forall, \exists, ARG\}$ représente la stratégie d'inférence, universelle, existentielle ou argumentative respectivement. L'étude comparative du point de vue de la complexité est présentée dans le cas propositionnel et le cas de Horn, cependant elle porte sur le problème d'inférence et non sur le problème de vérification de modèle. Par ailleurs, la stratégie d'intersection de sous-bases maximales cohérentes n'est pas envisagée.

6 Conclusion

Lorsque l'on s'intéresse aux opérateurs de révision de bases de croyances syntaxiques, on peut jouer sur deux paramètres, le critère de maximalité des sous-bases cohérentes et la stratégie pour exploiter les sous-bases cohérentes maximales. Lorsque le critère de maximalité est la cardinalité, nous avons rappelé la définition de l'opérateur RSRG et nous avons défini un nouvel opérateur RSRW dont la stratégie repose sur l'intersection des sous-bases cohérentes maximales, puis nous les avons généralisés au cas de bases de croyances stratifiées avec les opérateurs PRSRG et PRSRW. Ces opérateurs sont les analogues des opérateurs de Ginsberg et Widtio qui utilisent l'inclusion ensembliste comme critère de maximalité.

D'une part nous avons d'abord établi la complexité du problème de vérification de modèle pour l'opérateur Widtio dans le cas du fragment de Horn, répondant ainsi à une question laissée ouverte par Liberatore et Schaerf [13].

D'autre part nous avons étudié dans quelle mesure l'utilisation de la cardinalité comme critère de maximalité de sous-bases maximales cohérentes (au lieu de l'inclusion ensembliste) avait une influence sur la complexité du problème de vérification de modèle. Lorsque la stratégie considère toutes les sous-bases cohérentes aussi plausibles les unes que les autres, l'utilisation de la cardinalité comme critère de maximalité n'a pas d'influence sur la complexité du problème de vérification de modèle dans le cas propositionnel. Cependant, ce critère fait que la restriction aux formules de Horn ne rend pas le problème plus facile à traiter. En revanche, lorsque la stratégie repose sur l'intersection des sous-bases cohérentes maximales la complexité du problème de vérification de modèle diminue.

Il semble alors assez naturel de s'interroger sur l'impact de la cardinalité comme critère de maximalité pour d'autres stratégies de révision de bases de croyances, ce qui fera l'objet de futurs travaux.

Références

- [1] C. E. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change : Partial meet contraction and revision functions. *J. of Symb. Log.*, 50 :510–530, 1985.
- [2] S. Benferhat, J. Ben-Naim, O. Papini, and E. Würbel. An answer set programming encoding of prioritized removed sets revision : application to GIS. *Appl. Intell.*, 32(1) :60–87, 2010.
- [3] S. Benferhat, C. Cayrol, D. Dubois, J. Lang, and H. Prade. Inconsistency management and prioritized syntax-based entailment. In *Proc. of IJCAI'93*, pages 640–645, 1993.
- [4] C. Cayrol, M.-C. Lagasquie-Schiex, and T. Schiex. Nonmonotonic reasoning : From complexity to algorithms. *Ann. Math. Artif. Intell.*, 22(3-4) :207–236, 1998.
- [5] N. Creignou and B. Zanuttini. A Complete Classification of the Complexity of Propositional Abduction. *SIAM J. Comput.*, 36 :207–229, 2006.
- [6] T. Eiter and G. Gottlob. On the complexity of propositional knowledge base revision, updates, and counterfactuals. *Artif. Intell.*, 57(2-3) :227–270, 1992.
- [7] P. Gärdenfors. Belief revision and nonmonotonic logic : Two sides of the same coin? In *Proc. of ECAI'90*, pages 768–773, 1990.
- [8] M.R. Garey and D.S. Johnson. *Computers and Intractability : A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- [9] M.L. Ginsberg. Counterfactuals. *Artificial Intelligence*, 30 :35–79, 1986.
- [10] S. O. Hansson. Revision of belief sets and belief bases. In *Handb. of Defeas. Reas. and Uncert. Manag. Syst.*, volume 3, pages 17–75. Kluwer, 1998.
- [11] H. Katsuno and A. O. Mendelzon. Propositional knowledge base revision and minimal change. *Artif. Intell.*, 52 :263–294, 1991.
- [12] D. Lehmann. Belief revision, revised. In *Proc. of IJCAI'95*, pages 1534–1540, 1995.
- [13] P. Liberatore and M. Schaerf. Belief revision and update : Complexity of model checking. *J. Comput. Syst. Sci.*, 62(1) :43–72, 2001.
- [14] B. Nebel. Belief revision and default reasoning : Syntax-based approaches. In *Proc. of KR'91*, pages 417–428, 1991.
- [15] B. Nebel. How hard is it to revise a belief base? In *Belief Change*, volume 3 of *Handb. of Defeas. Reas. and Uncert. Manag. Syst.*, pages 77–145. Springer Netherlands, 1998.
- [16] C. H. Papadimitriou. *Computational Complexity*. Addison Wesley, 1994.
- [17] O. Papini. A complete revision function in propositional calculus. In *Proc. of ECAI'92*, pages 339–343, 1992.
- [18] L. J. Stockmeyer. The polynomial-time hierarchy. *Theor. Comput. Sci.*, 3(1) :1–22, 1976.
- [19] K. Wagner. More complicated questions about maxima and minima, and some closures of NP. *Theor. Comput. Sci.*, 51(1-2) :53–80, 1987.
- [20] M. Winslett. Sometimes updates are circumscription. In *Proc. of IJCAI'89*, pages 859–863, 1989.

Agrégation de Systèmes d’Argumentation

Jérôme Delobelle Sébastien Konieczny Srdjan Vesic

CRIL, CNRS – Université d’Artois

Lens, France

{delobelle,konieczny,vesic}@cril.fr

Résumé

Depuis quelques années, le problème de l’agrégation de systèmes d’argumentation abstraits de Dung est étudié. Plusieurs approches concernant l’agrégation ont été proposées, aussi bien pour les opérateurs que pour les propriétés. Ce papier décrit ces différentes approches et propose de nouveaux opérateurs d’agrégation. Nous étudions ces opérateurs à la lumière des propriétés existantes et montrons qu’ils ne satisfont pas beaucoup de propriétés. Les conclusions de ce travail sont que, d’une part, aucun de ces opérateurs existants ne satisfait toutes les propriétés, mais, d’autre part, certaines propriétés proposées semblent trop fortes.

Abstract

The problem of aggregation within Dung’s abstract argumentation frameworks has been studied by several researchers in recent years. Different aggregation approaches has been proposed as well as the properties they should satisfy. This paper departs from existing approaches and proposes new aggregation operators. We study these operators in light of the proposed properties and show that they do not satisfy a lot of proposed properties. The conclusions are that, on one hand, none of the existing operators seem fully satisfactory, but, on the other hand, some of the properties proposed so far seem also too demanding.

1 Introduction

L’argumentation est basée sur le principe d’interaction et d’évaluation des arguments. La théorie d’argumentation de Dung [12] s’appuie sur la notion de système d’argumentation représenté par un graphe binaire. Ce système prend en entrée un ensemble d’arguments et une relation binaire qui exprime une notion de contrariété (et plus particulièrement d’attaque) entre arguments. Le but étant de déterminer les ensembles d’arguments pouvant être conjointement acceptés à partir d’un graphe.

Le changement dans les systèmes d’argumentation est un sujet de recherche très actif depuis plusieurs années, avec notamment le problème d’évolution (révision) des systèmes d’argumentation [5, 3, 1, 11, 2]. Cependant, il existe peu de travaux concernant l’agrégation de systèmes d’argumentation. Celui-ci s’avère être un problème important pour les systèmes multi-agents : supposons que chaque membre d’un groupe d’agents possède son propre système d’argumentation, représentant ses croyances, le problème est de définir un système d’argumentation (social) qui représente les croyances du groupe. C’est une question importante en soi, mais également parce que le résultat peut être considérée comme la résultante d’un processus de négociation *idéal*, vis-à-vis duquel il est possible d’évaluer les résultats obtenus par des protocoles de négociation *réels* [4].

La majeure partie des travaux propose de nouveaux opérateurs d’agrégation [7, 20, 6, 15]. Dans cet article, nous nous focalisons sur les opérateurs conformes au cadre de Dung, ce qui signifie qu’ils doivent prendre, en entrée, un ensemble de systèmes d’argumentation abstraits classiques, et retourner soit un système d’argumentation abstrait (ou plus généralement un ensemble de systèmes d’argumentation), soit un ensemble d’extensions. C’est pourquoi certaines méthodes, telles que celle proposées par Gabbay et Rodrigues [15], retournant un ensemble ordonné d’arguments, ne sont pas prises en compte.

Une contribution intéressante est le travail effectué par Dunne et al. [14], où les auteurs définissent plusieurs propriétés “de rationalité” dédiées à ce type de méthodes d’agrégation sans, pour autant, vérifier si les opérateurs existants les satisfont.

C’est justement ce que nous proposons dans cet article. Nous vérifions quelles propriétés sont satisfaites par les opérateurs d’agrégation existants de la littérature. Nous proposons également une nouvelle méthode

d'agrégation basée sur les WAF (Weighted Argumentation Frameworks) [13, 8]. Dans ces travaux, une des interprétations possibles à l'ajout de poids sur les attaques est qu'ils représentent le nombre d'agents en accord avec cette attaque. Nous regardons donc comment formaliser cette idée afin de définir de nouveaux opérateurs d'agrégation.

Les conclusions de ce travail sont que, d'un côté, aucuns des opérateurs existants ne semble totalement satisfaisant, mais, d'un autre côté, certaines propriétés proposées paraissent trop fortes. Des travaux sont donc encore nécessaires à la fois dans l'élaboration de nouveaux opérateurs et dans l'étude des propriétés.

Le papier est organisé comme suit. Dans la section suivante, nous fournissons l'état de l'art requis sur les systèmes d'argumentation. Les sections 3 et 4 sont, respectivement, des rappels sur les *weighted argumentation frameworks* et les opérateurs d'agrégation existants. Dans la section 5, nous étudions les propriétés de ces opérateurs. Dans la section 6, nous vérifions quelles propriétés sont satisfaites par les opérateurs existants. Dans la section 7, nous proposons une nouvelle méthode d'agrégation (plus précisément trois variantes), et étudions ses propriétés. Enfin, dans la section 8, nous résumons et discutons des résultats obtenus avant de conclure en section 9.

2 Préliminaires

Dans cette section, nous rappelons brièvement quelques éléments clés sur les systèmes d'argumentation abstraits proposés par Dung [12].

Définition 1 *Un système d'argumentation (AF) est un couple $F = \langle A, R \rangle$ où A est un ensemble fini d'entités abstraites appelées **arguments** et R une relation binaire sur A , i.e. $R \subseteq A \times A$, appelée **relation d'attaque**. Un ensemble d'arguments $S \subseteq A$ attaque un argument $b \in A$, si $\exists a \in S$, tel que $(a, b) \in R$. Pour un système d'argumentation $F = \langle A, R \rangle$, notons $\text{Arg}(F) = A$ et $\text{Att}(F) = R$.*

L'objectif est de déterminer les ensembles d'arguments qui peuvent être conjointement acceptés. Introduisons d'abord les notions d'ensemble sans conflit et d'acceptabilité :

Définition 2 *Soit $F = \langle A, R \rangle$ un système d'argumentation. Un ensemble $S \subseteq A$ est **sans-conflit** ssi il ne contient pas deux arguments $a, b \in S$ tels que $(a, b) \in R$. Un argument $a \in A$ est **acceptable** pour un ensemble d'arguments S ssi pour chaque argument $b \in A$, si $(b, a) \in R$, alors b est attaqué par S .*

Un ensemble d'arguments est **admissible** quand il est sans conflit et que chacun de ses arguments est acceptable pour cet ensemble. Afin de déterminer le statut d'acceptation de chaque argument, Dung propose plusieurs sémantiques d'acceptabilité qui définissent des ensembles d'arguments - appelés **extensions** - pouvant être acceptés conjointement. Dans cet article, nous nous focalisons sur ces sémantiques, $\forall S \subseteq A$:

- S est une extension **complète** de F ssi c'est un ensemble admissible et tous les arguments, qui sont acceptables pour S , appartiennent à S .
- S est une extension **préférée** de F ssi c'est un ensemble admissible maximal pour l'inclusion.
- S est une extension **stable** de F ssi c'est un ensemble admissible maximal pour l'inclusion et attaque tous les arguments qui n'appartiennent pas à S .
- S est une extension **de base** (ou *grounded*) de F ssi c'est le plus petit point fixe de la fonction caractéristique de F ($H : 2^A \rightarrow 2^A$ avec $H(S) = \{a \mid a \text{ est acceptable pour } S\}$).

Notons $\mathcal{E}_\sigma(F)$ l'ensemble des extensions de F pour les sémantiques $\sigma \in \{\mathbf{comp}(lète), \mathbf{pref}(érée), \mathbf{sta}(ble), \mathbf{gr}(ounded)\}$.

Définissons, maintenant, le statut d'acceptabilité de chaque sous-ensemble d'arguments. Un argument a est **sceptiquement** accepté ssi il existe au moins une extension et a appartient à toutes les extensions. Un argument est **crédulement** accepté ssi il appartient à au moins une extension. Notons $sa_\sigma(F)$ (respectivement $ca_\sigma(F)$) l'ensemble des arguments sceptiquement (resp. crédulement) acceptés dans F .

3 Weighted Argumentation Frameworks

Dans cette partie, nous nous intéressons à une expansion des systèmes de Dung auxquels on ajoute un poids sur les attaques [13, 8, 9].

Définition 3 *Un Weighted Argumentation Framework est un triplet $WF = \langle A, R, w \rangle$ où $\langle A, R \rangle$ est un système d'argumentation de Dung, et $w : A \times A \rightarrow \mathbb{N}$ est une fonction qui associe un entier naturel à chaque attaque (i.e. $w(a, b) > 0$ ssi $(a, b) \in R$), zéro sinon (i.e. $w(a, b) = 0$ ssi $(a, b) \notin R$).*

Notons \overline{WF} le système d'argumentation standard correspondant au WF obtenu en retirant les poids, i.e. si $WF = \langle A, R, w \rangle$ alors $\overline{WF} = \langle A, R \rangle$. Nous montrons, plus tard, comment utiliser les WAF afin d'agréger plusieurs AF. Dans ce but, commençons par rappeler deux méthodes existantes visant à exploiter les poids d'un WAF : les *relaxing extensions* et les *best extensions*.

3.1 Relaxing Extensions

À l'origine, les WAF ont été introduit dans le but de garantir l'obtention d'extensions non-triviales [13]. L'idée est de supprimer certaines attaques afin de rendre les extensions non-triviales. Les définitions suivantes ont été définies par Coste-Marquis et al. [9] :

Définition 4 Une fonction d'agrégation \oplus est une fonction de \mathbb{N}^n vers \mathbb{N} telle que :

- si $x_i \geq x'_i$, alors

$$\oplus(x_1, \dots, x_i, \dots, x_n) \geq \oplus(x_1, \dots, x'_i, \dots, x_n)$$
- $\oplus(x_1, \dots, x_n) = 0$ ssi pour tout i , $x_i = 0$
- $\oplus(x) = x$.

Définition 5 Soient $WF = \langle A, R, w \rangle$ un *weighted argumentation framework*, \otimes une fonction d'agrégation et σ une sémantique. L'agrégation des poids sur les attaques pour un ensemble $S \subseteq R$ est $w_{\otimes}(S, w) = \otimes_{(a,b) \in S} w(a, b)$.

La fonction $Sub(R, w, \beta)$ retourne l'ensemble des sous-ensembles de R dont le poids total n'excède pas β : $Sub(R, w, \beta) = \{S \mid S \subseteq R \text{ et } w_{\otimes}(S, w) \leq \beta\}$. L'ensemble des *relaxing extensions* (σ_{\otimes}^{β} -extensions) de WF , noté $\mathcal{E}_{\sigma}^{\otimes, \beta}(WF)$, est défini comme suit : $\mathcal{E}_{\sigma}^{\otimes, \beta}(\langle A, R, w \rangle) = \{E \in \mathcal{E}_{\sigma}(\langle A, R \setminus S \rangle) \mid S \in Sub(R, w, \beta)\}$.

Contrairement aux ensembles de Dung où il existe une extension de base unique, il est possible d'en obtenir plusieurs en utilisant les *relaxing extensions*. De plus, il existe des cas où l'ensemble vide est présent dans l'ensemble des *relaxing extensions*, ce qui peut poser problème lors de l'utilisation de l'inférence sceptique. La définition suivante permet de supprimer cet ensemble vide de l'ensemble des extensions tout en choisissant la valeur de β la plus intéressante, ce qui revient à prendre la plus petite valeur permettant d'obtenir une extension non-triviale (pour la sémantique choisie) :

Définition 6 Etant donné un *weighted argumentation framework* $WAF = \langle A, R, w \rangle$, une sémantique σ et une fonction d'agrégation \otimes , l'ensemble des σ_{\otimes} -extensions du WAF, noté $\mathcal{E}_{\sigma}^{\otimes}(WF)$ est défini comme $\mathcal{E}_{\sigma}^{\otimes}(\langle A, R, w \rangle) = \underline{\mathcal{E}_{\sigma}^{\otimes, \beta}(\langle A, R, w \rangle)}$ où¹ :

- $\mathcal{E}_{\sigma}^{\otimes, \beta}(\langle A, R, w \rangle)$ est non-trivial²,
- Il n'existe pas de $\beta' < \beta$ tel que $\mathcal{E}_{\sigma}^{\otimes, \beta'}(\langle A, R, w \rangle)$ est non-trivial,
- Pour un ensemble \mathcal{E} d'extensions, $\underline{\mathcal{E}} = \mathcal{E} \setminus \{\emptyset\}$.

1. Nous ajoutons une troisième condition comparé à [9], afin de directement encoder l'inférence sceptique non-triviale.

2. Un ensemble d'extensions est non-trivial s'il existe au moins une extension non-triviale.

3.2 Best Extensions

En général, un système d'argumentation peut admettre un nombre important d'extensions pour une sémantique donnée. En utilisant les WAF, Coste-Marquis et al. [8] montrent qu'il est possible de tirer avantage des poids sur les attaques, afin de sélectionner les extensions pouvant le mieux se défendre elles-mêmes. Ce processus consiste, dans un premier temps, à calculer puis à comparer les scores obtenues par chaque extensions :

Définition 7 Soit $WAF = \langle A, R, w \rangle$ un *weighted argumentation framework*. Soient E et F deux extensions de WF pour une sémantique donnée σ et \oplus une fonction d'agrégation. La \oplus -attaque de E vers F est : $S_{\oplus}(E \rightarrow F) = \oplus_{a \in E, b \in F} w(a, b)$. Alors $E >_{\oplus} F$ ssi $S_{\oplus}(E \rightarrow F) > S_{\oplus}(F \rightarrow E)$.

Une fois les scores calculés, Coste-Marquis et al. [8] proposent quatre façons intuitives d'exploiter l'ordre $>_{\oplus}$ afin de sélectionner les meilleures extensions. Par rapport à la définition originale, nous ajoutons un paramètre qui nous sera utile afin de définir de nouveaux opérateurs d'agrégation dans la section 7.

Définition 8 Soient WF un *weighted argumentation framework* et X un ensemble tel que $\forall x \in X, x \subseteq Arg(WF)$. Soit \oplus une fonction d'agrégation. Alors :

- $best_1^{\oplus}(X, WF) = \{E \in X \mid \nexists E' \in X, E' >_{\oplus} E\}$
- $best_2^{\oplus}(X, WF) = argmax_{E \in X} |\{E' \in X \mid E >_{\oplus} E'\}|$
- $best_3^{\oplus}(X, WF) = argmax_{E \in X} |\{E' \in X \mid E >_{\oplus} E'\}| - |\{E' \in X \mid E' >_{\oplus} E\}|$
- $best_4^{\oplus}(X, WF) = argmax_{E \in X} KS_{\oplus}(E)$,
avec $KS_{\oplus}(E) = min_{E' \in X, E' \neq E} (S_{\oplus}(E \rightarrow E'))$

Définition 9 Soient WF un *weighted argumentation framework* et \oplus une fonction d'agrégation. Alors, $\forall i \in \{1, 2, 3, 4\}$, $best_i^{\sigma, \oplus}(WF) = best_i^{\oplus}(\mathcal{E}_{\sigma}(WF), WF)$.

La première règle sélectionne les extensions qui ne sont jamais battues par d'autres extensions. Cette méthode peut retourner l'ensemble vide comme résultat, ce qui n'est pas le cas des autres fonctions. La seconde approche sélectionne les extensions en comptant celles qui battent le plus d'autres extensions. La troisième règle est similaire à la règle de Copeland [18], car le score d'une extension E correspond à la différence entre le nombre d'extensions battues par E et le nombre d'extensions qui battent E . Enfin, la dernière règle sélectionne les extensions ayant obtenues le score minimal le plus élevé contre d'autres extensions. Cette règle est étroitement liée à la règle de vote de Kramer-Simpson [19, 17].

4 Opérateurs d'Agrégation

Certains opérateurs d'agrégation de systèmes d'argumentation [7] se sont inspirés des opérateurs de fusion en logique propositionnelle [16]. Le but étant de sélectionner les systèmes d'argumentation étant les plus proches possibles de la position du groupe, en utilisant une notion de distance entre ces systèmes d'argumentation et le profil du groupe.

À partir de maintenant, nous faisons l'hypothèse que tous les systèmes d'argumentation sont définis sur le même ensemble d'arguments³ \mathcal{X} . Notons \mathbb{AF} l'ensemble de tous les systèmes d'argumentation définis à partir de \mathcal{X} .

Définition 10 Soient $p = \langle AF_1, \dots, AF_n \rangle$ un profil, d une distance entre AF , et \oplus une fonction d'agrégation. L'opérateur d'agrégation Δ_d^\oplus est défini comme suit : $\Delta_d^\oplus(\langle AF_1, \dots, AF_n \rangle) = \{AF \in \mathbb{AF} \mid AF \text{ minimise } \oplus_{i=1}^n d(AF, AF_i)\}$.

La distance utilisée dans l'article de Coste-Marquis et al. est l'*edit distance* (de), qui est équivalente, dans notre cas, à la cardinalité de la différence symétrique entre les deux relations d'attaque. Les exemples typiques de fonction d'agrégation sont la *somme* (Σ) et le *leximax*. Nous nous focaliserons sur ces deux fonctions dans la suite de l'article.

Inspirés des méthodes de vote, d'autres opérateurs d'agrégation ont été définis par Tohmé et al. [20] avec, en particulier, la méthode *qualified voting* :

Définition 11 Soient $p = \langle AF_1, \dots, AF_n \rangle$ un profil et $U \subset \{AF_1, \dots, AF_n\}$. *Qualified voting* est définie par $QV(\langle AF_1, \dots, AF_n \rangle) = (A, R)$, où A est l'ensemble des arguments utilisés par les agents, et $R = \{(a, b) \mid a, b \in A \text{ et } |\{AF_i : (a, b) \in Att(AF_i)\}| > \max(|\{AF_i : (b, a) \in Att(AF_i)\}|, |\{AF_i : (a, b) \notin Att(AF_i) \text{ et } (b, a) \notin Att(AF_i)\}|)\}$ et $U \subseteq \{AF_i \mid (a, b) \in Att(AF_i)\}$.

5 Propriétés des Fonctions d'Agrégation

Dunne et al. [14] proposent plusieurs propriétés rationnelles visant à caractériser l'agrégation d'un ensemble de systèmes d'argumentation. Ces propriétés sont basées sur l'adaptation des propriétés provenant de la théorie de choix social au cadre des systèmes d'argumentation.

Rappelons que \mathbb{AF} représente l'ensemble de tous

3. Nous simplifions les définitions des AF en entrée par rapport aux systèmes d'argumentation partiels définis par Coste-Marquis et al. [7]. Ces systèmes permettent de représenter l'ignorance d'un agent sur le fait qu'une attaque existe ou non, mais nécessitent une étape préliminaire pour l'agrégation.

les systèmes d'argumentation définis à partir d'un ensemble (fini) d'arguments et nous supposons que tous les agents possèdent le même ensemble d'arguments. Notons \mathcal{N} l'ensemble de tous les agents. Une fonction d'agrégation γ est définie par $\gamma : \mathbb{AF}^n \rightarrow \mathbb{AF}$. Notons $\hat{F} = (F_1, \dots, F_n)$ un tuple de \mathbb{AF}^n . Sauf indication contraire, toutes les propriétés sont définies $\forall \hat{F} \in \mathbb{AF}^n$.

Anonymat. La fonction d'agrégation γ est **anonyme** si, pour toutes permutations $\Pi(\hat{AF})$ en entrée, la fonction retourne le même système d'argumentation.

$$(\mathbf{ANO}) \quad \forall \hat{F}' \in \Pi(\hat{F}) : \gamma(\hat{F}) = \gamma(\hat{F}')$$

Non-Trivialité. Un système d'argumentation est considéré comme non-trivial, pour une sémantique σ , s'il possède au moins une extension non-vidue : $|\mathcal{E}_\sigma(F)| \geq 1$ et $\mathcal{E}_\sigma(F) \neq \{\emptyset\}$. Notons \mathbb{AF}_{NT_σ} l'ensemble des systèmes d'argumentation non-triviaux pour une sémantique σ . La fonction d'agrégation γ est **σ -fortement non-triviale** si l'AF en sortie est toujours non-trivial :

$$(\sigma\text{-SNT}) \quad \gamma(\hat{F}) \in \mathbb{AF}_{NT_\sigma}$$

La fonction d'agrégation γ est **σ -faiblement non-triviale** si, quand tous les systèmes d'argumentation en entrée sont non-triviaux, alors l'AF en sortie est non-trivial :

$$(\sigma\text{-WNT}) \quad \forall \hat{F} \in \mathbb{AF}_{NT_\sigma}^n : \gamma(\hat{F}) \in \mathbb{AF}_{NT_\sigma}$$

Décisive. Un système d'argumentation est décisif, pour une sémantique σ , s'il possède exactement une extension non-vidue : $|\mathcal{E}_\sigma(F)| = 1$ et $\mathcal{E}_\sigma(F) \neq \{\emptyset\}$. Notons \mathbb{AF}_{D_σ} l'ensemble des systèmes d'argumentation décisifs pour une sémantique σ . La fonction d'agrégation γ est **σ -fortement décisive** si l'AF en sortie est toujours décisif :

$$(\sigma\text{-SD}) \quad \gamma(\hat{F}) \in \mathbb{AF}_{D_\sigma}$$

La fonction d'agrégation γ est **σ -faiblement décisive** si, quand tous les systèmes d'argumentation en entrée sont décisifs, alors l'AF en sortie est décisif :

$$(\sigma\text{-WD}) \quad \forall \hat{F} \in \mathbb{AF}_{D_\sigma}^n : \gamma(\hat{F}) \in \mathbb{AF}_{D_\sigma}$$

Unanimité. Cette propriété spécifique que si tous les agents sont unanimes sur un aspect du domaine (extensions, attaques, ...), pour une sémantique σ , alors cette unanimité doit être reflétée dans le système d'argumentation de sortie.

• **Attaque unanime** concerne les attaques :

$$(\mathbf{UA}) \quad \bigcap_{k=1}^n Att(F_k) \subseteq Att(\gamma(\hat{F}))$$

• **σ -unanime** concerne les extensions :

$$(\sigma\text{-U}) \quad \bigcap_{k=1}^n \mathcal{E}_\sigma(F_k) \subseteq \mathcal{E}_\sigma(\gamma(\hat{F}))$$

• **ca_σ -unanime** concerne l'inférence crédule :

$$(ca_\sigma\text{-U}) \quad \bigcap_{k=1}^n ca_\sigma(F_k) \subseteq ca_\sigma(\gamma(\hat{F}))$$

- **sa $_\sigma$ -unanime** concerne l'inférence sceptique :

$$(sa_\sigma\text{-U}) \quad \bigcap_{k=1}^n sa_\sigma(F_k) \subseteq sa_\sigma(\gamma(\hat{F}))$$

Majorité. Si une stricte majorité d'agents sont d'accord sur un aspect du domaine, alors cette majorité doit être reflétée dans le système d'argumentation de sortie.

- **Attaque majoritaire** concerne les attaques :
(MAJ-A)

$$(|\{F_i : a \in Att(F_i)\}| > \frac{n}{2}) \Rightarrow a \in Att(\gamma(\hat{F}))$$

- **σ -majoritaire** concerne les extensions :

(σ -MAJ)

$$(|\{F_i : S \in \mathcal{E}_\sigma(F_i)\}| > \frac{n}{2}) \Rightarrow S \in \mathcal{E}_\sigma(\gamma(\hat{F}))$$

- **ca $_\sigma$ -majoritaire** concerne l'inférence crédule :

(ca $_\sigma$ -MAJ)

$$(|\{F_i : x \in ca_\sigma(F_i)\}| > \frac{n}{2}) \Rightarrow x \in ca_\sigma(\gamma(\hat{F}))$$

- **sa $_\sigma$ -majoritaire** concerne l'inférence sceptique :

(sa $_\sigma$ -MAJ)

$$(|\{F_i : x \in sa_\sigma(F_i)\}| > \frac{n}{2}) \Rightarrow x \in sa_\sigma(\gamma(\hat{F}))$$

Fermeture. Cette propriété stipule que la fonction d'agrégation ne doit rien inventer, i.e. si elle produit un aspect du domaine quelconque en sortie, alors cet aspect doit être obligatoirement présent en entrée.

- **Fermeture** dit que le graphe de sortie doit correspondre exactement à un des graphes d'entrée :

$$(CLO) \quad \exists i \in \mathcal{N} : Att(\gamma(\hat{F})) = Att(F_i)$$

- **Attaque fermée** concerne les attaques :

$$(AC) \quad Att(\gamma(\hat{F})) \subseteq Att(F_1) \cup \dots \cup Att(F_n)$$

- **σ -fermeture** concerne les extensions :

$$(\sigma\text{-C}) \quad \forall S \in \mathcal{E}_\sigma(\gamma(\hat{F})) : S \in \bigcup_{k=1}^n \mathcal{E}_\sigma(F_k)$$

- **ca $_\sigma$ -fermeture** concerne l'inférence crédule :

$$(ca_\sigma\text{-C}) \quad \forall x \in ca_\sigma(\gamma(\hat{F})) : x \in \bigcup_{k=1}^n ca_\sigma(F_k)$$

- **sa $_\sigma$ -fermeture** concerne l'inférence sceptique :

$$(sa_\sigma\text{-C}) \quad \forall x \in sa_\sigma(\gamma(\hat{F})) : x \in \bigcup_{k=1}^n sa_\sigma(F_k)$$

Tohmé et al. [20] proposent également plusieurs propriétés inspirées de la théorie de choix social permettant la caractérisation des bons opérateurs d'agrégation. Ils proposent une propriété, appelée *Pareto condition*, qui correspond exactement à Attaque unanime (UA), et une propriété de non-dictateur, qui est satisfaite par tout opérateur d'agrégation raisonnable. Les deux autres propriétés sont les suivantes :

Réponse positive. Cette propriété dit que le fait d'augmenter le nombre d'agents en accord avec une attaque ne doit pas diminuer les chances que cette attaque apparaisse dans l'AF de sortie :

- (PR) Soient \hat{F} et \hat{F}' deux profils de \mathbb{AF}^n . Si $\{F_i \in$

$\hat{F} \mid (a, b) \in Att(F_i)\} \subseteq \{F'_i \in \hat{F}' \mid (a, b) \in Att(F'_i)\}$, et $(a, b) \in Att(\gamma(\hat{F}))$, alors $(a, b) \in Att(\gamma(\hat{F}'))$

Indépendance des alternatives non-pertinentes.

La décision qu'une attaque existe ou non en sortie doit uniquement prendre en compte les attaques entre ces deux arguments dans les AF en entrée :

(IIA) Soient \hat{F} et \hat{F}' deux profils de \mathbb{AF}^n . Si $(\forall i (a, b) \in Att(F_i) \text{ ssi } (a, b) \in Att(F'_i))$, alors $((a, b) \in Att(\gamma(\hat{F})) \text{ ssi } (a, b) \in Att(\gamma(\hat{F}')))$

Ce n'est pas mentionné dans [20] mais il est facile de montrer que PR implique IIA :

Proposition 1 *Réponse positive implique Indépendance des alternatives non-pertinentes.*

Une propriété basique nous semble manquante parmi celles déjà proposées : il s'agit d'une propriété d'identité. Si on a une unique entrée (éventuellement dupliquée un certain nombre de fois), le résultat doit être identique à l'entrée (nous ne contraignons pas le cas où l'entrée est triviale, car dans ce cas, il peut être intéressant de résoudre cette trivialité) :

- **Attaque Identité** concerne les attaques :

$$(A\text{-ID}) \quad Att(\gamma(F, \dots, F)) = Att(F)$$

- **σ -Identité** concerne les extensions :

$$(\sigma\text{-ID}) \quad \forall F \in \mathbb{AF}_{NT_\sigma} : \mathcal{E}_\sigma(\gamma(F, \dots, F)) = \mathcal{E}_\sigma(F)$$

- **ca $_\sigma$ -Identité** concerne l'inférence crédule :

$$(ca_\sigma\text{-ID}) \quad \forall F \in \mathbb{AF}_{NT_\sigma} : ca_\sigma(\gamma(F, \dots, F)) = ca_\sigma(F)$$

- **sa $_\sigma$ -Identité** concerne l'inférence sceptique :

$$(sa_\sigma\text{-ID}) \quad \forall F \in \mathbb{AF}_{NT_\sigma} : sa_\sigma(\gamma(F, \dots, F)) = sa_\sigma(F)$$

Ces quatre propriétés peuvent être vue comme un cas particulier de la propriété d'unanimité. Dans le cas où l'on fusionne un unique système d'argumentation (qui correspond aux croyances d'un agent) alors la propriété stipule que l'opérateur ne doit pas changer cet AF. Même si cette propriété semble assez intuitive, elle n'est cependant pas respectée par toutes les méthodes d'agrégation.

6 Propriétés des Opérateurs Existants

Commençons par vérifier quelles sont les propriétés satisfaites par les opérateurs d'agrégation de Coste-Marquis et al. [7]. Rappelons que les propriétés introduites par Dunne et al. sont définies pour un unique AF en sortie, alors que ces opérateurs peuvent en retourner plusieurs. Nous généralisons donc les propriétés de la section précédente comme suit : au lieu de déterminer si une propriété est satisfaite pour l'*unique* AF en sortie, on demande que cette propriété soit satisfaite pour *tous* les AF en sortie. En suivant ce raisonnement, il est possible de généraliser très simplement une grande partie des propriétés à l'exception de

la propriété IIA. Si on ajoute cela au fait que PR implique IIA, nous décidons de ne pas considérer cette propriété par la suite.

Proposition 2 Δ_{de}^{Σ} satisfait Anonymat (ANO), les propriétés d'Identité (A-ID, σ -ID, ca_{σ} -ID, sa_{σ} -ID), les propriétés d'Attaque unanime (UA), Attaque majoritaire (MAJ-A), Attaque fermée (AC) et Réponse positive (PR) pour toutes sémantiques $\sigma \in \{comp, pref, sta, gr\}$. Les autres propriétés ne sont pas satisfaites.

Vérifions maintenant s'il y a plus de propriétés satisfaites quand le leximax est utilisé comme fonction d'agrégation.

Proposition 3 $\Delta_{de}^{leximax}$ satisfait Anonymat (ANO), les propriétés d'Identité (A-ID, σ -ID, ca_{σ} -ID, sa_{σ} -ID), les propriétés d'Attaque unanime (UA), attaque fermée (AC) et Réponse positive (PR) pour toutes sémantiques $\sigma \in \{comp, pref, sta, gr\}$. Les autres propriétés ne sont pas satisfaites.

Enfin, regardons quelles propriétés sont satisfaites par la méthode *qualified voting* [20].

Proposition 4 *QV* satisfait *gr-faiblement non triviale* (*gr-WNT*), *gr-faiblement décisive* (*gr-WD*), *Attaque fermée* (AC) et *Réponse positive* (PR). Les autres propriétés ne sont pas satisfaites.

7 Utilisation des WAF pour l'Agrégation d'AF

Dans cette section, nous proposons trois nouvelles méthodes d'agrégation basées sur les WAF. Quand les WAF ont été introduit [13, 8, 9] une des interprétations possibles concernant les poids sur les attaques est qu'ils peuvent représenter le nombre d'agents en accord avec cette attaque. Nous utilisons donc cette interprétation afin de définir des opérateurs d'agrégation d'AF tout en utilisant les techniques développées pour les WAF en section 3.

7.1 FUS_{All}

Un première méthode, notée *FUS_{All}*, consiste à prendre en entrée un ensemble de système d'argumentation de Dung (représentant les croyances de chaque agent), afin de les fusionner pour obtenir un Weighted Argumentation Framework (WAF). Puis, à partir de ce WAF, nous appliquons une des quatre méthodes *best* (voir section 3.2) afin d'obtenir, en sortie, un ensemble d'extensions représentant le résultat de cette fusion.

Définition 12

$$FUS_{All}^{best_i^{\sigma, \oplus}}(\hat{AF}) = best_i^{\sigma, \oplus}(waf(\hat{AF}))$$

où $waf(\hat{AF}) = \langle A, R, w \rangle$, avec :

- $A = \mathcal{X}$,
- $R = \bigcup_{i=1}^n R_i$,
- et $w(a, b) = |\{AF_i \in \hat{AF} \mid (a, b) \in R_i\}|$.

Notons que la construction du *waf*(\hat{AF}) est exactement la même que celle proposée par Cayrol et Lagasque-Schiex [6] si on normalise le poids sur les attaques. Cependant, dans [6], rien n'est dit à propos de l'utilisation possible de ce WAF. Nous proposons d'utiliser les méthodes *best* afin de trouver les meilleurs extensions possibles.

Vérifions maintenant quelles propriétés sont satisfaites par *FUS_{All}*. Il est important de noter que ces opérateurs retournent un ensemble d'extensions. Par conséquent, certaines propriétés, basées sur les attaques, telles que Attaque unanime (UA), Attaque majoritaire (MAJ-A), Fermeture (CLO), Attaque fermée (AC), Attaque Identité (ID-A) et Réponse positive (PR) ne sont pas applicables dans ce cadre. Nous rappelons également que nous nous focalisons sur les sémantiques principales définies par Dung : $\sigma \in \{comp, pref, sta, gr\}$. Enfin, concernant les méthodes *best*, nous choisissons d'étudier les quatre méthodes proposées utilisant la somme et le max comme fonction d'agrégation ($\oplus \in \{\Sigma, max\}$).

Proposition 5 Soient $\sigma \in \{comp, pref, sta, gr\}$ une sémantique et $\oplus \in \{\Sigma, max\}$ une fonction d'agrégation. *FUS_{All}* satisfait Anonymat (ANO) et les propriétés *gr-Identité* (*gr-ID*), *ca_{gr}-Identité* (*ca_{gr}-ID*) et *sa_{gr}-Identité* (*sa_{gr}-ID*) quelle que soit la méthode *best* utilisée. Les autres propriétés ne sont pas satisfaites.

Une des propriétés non satisfaites est la propriété de non-trivialité, mettant en avant le fait que *FUS_{All}* ne garantit pas toujours de retourner un résultat non-trivial, même si les AF en entrée sont tous non-triviaux.

7.2 FUS_{AllNT}

Une solution permettant de satisfaire la non-trivialité, et donc d'assurer l'obtention d'un ensemble d'extensions de \overline{WF} non-nul, est d'utiliser les *relaxing extensions* [9].

Définition 13

$$FUS_{AllNT}^{\sigma, best_i^{\oplus, \otimes}}(\hat{AF}) = best_i^{\oplus}(\mathcal{E}_{\sigma}^{\otimes}(\overline{waf(\hat{AF})}, waf(\hat{AF})))$$

où $waf(\hat{AF}) = \langle A, R, w \rangle$, avec :

- $A = \mathcal{X}$,
- $R = \bigcup_{i=1}^n R_i$,
- et $w : w(a, b) = |\{AF_i \in \hat{AF} \mid (a, b) \in R_i\}|$.

Concernant la fonction d'agrégation utilisée pour les *relaxing extensions*, nous nous focalisons uniquement sur la somme ($\otimes = \Sigma$).

Proposition 6 Soient $\sigma \in \{comp, pref, sta, gr\}$ une sémantique, $\oplus \in \{\Sigma, max\}$ et $\otimes = \Sigma$ deux fonctions d'agrégation. FUS_{AllNT} satisfait Anonymat (ANON), σ -fortement non-triviale (σ -SNT), σ -faiblement non-triviale (σ -WNT) ainsi que les propriétés gr-Identité (gr-ID), ca_{gr} -Identité (ca_{gr} -ID) et sa_{gr} -Identité (sa_{gr} -ID) quelle que soit la méthode best utilisée. Les autres propriétés ne sont pas satisfaites.

Cet opérateur est intéressant dans le cas où l'on veut prendre en considération toutes les attaques données par les agents. Cependant, le résultat ne représente pas toujours l'opinion de la majorité. Par exemple, supposons qu'en entrée nous ayons neuf AF avec $A = \{a, b\}$ et $R = \{\}$ et un AF avec les mêmes arguments mais $R = \{(a, b)\}$. Si on fusionne ces dix AF en utilisant un des deux opérateurs d'agrégation proposés précédemment, alors l'attaque (a, b) , proposée seulement par un agent, sera présente dans le système résultant. Cela va clairement à l'encontre de l'opinion exprimée par la majorité.

7.3 FUS_{MajNT}

Afin de contrer ce problème d'agent « isolé » (ou d'une minorité d'agents), une solution possible est d'utiliser la majorité lors de la construction du *weighted argumentation framework*. En effet, au lieu de représenter, dans le WAF, toutes les attaques ayant au moins un agent en accord avec elle, nous sélectionnons les attaques étant acceptées par une majorité stricte d'agents.

Définition 14

$$FUS_{MajNT}^{\sigma, best_i^{\oplus}, \otimes}(\hat{AF}) = best_i^{\oplus}(\mathcal{E}_{\sigma}^{\otimes}(\overline{mwf(\hat{AF})}, mwf(\hat{AF})))$$

où $mwf(\hat{AF}) = \langle A, R, w \rangle$, avec :

- $A = \mathcal{X}$,
- $R = \{(a, b) \mid |\{AF_i \mid (a, b) \in Att(AF_i)\}| > \frac{n}{2}\}$,
- et $w(a, b) = |\{AF_i \in \hat{AF} \mid (a, b) \in R_i\}|$ si $(a, b) \in R$, et $= 0$ sinon.

De la même façon que pour les autres méthodes, vérifions quelles sont les propriétés satisfaites par la méthode FUS_{MajNT} .

Proposition 7 Soient $\sigma \in \{comp, pref, sta, gr\}$ une sémantique, $\oplus \in \{\Sigma, max\}$ et $\otimes = \Sigma$ deux fonctions d'agrégation. FUS_{MajNT} satisfait Anonymat (ANON), σ -fortement non-triviale (σ -SNT), σ -faiblement non-triviale (σ -WNT) et les propriétés gr-Identité (gr-ID), ca_{gr} -Identité (ca_{gr} -ID) et sa_{gr} -Identité (sa_{gr} -ID) quelle que soit la méthode best utilisée. Les autres propriétés ne sont pas satisfaites.

Il est important de noter que malgré le fait d'utiliser la majorité lors de la construction du WAF, les propriétés de majorité ne sont pas satisfaites par FUS_{MajNT} .

8 Discussion

La Table 1 résume les propriétés satisfaites par toutes les méthodes d'agrégation étudiées dans cet article. Une croix \times signifie que la propriété n'est pas satisfaite, le symbole \checkmark signifie que la propriété est satisfaite pour toutes les sémantiques (gr, comp, sta et pref), \checkmark^{σ} signifie que la propriété est satisfaite par la sémantique σ , et le symbole $-$ signifie que la propriété n'est pas applicable à l'opérateur (car le type de retour de l'opérateur n'est pas compatible avec la contrainte établie par la règle).

Propriétés	Δ_{de}^{Σ}	$\Delta_{de}^{leximax}$	QV	FUS_{All}	FUS_{AllNT}	FUS_{MajNT}
ANON	\checkmark	\checkmark	\times	\checkmark	\checkmark	\checkmark
σ -SNT	\times	\times	\times	\times	\checkmark	\checkmark
σ -WNT	\times	\times	\checkmark^{gr}	\times	\checkmark	\checkmark
σ -SD	\times	\times	\times	\times	\times	\times
σ -WD	\times	\times	\checkmark^{gr}	\times	\times	\times
UA	\checkmark	\checkmark	\times	-	-	-
σ -U	\times	\times	\times	\times	\times	\times
ca_{σ} -U	\times	\times	\times	\times	\times	\times
sa_{σ} -U	\times	\times	\times	\times	\times	\times
MAJ-A	\checkmark	\times	\times	-	-	-
σ -MAJ	\times	\times	\times	\times	\times	\times
ca_{σ} -MAJ	\times	\times	\times	\times	\times	\times
sa_{σ} -MAJ	\times	\times	\times	\times	\times	\times
CLO	\times	\times	\times	-	-	-
AC	\checkmark	\checkmark	\checkmark	-	-	-
σ -C	\times	\times	\times	\times	\times	\times
ca_{σ} -C	\times	\times	\times	\times	\times	\times
sa_{σ} -C	\times	\times	\times	\times	\times	\times
A-ID	\checkmark	\checkmark	\times	-	-	-
σ -ID	\checkmark	\checkmark	\times	\checkmark^{gr}	\checkmark^{gr}	\checkmark^{gr}
ca_{σ} -ID	\checkmark	\checkmark	\times	\checkmark^{gr}	\checkmark^{gr}	\checkmark^{gr}
sa_{σ} -ID	\checkmark	\checkmark	\times	\checkmark^{gr}	\checkmark^{gr}	\checkmark^{gr}
PR	\checkmark	\checkmark	\checkmark	-	-	-

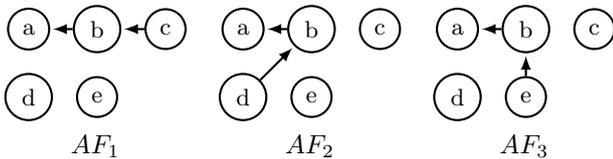
TABLE 1 – Propriétés satisfaites par les opérateurs d'agrégation étudiés

Il est clair que peu de propriétés sont satisfaites par les opérateurs d'agrégation existants. Deux explications (non-exclusives) sont possibles : soit les opérateurs existants ne sont pas satisfaisant du côté des

propriétés de rationalité, soit les propriétés sont trop fortes. Nous sommes d’avis que les deux sont vraies dans une certaine mesure; cela signifie qu’une étude plus approfondie est nécessaire à la fois dans la définition d’un ensemble de propriétés rationnelles qui capture plus adéquatement le comportement souhaitable d’un opérateur d’agrégation, mais que l’on doit également tenter de définir de meilleurs opérateurs d’agrégation de systèmes d’argumentation.

Commençons par argumenter sur le fait que certaines propriétés sont trop fortes. Pour cela, nous utilisons un contre-exemple simple et intuitif qui contredit pas moins de neuf propriétés.

Exemple 1 Soient AF_1 , AF_2 et AF_3 trois systèmes d’argumentation, représentés sur la Figure 1, possédant tous la même (unique) extension complète/préférée/stable/de base : $\{a, c, d, e\}$.



Résultat de l’agrégation souhaité :

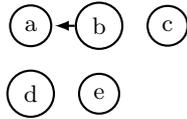


FIGURE 1 – Exemple contredisant les propriétés de majorité, d’unanimité et de fermeture

Ce résultat peut paraître, à première vue, illogique si on regarde les extensions obtenues (et de ce fait les arguments acceptés). En effet, chaque AF_i possédant la même extension $\{a, c, d, e\}$, on pourrait s’attendre à voir cette extension comme étant le résultat de l’agrégation. Cependant, en regardant plus attentivement cet exemple, on remarque que cette même extension est obtenue pour des raisons différentes. Chaque agent possède une raison (argument) pour rejeter b , mais cette attaque est contestée par tous les autres agents, ce qui peut être considéré comme une erreur pour cet agent. Il semble donc plus naturel de rejeter toutes les attaques de b du résultat de l’agrégation, ce qui a pour conséquence d’obtenir l’extension $\{b, c, d, e\}$ à la place de $\{a, c, d, e\}$. De plus, ce résultat est obtenu en utilisant la méthode majoritaire (vote majoritaire sur la relation d’attaque) : tous les AF_i sont d’accord sur le fait que b attaque a , alors que toutes les autres attaques ne possèdent au maximum qu’un agent qui les supporte. C’est notamment le résultat obtenu en utilisant l’opérateur FUS_{MAJNT} . Néanmoins, ce raisonnement va à

l’encontre des propriétés d’unanimité, de majorité et de fermeture liées aux extensions et aux arguments acceptés sceptiquement/crédulement⁴. Donc si le résultat proposé dans l’exemple est jugé désirable, les seules propriétés qui ne semblent pas problématiques sont celles basées sur les relations d’attaque entre arguments.

Nous voulons insister sur le fait que, sur cet exemple, uniquement trois systèmes d’argumentation sont utilisés, mais il est possible de le généraliser à 100 agents (et 102 arguments) soutenant tous que b attaque a , et, pour chaque agent i , il existe une seule attaque supplémentaire entre un argument a_i et b (qu’il est le seul à posséder). Dans ce cas, la notion de quasi-unanimité (tous les agents sauf un sont contre les autres attaques) est beaucoup plus flagrante.

Les propriétés décisives (σ -WD et σ -SD) semblent également trop fortes car elles vont à l’encontre de la plupart des sémantiques retournant un ensemble d’extensions (et sont triviales pour celles qui acceptent au plus une extension), c’est pourquoi nous proposons de ne pas les inclure dans la liste des propriétés de base.

Fondamentalement, nous pensons que les propriétés proposées sont plus ou moins des traductions directes des propriétés provenant de la théorie de choix social. Cela représente en soi une première étape importante. Cependant, les systèmes d’argumentation possèdent une structure plus importante que les ensembles de candidats utilisés dans les problèmes de vote, et cette structure doit être prise en considération. Nous soutenons que ces spécificités structurales invalident la plupart des propriétés provenant de la théorie de choix social. Cela ne veut pas dire qu’elles sont sans intérêt, puisqu’elles peuvent être utilisées afin de caractériser certaines méthodes d’agrégation (il est possible que certaines méthodes les satisfassent), cependant elles ne peuvent être considérées comme absolument nécessaires.

Les lignes grisées dans la Table 1 contiennent les propriétés que nous jugeons indispensables lorsque l’on se concentre sur la relation d’attaque. Notons également qu’aucune des méthodes existantes ne satisfait pleinement l’ensemble des propriétés souhaitées. Cela signifie que la définition de bons opérateurs d’agrégation est encore nécessaire. De plus, l’exemple 1 permet de mettre en avant certaines incompatibilités entre les propriétés de rationalité d’agrégation de systèmes d’argumentation basées sur les extensions et celles basées sur les attaques. Les deux approches étant raisonnables, la définition de deux ensembles de postulats, l’un basé sur les extensions et l’autre sur les attaques, semble plus appropriée.

4. σ -MAJ, ca_σ -MAJ, sa_σ -MAJ, σ -U, ca_σ -U, sa_σ -U, CLO, σ -C, ca_σ -C, and sa_σ -C

9 Conclusion

Dans cet article, nous avons d’abord décrit les travaux existants où les auteurs proposent des méthodes d’agrégation de systèmes d’argumentation abstraits de Dung. Nous nous sommes focalisés sur les méthodes prenant en entrée un ensemble de systèmes d’argumentation abstraits et retournent un système d’argumentation, un ensemble de systèmes d’argumentation ou un ensemble d’extensions.

Nous avons également étudié comment utiliser les *weighted argumentation frameworks* (WAF) afin d’agrèger un ensemble de systèmes d’argumentation, et proposons trois définitions possibles dont FUS_{MajNT} semble la plus convaincante.

Nous montrons qu’il existe peu de propriétés satisfaites par les opérateurs d’agrégation existants. L’explication semble incriminer deux suspects : les propriétés et les opérateurs. D’un côté, certaines propriétés semblent trop fortes dans le cas général. De l’autre, les opérateurs existants ne satisfont pas les propriétés jugées désirables.

Nos résultats semblent indiquer que beaucoup d’efforts sont encore nécessaires sur les deux fronts. Que ce soit sur une étude plus approfondie des propriétés de rationalité pour les méthodes d’agrégation pour l’argumentation abstraite, que sur la définition d’autres (meilleures ?) méthodes d’agrégation.

Nous prévoyons d’étudier si l’utilisation des propriétés provenant de la fusion en logique propositionnelle [16] pourrait être plus appropriée. En effet, ces propriétés ont été définies pour un cadre (la logique propositionnelle) qui dispose d’une certaine structure. Nous travaillons donc sur la traduction de ces propriétés afin de les adapter au cadre des systèmes d’argumentation, de façon similaire à ce qui a été fait récemment pour la révision des croyances [10], afin de définir si les propriétés de rationalité peuvent être jugées plus adéquates, et ainsi obtenir quelques idées visant à définir de nouvelles méthodes d’agrégation.

Références

- [1] Leila Amgoud and Srdjan Vesic. On revising argumentation-based decision systems. In *10th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU’09)*, pages 71–82. Springer, 2009.
- [2] Ringo Baumann. Normal and strong expansion equivalence for argumentation frameworks. *Artificial Intelligence*, 193 :18–44, 2012.
- [3] Guido Boella, Souhila Kaci, and Leendert van der Torre. Dynamics in argumentation with single extensions : Abstraction principles and the grounded extension. In *10th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU’09)*, pages 107–118. Springer, 2009.
- [4] Elise Bonzon and Nicolas Maudet. On the outcomes of multiparty persuasion. In *10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS’11)*, pages 47–54, 2011.
- [5] Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasque-Schiex. Revision of an argumentation system. In *11th International Conference on the Principles of Knowledge Representation and Reasoning (KR’08)*, pages 124–134, 2008.
- [6] Claudette Cayrol and Marie-Christine Lagasque-Schiex. Weighted argumentation systems : A tool for merging argumentation systems. In *IEEE 23rd International Conference on Tools with Artificial Intelligence (ICTAI’11)*, pages 629–632, 2011.
- [7] Sylvie Coste-Marquis, Caroline Devred, Sébastien Konieczny, Marie-Christine Lagasque-Schiex, and Pierre Marquis. On the merging of Dung’s argumentation systems. *Artificial Intelligence*, 171(10-15) :730–753, 2007.
- [8] Sylvie Coste-Marquis, Sébastien Konieczny, Pierre Marquis, and Mohand-Akli Ouali. Selecting extensions in weighted argumentation frameworks. In *4th International Conference on Computational Models of Argument (COMMA’12)*, pages 342–349, 2012.
- [9] Sylvie Coste-Marquis, Sébastien Konieczny, Pierre Marquis, and Mohand-Akli Ouali. Weighted attacks in argumentation frameworks. In *13th International Conference on the Principles of Knowledge Representation and Reasoning (KR’12)*, pages 593–597, 2012.
- [10] Sylvie Coste-Marquis, Sébastien Konieczny, Jean-Guy Mailly, and Pierre Marquis. On the revision of argumentation systems : Minimal change of arguments statuses. In *14th International Conference on the Principles of Knowledge Representation and Reasoning (KR’14)*, pages 52–61, 2014.
- [11] Céilia da Costa Pereira, Andrea Tettamanzi, and Serena Villata. Changing one’s mind : Erase or rewind ? In *22nd International Joint Conference on Artificial Intelligence (IJCAI’11)*, pages 164–171, 2011.
- [12] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2) :321–358, 1995.

- [13] Paul E. Dunne, Anthony Hunter, Peter McBurney, Simon Parsons, and Michael Wooldridge. Weighted argument systems : Basic definitions, algorithms, and complexity results. *Artificial Intelligence*, 175(2) :457–486, 2011.
- [14] Paul E. Dunne, Pierre Marquis, and Michael Wooldridge. Argument aggregation : Basic axioms and complexity results. In *4th International Conference on Computational Models of Argument (COMMA'12)*, pages 129–140, 2012.
- [15] Dov M. Gabbay and Odinaldo Rodrigues. A numerical approach to the merging of argumentation networks. In *13th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA'12)*, pages 195–212, 2012.
- [16] Sébastien Konieczny and Ramón Pino Pérez. Merging information under constraints : a logical framework. *Journal of Logic and Computation*, 12(5) :773–808, 2002.
- [17] Gerald H. Kramer. A dynamical model of political equilibrium. *Journal of Economic Theory*, 16(2) :310–334, December 1977.
- [18] Donald G. Saari and Vincent R. Merlin. The copeland method 1 ; relationships and the dictionary. *Economic Theory*, 8, 1996.
- [19] Paul B Simpson. On Defining Areas of Voter Choice : Professor Tullock on Stable Voting. *The Quarterly Journal of Economics*, 83(3) :478–90, August 1969.
- [20] Fernando A. Tohmé, Gustavo Adrian Bodanza, and Guillermo Ricardo Simari. Aggregation of attack relations : A social-choice theoretical analysis of defeasibility criteria. In *5th International Symposium on Foundations of Information and Knowledge Systems (FoIKS'08)*, pages 8–23, 2008.

Le cube des oppositions - Une structure à la base de nombreux formalismes de représentation des connaissances

Didier Dubois¹ Henri Prade¹ Agnès Rico²

1. IRIT- Institut de Recherche en Informatique de Toulouse,
Université Paul Sabatier, 31062 Toulouse Cedex 9
2. ÉRIC- Equipe de Recherche en Ingénierie des Connaissances,
Université Claude Bernard Lyon 1, 69100 Villeurbanne
{dubois, prade}@irit.fr, agnes.rico@univ-lyon1.fr

Résumé

Le carré des oppositions est une structure inventée du temps d'Aristote, qui met en jeu deux négations involutives en reliant des énoncés quantifiés. Redécouvert dans la seconde moitié du $XX^{\text{ème}}$ siècle pour son intérêt dans la modélisation des structures conceptuelles et le traitement de problèmes en logiques paraconsistantes, le carré des oppositions a été récemment complété en un cube, ce qui correspond à l'introduction d'une troisième négation. Un tel cube se rencontre dans des cadres très différents de représentation des connaissances, comme la logique modale, la théorie des possibilités dans sa version tout-ou-rien, l'analyse formelle de concepts, la théorie des ensembles approximatifs, ou l'argumentation abstraite. Après avoir rappelé ces résultats dans une perspective unifiée, l'article propose une extension graduelle du cube et montre comment plusieurs formalismes, qualitatifs aussi bien que quantitatifs, comme l'intégrale de Sugeno utilisée en agrégation multi-critères et en décision qualitative, ou encore les fonctions de croyance et les intégrales de Choquet, obéissent à des transformations qui forment des cubes d'oppositions graduelles. Cette découverte introduit une nouvelle perspective pour de nombreux formalismes de représentation des connaissances, soulignant leurs traits communs. Le cube des oppositions met en évidence de fructueux parallélismes entre différents formalismes, ce qui peut conduire à mettre en lumière des composants présents dans un formalisme et encore manquants ou négligés dans un autre.

Abstract

The square of opposition is a structure involving two involutive negations and relating quantified statements, invented in Aristotle time. Rediscovered in the second

half of the XX^{th} century, and advocated as being of interest for understanding conceptual structures and solving problems in paraconsistent logics, the square of opposition has been recently completed into a cube, which corresponds to the introduction of a third negation. Such a cube can be encountered in very different knowledge representation formalisms, such as modal logic, possibility theory in its all-or-nothing version, formal concept analysis, rough set theory and abstract argumentation. After restating these results in a unified perspective, the paper proposes a graded extension of the cube and shows that several qualitative, as well as quantitative formalisms, such as Sugeno integrals used in multiple criteria aggregation and qualitative decision theory, or yet belief functions and Choquet integrals, are amenable to transformations that form graded cubes of opposition. This discovery leads to a new perspective on many knowledge representation formalisms, laying their underlying common features. The cube of opposition exhibits fruitful parallelisms between different formalisms, which leads to highlight some missing components present in one formalism and currently absent from another.

1 Introduction

On peut considérer que la première tentative de modélisation du raisonnement humain remonte en Occident à l'étude des syllogismes, qui débuta dans l'Antiquité Grecque, et qui fut poursuivie tout au long des siècles, jusqu'à Euler [20] qui en donna une représentation diagrammatique, et à Gergonne [23] [21] qui fut le premier à établir quels syllogismes étaient valides, et lesquels ne l'étaient pas, et ce sur une base

rigoureuse [26]. C'était donc bien avant les débuts de l'intelligence artificielle (IA) qu'on s'est intéressé à la formalisation du raisonnement. En même temps que les syllogismes, Aristote et son école introduisit aussi le carré des oppositions [27], un schéma qui présente différentes formes d'opposition entre des énoncés universellement ou existentiellement quantifiés qui peuvent apparaître comme prémisses dans des syllogismes. Les oppositions dans le carré résultent de l'interaction entre une négation "externe" et une négation "interne", toutes deux involutives. L'intérêt pour ce carré semble disparaître avec les débuts de la logique moderne à la fin du XIX^{ème} siècle. Un regain d'intérêt pour les structures d'opposition voit le jour dans les années 1950's quand un logicien français, Robert Blanché [6] découvre que le carré peut être complété en un hexagone contenant trois carrés d'opposition, hexagone qu'on retrouve dans l'organisation de nombreuses structures conceptuelles, comme par exemple, les comparateurs mathématiques, ou les modalités déontiques [7]. Cet intérêt se confirmera plus tard, quand les structures du carré et de l'hexagone s'avèrent utiles pour résoudre des questions délicates, en particulier en logique modale paraconsistante [4, 5].

Plus récemment, il a été mis en évidence qu'une extension cubique particulière du carré des oppositions, mettant en jeu une troisième négation, se rencontre dans différents formalismes de représentation des connaissances utilisés en IA, à savoir la logique modale, la théorie des possibilités dans sa version tout-ou-rien, l'analyse formelle de concepts, la théorie des ensembles approximatifs ("rough sets") et l'argumentation abstraite [15, 2, 9]. Cet état de fait est d'autant plus remarquable que ces formalismes ont été développés indépendamment les uns des autres, et en général avec des objectifs très différents en représentation des connaissances. Les bénéfices escomptés de la découverte de telles analogies entre des formalismes ayant des motivations différentes, sont de deux ordres. La découverte que cette structure cubique d'opposition est à l'œuvre dans un formalisme peut apporter un autre éclairage sur sa compréhension, et ce qui est plus important peut conduire à mettre en évidence de nouveaux éléments, jusqu'alors négligés dans ce formalisme, alors que leur contrepartie dans un autre formalisme est bien connue et joue un rôle important.

Le carré et par suite le cube des oppositions sont des structures où les sommets sont traditionnellement associés à des énoncés qui sont vrais ou faux, ou qui comportent des modalités binaires. Dans la suite on montre que cela fait sens d'étendre le carré et le cube à des structures graduelles. On peut alors envisager d'étudier d'autres formalismes à la lumière du cube des oppositions, tels que la théorie des possibilités (gra-

duelles), les opérations qualitatives d'agrégation multi-critère, et plus généralement les intégrales de Sugeno, ou des notions plus quantitatives comme les fonctions de croyance et l'intégrale de Choquet.

L'article est structuré comme suit. La Section 2 présente un rappel sur le carré, l'hexagone, et le cube des oppositions. En Section 3, on propose une lecture très simple du cube en termes d'indicateurs ensemblistes, qui s'avère très proche de la théorie des possibilités booléennes, et on montre la complémentarité des différentes informations apparaissant sur le cube. En Section 4, une autre lecture essentielle, en termes de composition relationnelle est rappelée, qui explique pourquoi la logique modale, l'analyse formelle de concepts, les "rough sets", et l'argumentation abstraite ont pour soubassement la structure du cube des oppositions. La Section 5 discute comment le cube se généralise naturellement à des structures graduelles. Ceci est exemplifié en Section 6 avec l'agrégation multi-critère et les intégrales de Sugeno, et en Section 7 avec les fonctions de croyance et les intégrales de Choquet.

2 Carré et cube

Le carré des oppositions traditionnel [27] est construit à partir d'énoncés universellement et existentiellement quantifiés, de la manière suivante. Considérons l'énoncé (**A**) de la forme "tous les P sont des Q ", dont la négation est l'énoncé (**O**) "au moins un P n'est pas un Q ", ainsi que l'énoncé (**E**) "aucun P n'est un Q ", qui est clairement en une opposition plus forte avec le premier énoncé (**A**). Ces trois énoncés, complétés par la négation du dernier énoncé (**E**), c'est-à-dire (**I**) "au moins un P est un Q ", peuvent être disposés sur un carré dont les sommets sont traditionnellement dénotés par les lettres **A**, **I** (moitié affirmative) et **E**, **O** (moitié négative), comme montré sur la Figure 1 (où \bar{Q} signifie "non Q ").

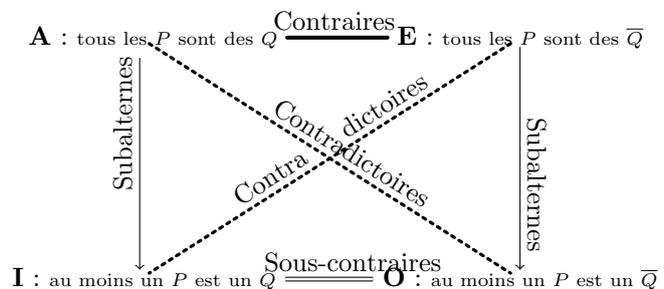


FIGURE 1 – Carré des oppositions

Comme on peut le vérifier, des relations remarquables tiennent dans le carré :

- (i) **A** et **O** (resp. **E** et **I**) sont les négations de l'autre élément de la paire;
- (ii) **A** implique **I**, et **E** implique **O** (on suppose qu'il y a au moins un P afin d'éviter les problèmes d'"import existentiel");
- (iii) **A** et **E** ne peuvent être vrais ensemble, mais peuvent éventuellement être faux ensemble;
- (iv) **I** et **O** ne peuvent être faux ensemble, mais peuvent éventuellement être vrais ensemble.

Blanché [6, 7] remarqua que si on ajoutait deux autres sommets **U** et **Y** définis respectivement comme la disjonction de **A** et **E**, et comme la conjonction de **I** et **O**, on obtient un hexagone **AUEOYI** qui comprend 3 carrés d'opposition, **AEIOI**, **YAUO**, **YEUI** chacun satisfaisant les quatre types de relation ci-dessus. Un tel hexagone est obtenu chaque fois qu'on part de 3 situations mutuellement exclusives, telles que **A**, **E**, et **Y** [15]. Ces hexagones présentent un intérêt particulier pour l'argumentation abstraite, les "rough sets" et l'analyse formelle de concepts [1, 9] notamment. Cependant, nous les laissons en dehors du cadre de cette étude, pour se concentrer sur une autre extension du carré, à savoir le cube des oppositions.

En utilisant les notations de la logique du premier ordre, et en changeant P en $\neg P$, et Q en $\neg Q$ on obtient un autre carré des oppositions **aeoi** similaire, où on suppose aussi que l'ensemble des "non- P " est non-vide. Les 8 énoncés, **A**, **I**, **E**, **O**, **a**, **i**, **e**, **o** peuvent être alors organisés en ce qui peut être appelé un *cube des oppositions* [15] comme en Figure 2. La face de devant et la face de derrière du cube sont des carrés des oppositions au sens traditionnel, où les segments épais sans flèche relient des contraires, les segments à lignes doubles fines, sans flèche, des sous-contraires, les diagonales en pointillés sans flèche des contradictoires, et les lignes verticales avec une flèche vers le bas pointent vers des subalternes, et expriment des implications.

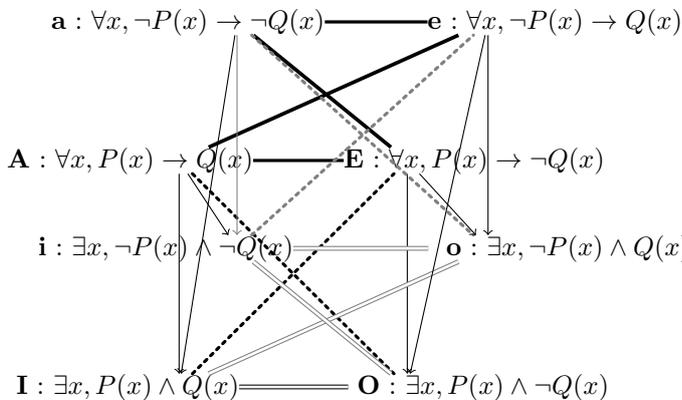


FIGURE 2 – Cube des oppositions des énoncés quantifiés

On suppose non seulement qu'il y a au moins un P et au moins un non- P , mais aussi qu'il y a au moins un Q et au moins un non- Q . On peut vérifier alors que **A** implique **i**, **a** implique **I**, **e** implique **O**, et **E** implique **o**. On peut noter aussi que les sommets **a** et **E**, ainsi que **A** et **e** ne peuvent pas être vrais ensemble (e.g., en effet avoir à la fois **A** et **e** vrais contredirait que $\exists x, \neg Q(x)$), tandis que les sommets **i** et **O**, ainsi que **I** et **o** ne peuvent pas être faux ensemble. Enfin, notons qu'il n'y a aucun lien logique entre **A** et **a**, **E** et **e**, **I** et **i**, ou **O** et **o**.

De manière assez remarquable, on peut vérifier que le groupe de transformations de Piaget I, N, R, C [29] est à l'œuvre dans les plans diagonaux entre les deux faces latérales du cube de la Figure 2. Les transformations de ce groupe s'appliquent à tout énoncé $\Phi(p, q, \dots)$ et ont pour résultats $N(\Phi(p, q, \dots)) = \neg\Phi(p, q, \dots)$, $R(\Phi(p, q, \dots)) = \Phi(\neg p, \neg q, \dots)$, $C(\Phi(p, q, \dots)) = \neg\Phi(\neg p, \neg q, \dots)$, où p, q dénotent des littéraux, et I est l'identité. En effet, on a $\mathbf{a} = R(\mathbf{A})$, $\mathbf{o} = N(\mathbf{a})$, $\mathbf{o} = C(\mathbf{a})$, et $N \circ R \circ C = I$. Dans ces transformations, deux négations involutives interviennent, une externe N , et une interne R . Dans le carré initial, la négation "interne" s'applique seulement à Q , tandis qu'une seconde négation "interne" s'applique seulement à P , quand on passe du carré au cube. Cela devient encore plus clair avec l'interprétation relationnelle du cube présentée dans la section suivante. Il est clair que la propriété d'involution de la négation est cruciale pour obtenir les propriétés du cube (ainsi que la propriété de contraposition de l'implication).

3 Le cube relationnel

Il a été récemment remarqué [9] que toute relation binaire R sur un produit cartésien $X \times Y$ (on peut avoir $Y = X$), composée avec un sous-ensemble, donnait naissance à un cube des oppositions. On suppose $R \neq \emptyset$. Soit $xR = \{y \in Y \mid (x, y) \in R\}$. \bar{R} dénote la relation complémentaire ($x\bar{R}y$ ssi $(x, y) \notin R$), et R^t la relation transposée ($xR^t y$ si et seulement si yRx); yR^t est aussi dénoté de façon équivalente $Ry = \{x \in X \mid (x, y) \in R\}$. De plus, on suppose que $\forall x, xR \neq \emptyset$, ce qui signifie que la relation R est *sérielle*, c'est-à-dire $\forall x, \exists y$ t. q. $(x, y) \in R$. De manière similaire R^t est supposée sérielle, i.e., $\forall y, Ry \neq \emptyset$, de même pour sa \bar{R} et sa transposée, on suppose $\forall x, xR \neq Y$ et $\forall y, Ry \neq X$.

Soit T un sous-ensemble de Y et \bar{T} son complément. On suppose $T \neq \emptyset$ et $T \neq Y$. la composition est définie de la façon usuelle : $R(T) = \{x \in X \mid \exists t \in T, (x, t) \in R\}$. A partir de la relation R et du sous-ensemble T , on peut définir les quatre sous-ensembles suivants de

X (et leurs compléments) :

$$R(T) = \{x \in X \mid T \cap xR \neq \emptyset\} \quad (1)$$

$$\overline{R(\overline{T})} = \{x \in X \mid xR \subseteq T\} \quad (2)$$

$$\overline{\overline{R(T)}} = \{x \in X \mid T \subseteq xR\} \quad (3)$$

$$\overline{R(\overline{T})} = \{x \in X \mid T \cup xR \neq X\} \quad (4)$$

Ces quatre sous-ensembles et leurs compléments forment un cube des oppositions remarquable (Fig.3). On peut vérifier que les contreparties ensemblistes des relations existant entre les énoncés logiques des sommets du cube de la Fig. 2 sont bien satisfaites :

i) les diagonales des faces de devant et de derrière lient des compléments ;

ii) $\overline{R(\overline{T})} \subseteq R(T)$, $\overline{R(\overline{T})} \subseteq \overline{R(\overline{T})}$, $\overline{\overline{R(T)}} \subseteq \overline{R(T)}$, et $\overline{\overline{R(T)}} \subseteq \overline{\overline{R(T)}}$ de même que $\overline{R(\overline{T})} \subseteq \overline{\overline{R(T)}}$, $\overline{\overline{R(T)}} \subseteq \overline{R(\overline{T})}$, et $\overline{\overline{R(T)}} \subseteq \overline{R(T)}$ grâce aux hypothèses de sérialité ; ces inclusions sont figurées par des flèches sur la Fig. 3 ;

iii) $\overline{R(\overline{T})} \cap \overline{\overline{R(T)}} = \emptyset$, $\overline{R(\overline{T})} \cap \overline{\overline{R(T)}} = \emptyset$, $\overline{\overline{R(T)}} \cap \overline{R(\overline{T})} = \emptyset$, et $\overline{\overline{R(T)}} \cap \overline{R(\overline{T})} = \emptyset$; ces intersections vides correspondent aux traits épais de la Fig. 3 ; de plus on peut avoir $\overline{R(\overline{T})} \cup \overline{\overline{R(T)}} \neq Y$, etc. ;

iv) $R(T) \cup \overline{R(\overline{T})} = X$, $\overline{R(\overline{T})} \cup \overline{\overline{R(T)}} = X$, $R(T) \cup \overline{\overline{R(T)}} = X$, et $\overline{R(\overline{T})} \cup \overline{R(\overline{T})} = X$; de plus on peut avoir $R(T) \cap \overline{R(\overline{T})} \neq \emptyset$, etc. ; ces unions “pleines” correspondent aux doubles lignes fines de la Fig. 3.

Les conditions (iii)-(iv) tiennent grâce à la sérialité.

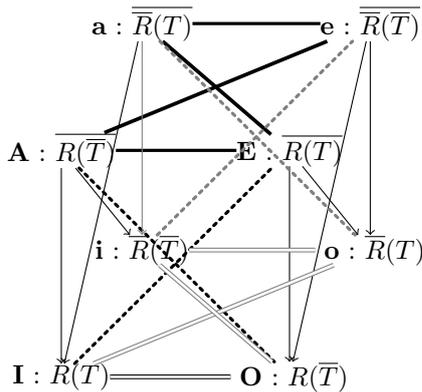


FIGURE 3 – Cube induit par une relation R et un sous-ensemble T

La face de devant du cube correspond exactement à une lecture de type logique modale du carré, où R est vue comme une relation d’accessibilité définie sur $X \times X$, et T comme l’ensemble des modèles d’une proposition p . En effet, dire que $\Box p$ (resp. $\Diamond p$) est vrai dans le monde x signifie que p est vrai dans *tous les*

(resp. dans *au moins un*) monde(s) possible(s) accessible(s) à partir de x ; ceci correspond à $\overline{R(\overline{T})}$ (resp. $R(T)$) qui est l’ensemble des mondes où $\Box p$ (resp. $\Diamond p$) est vrai. De plus, **A** implique **I** correspond à l’axiome (D) de la logique modale, connu pour requérir la sérialité de la relation d’accessibilité [8]. Construire une logique modale incluant toutes les modalités apparaissant sur les sommets du cube entier fait sens en logique épistémique. On dispose alors de deux modalités distinctes, une exprimant ce qui est *connu au moins* ($\overline{R(\overline{T})}$) et l’autre ce qui est *connu au plus* ($\overline{\overline{R(T)}}$) [10]. La conjonction de ces deux modalités permet d’exprimer “qu’on sait seulement que” [25]. Ces deux modalités interviennent aussi quand on modélise à la fois des croyances et des désirs [11].

En dehors de la sémantique des logiques modales, il y a de nombreux formalismes en IA qui exploitent une relation : notamment l’analyse formelle de concepts [22], où la relation représente un contexte formel $R \subseteq X \times Y$ liant objets et leurs propriétés, les “rough sets” [28] qui sont induits par une relation d’indiscernabilité, ou l’argumentation abstraite basée sur une relation d’attaque entre arguments [18]. Les concepts formels sont définis par des paires $(S, T) \subseteq X \times Y$ telles que $\overline{R(T)} = S$ et $\overline{R^t(S)} = T$, qui correspondent à l’application de l’Eq. (3). Cependant, placer l’analyse formelle de concepts dans la perspective du cube, conduit à considérer également les opérateurs définis par les trois autres équations, ce qui s’avère fructueux [15, 9, 16], e.g., l’Eq. (2) est à la base de la définition de sous-contextes indépendants [16]. Les approximations inférieures et supérieures qui constituent un “rough set” sont définies à partir de la relation R sur $X \times X$ au moyen des Eq. (1) et Eq. (2), où xR est l’ensemble des éléments qui sont indiscernables de x w.r.t. R ; voir [9] pour une discussion préliminaire de l’intérêt de considérer également les autres équations dans des cadres généralisant les “rough sets” au delà du cas classique où R est une relation d’équivalence. En ce qui concerne l’argumentation abstraite, il est avantageux de considérer le complément de la relation d’attaque entre arguments, puisqu’alors la contrepartie d’un concept formel est une extension stable, et que les différents sous-ensembles d’arguments associés aux sommets du cube sont dignes d’intérêt [1].

4 Cube des indicateurs. Possibilité tout-ou-rien

Une autre lecture simple, intéressante du cube, non considérée jusqu’à présent, n’est pas relationnelle, mais est en termes d’indicateurs ensemblistes. Revenant au cube de la Fig. 2, les sommets de la face de dessus

peuvent être réécrits en termes d'intersections vides d'ensembles d'objets A , B , ou de leurs compléments \bar{A} , \bar{B} , tandis que la face du dessous correspond à des intersections non vides. Voir Fig.4. Il est à noter qu'on suppose ici que $A \neq \emptyset$, $\bar{A} \neq \emptyset$, $B \neq \emptyset$, et $\bar{B} \neq \emptyset$, afin d'éviter la contrepartie des problèmes d'"import existentiel", puisque maintenant les ensembles A et B jouent des rôles symétriques dans les expressions associées aux sommets du cube.

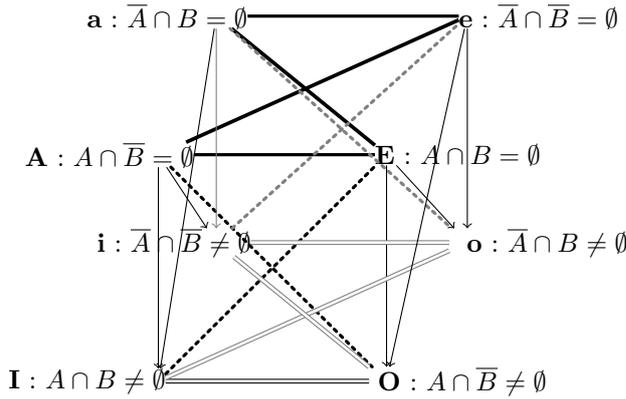


FIGURE 4 – Cube des oppositions des indicateurs de comparaison

Il est à noter que les faces latérales du cube de la Fig. 4 exhibent les quatre indicateurs de comparaison de base existant entre deux ensembles A et B , à savoir ce qu'ils ont en commun de manière positive ($S = A \cap B$), ou négative ($T = \bar{A} \cap \bar{B}$), en quoi A diffère de B ($U = A \cap \bar{B}$), et en quoi B diffère de A ($V = \bar{A} \cap B$). Quand on compare deux sous-ensembles, considérant qu'un indicateur ensembliste peut ou non être vide, on a $2^4 = 16$ configurations qui sont résumées en Table 1.

configuration	$S = A \cap B \neq \emptyset$	$U = A \cap \bar{B} \neq \emptyset$	$V = \bar{A} \cap B \neq \emptyset$	$T = \bar{A} \cap \bar{B} \neq \emptyset$
1	$A \cap B \neq \emptyset, A \not\subseteq B;$ $B \not\subseteq A; A \cup B \neq U$	1	1	1
2	$A \cap B \neq \emptyset, A \not\subseteq B;$ $B \not\subseteq A; A \cup B = U$	1	1	0
3	$B \subset A \subset U$	1	0	1
4	$B \subset A; A = U$	1	0	0
5	$A \subset B \subset U$	0	1	1
6	$A \subset B; B = U$	0	1	0
7	$A = B \subset U$	1	0	1
8	$A = B = U$	1	0	0
9	$A \cap B = \emptyset; A \cup B \neq U$	0	1	1
10	$A \cap B = \emptyset; A \cup B = U$	0	1	0
11	$A \subset U; B = \emptyset;$	0	1	1
12	$A = \emptyset; B = \emptyset$	0	1	0
13	$A = \emptyset; B \subset U$	0	0	1
14	$A = \emptyset; B = U$	0	0	0
15	$A = B = \emptyset; U \neq \emptyset$	0	0	1
16	$A = B = \emptyset = U$	0	0	0

TABLE 1 – Configurations respectives de deux sous-ensembles

Comme on peut le voir sur la Table 1, les lignes 1 et 2 correspondent à des situations de chevauchement sans inclusion, avec couverture du référentiel ($A \cup B = U$) ou non. Les lignes 3, 4, 5 et 6 correspondent à des

situations d'inclusion, avec couverture du référentiel ou non. Les lignes 7 et 8 correspondent à des situations d'égalité, avec couverture du référentiel ou non. Les lignes 9 et 10 correspondent à des situations de non chevauchement, avec couverture du référentiel ou non. Les 6 dernières lignes correspondent à des situations pathologiques où A ou B sont vides, avec couverture du référentiel ou non. Ceci montre que les quatre indicateurs de la Table 1 sont *conjointement nécessaires* pour décrire toutes les situations possibles relatives aux positions possibles de deux sous-ensembles A et B , qui peuvent être vides, dans un référentiel U . De plus on peut vérifier les propriétés suivantes sur la Table 1 :

$$- S \cup T \cup U \cup V = U.$$

Cela signifie que les 4 ensembles ne peuvent être simultanément vides, sauf si le référentiel U est vide ce qui est le cas en ligne 16 de la Table 1.

- Sous les conditions $A \neq \emptyset, B \neq \emptyset, A \neq U, B \neq U$, on a

si $A \cap \bar{B} = \emptyset$ ou $\bar{A} \cap B = \emptyset$, alors $A \cap B \neq \emptyset$ et $\bar{A} \cap \bar{B} \neq \emptyset$.

Cela correspond aux lignes 1, 2, 3, 5, 7, 9, 10 de la Table 1. Cela correspond aussi aux 5 configurations possibles de deux sous-ensembles non-vides A et B (lignes 1, 3, 5, 7, 9), identifiées pour première fois par Gergonne [23, 21] dans sa discussion des syllogismes, plus les deux configurations (lignes 2, 10) où $A \cup B = U$ (mais où $A \neq U, B \neq U$).

Un mérite du cube de la Fig. 4 est qu'il rend évident que le cube des oppositions est compatible avec une version tout-ou-rien de la théorie des possibilités, comme déjà remarqué d'une autre manière dans [15]. En effet, soit $B = E$ ($E \neq \emptyset, E \neq U$) un sous-ensemble qui représente l'information disponible, c'est-à-dire qu'on sait que le monde réel est dans E . Si on considère un événement A , les sommets **A**, **I**, **a** et **i** correspondent respectivement exactement à $N(A) = 1$ (défini par $N(A) = 1$ si $A \subseteq E$, et $N(A) = 0$ sinon), $\Pi(A) = 1$ (défini par $\Pi(A) = 1$ si $A \cap E \neq \emptyset$, et $\Pi(A) = 0$ sinon), $\Delta(A) = 1$ (défini par $\Delta(A) = 1$ si $E \subseteq A$, et $\Delta(A) = 0$ sinon), $\nabla(A) = 1$ (défini par $\nabla(A) = 1$ si $A \cup E \neq U$, et $\nabla(A) = 0$ sinon), où N, Π, Δ , et ∇ sont respectivement la nécessité forte, la possibilité faible, la possibilité forte, et la nécessité faible. De plus la propriété suivante tient : $\max(N(A), \Delta(A)) \leq \min(\Pi(A), \nabla)$ qui exprime que si un événement est fortement nécessaire ou fortement possible, il doit être à la fois faiblement possible et faiblement nécessaire.

5 Le cube graduel et la théorie des possibilités

Certaines notions sont naturellement une question de degré comme l'incertitude, la similarité, la satisfaction, ou le niveau d'attaque en argumentation. Dans les sections précédentes, tout était binaire dans le carré et dans le cube des oppositions. Mais cela fait sens d'avoir des modalités graduées, que la théorie des possibilités soit graduelle, et plus généralement d'avoir des relations graduées ou des sous-ensembles flous dans les constructions précédentes du cube. Cela permettrait aussi d'englober des extensions graduelles de la théorie des "rough sets", de l'analyse formelle de concepts (comme, e.g., celle proposée dans [3]), ou de l'argumentation abstraite [19]. Dans ce qui suit, on définit une extension graduelle du cube des oppositions, et indiquons ensuite comment elle s'applique à la théorie des possibilités graduées, mais aussi à d'autres cadres graduels comme l'agrégation multicritère et à l'intégrale de Sugeno d'une part, et aux fonctions de croyance de Shafer d'autre part, laissant pour des études détaillées futures les autres extensions graduées mentionnées ci-dessus.

Les extensions graduées du cube doivent satisfaire une version multi-valuée des contraintes (i)-(iv) du carré des oppositions (voir Section 2) pour les faces de devant et de derrière, ainsi que des contraintes d'implication des faces latérales, des contraintes exclusion mutuelle de la face de dessus, et des contraintes duales de la face de dessous. Soient $\alpha, \iota, \epsilon, o$, et $\alpha', \iota', \epsilon', o'$ les degrés dans $[0, 1]$ associés respectivement aux sommets **A**, **I**, **E**, **O** et **a**, **i**, **e**, **o**. Etant donné une négation involutive n , une conjonction symétrique $*$, et interprétant l'implication en logique multi-valente par l'inégalité \leq (la conclusion est au moins aussi vraie que la prémisse), les contraintes pour les faces de devant et de derrière du cube s'écrivent alors

- (i) $\alpha = n(o)$, $\epsilon = n(\iota)$ et $\alpha' = n(o')$ et $\epsilon' = n(\iota')$;
- (ii) $\alpha \leq \iota$, $\epsilon \leq o$ et $\alpha' \leq \iota'$, $\epsilon' \leq o'$;
- (iii) $\alpha * \epsilon = 0$ et $\alpha' * \epsilon' = 0$;
- (iv) $n(\iota) * n(o) = 0$ et $n(\iota') * n(o') = 0$.

Les contraintes associées aux faces latérales

- (v) $\alpha \leq \iota'$, $\alpha' \leq \iota$ et $\epsilon' \leq o$, $\epsilon \leq o'$;

et pour les faces de dessus et de dessous, on a :

- (vi) $\alpha' * \epsilon = 0$, $\alpha * \epsilon' = 0$;
- (vii) $n(\iota') * n(o) = 0$, $n(\iota) * n(o') = 0$.

Le choix standard pour une négation involutive est $n(\gamma) = 1 - \gamma$. Plusieurs choix peuvent être considérés pour l'opérateur de conjonction $*$. Deux choix présentant un intérêt particulier sont $*$ = min, tel que $\min(\gamma, \delta) = 0$ ssi $\gamma = 0$ ou $\delta = 0$, et la conjonction de Łukasiewicz $*$ = $\max(0, \cdot + \cdot - 1)$, telle que $\max(0, \gamma + \delta - 1) = 0$ ssi $\gamma \leq n(\delta)$ ssi $\min(\gamma, \delta) \leq 0.5$.

Notons que si $*$ est la conjonction de Łukasiewicz, alors les conditions (vi-vii) du dessus et du dessous sont équivalentes à celles des faces latérales (v). En effet, de $\alpha' * \epsilon = 0$, on obtient $\alpha' * n(\iota) = 0$ qui tient ssi $\alpha' \leq \iota$. Les autres conditions s'obtiennent de manière similaire. Des résultats plus faibles peuvent être établis pour $*$ = min : les conditions (v) sont plus faibles que les conditions (vi-vii). En effet, $\min(\alpha', \epsilon) = 0$ implique $\alpha' = 0$ ou $\epsilon = 0$, et par conséquent $\alpha' \leq \iota = n(\epsilon)$. D'autre part, α' peut être plus petit ou égal à ι avec $\alpha \neq 0$ et $\iota \neq 1$.

Un tel cube graduel peut recevoir différentes instanciations. Une est en termes de théorie des possibilités (graduées) [14], comme indiqué brièvement ci-dessous. Considérons une distribution de possibilités normalisée $\pi : \Omega \rightarrow [0, 1]$, qui est aussi telle que $1 - \pi$ est normalisée (i.e., $\exists \omega \in \Omega, \pi(\omega) = 0$), soit $\Delta(A) = \min_{\omega \in A} \pi(\omega)$ la quantité possibilité forte d'une proposition dont A est l'ensemble des modèles, et soit $\nabla(A) = 1 - \Delta(\bar{A})$ sa conjuguée. On peut instancier le carré des oppositions graduel en posant $\alpha' = \Delta(A)$, $\epsilon' = \Delta(\bar{A})$, $\iota' = \nabla(A)$, $o' = \nabla(\bar{A})$. Grâce à la dualité entre ι' , o' on a un carré des oppositions pour $*$ = min, et $n(\gamma) = 1 - \gamma$. En particulier, puisque $\Delta(A) \leq \Pi(A)$ et $\nabla(A) \leq N(A)$, les contraintes des faces latérales tiennent sous la forme (v).

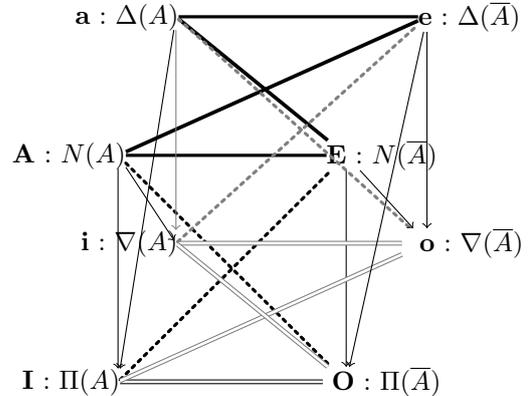


FIGURE 5 – Cube des oppositions en théorie des possibilités

En accord avec la logique épistémique, on peut exprimer qu'au moins C est sûr à un certain degré (i.e., tous les éléments en dehors de C sont quelque peu impossibles), ce qui est représenté par la contrainte $N(C) \geq \gamma > 0$; et qu'aucun énoncé plus précis que D n'est sûr (i.e., tous les éléments dans D sont possibles à un certain degré), ce qui est représenté par la contrainte $\Delta(D) \geq \delta > 0$. Notons que $N_\pi(C) = \Delta_{1-\pi}(\bar{C})$; cependant les quantités $N_\pi(A)$ et $\Delta_\pi(A)$ sont complètement indépendantes l'une de l'autre.

6 Agrégations pondérées. Intégrales de Sugeno

Comme dit dans la partie précédente, la satisfaction est généralement une question de degrés. C'est le cas par exemple pour l'agrégation multicritères où des objets sont évalués par rapport à des critères $i \in \{1, \dots, n\}$. L'évaluation d'un objet f par rapport à un critère i est notée f_i et l'objet est représenté par le vecteur $f = (f_1, \dots, f_i, \dots, f_n)$. On suppose que $\forall i, f_i \in [0, 1]$. $f_i = 1$ signifie que l'objet satisfait pleinement le critère i , tandis que $f_i = 0$ exprime une totale absence de satisfaction. Soit $\pi_i \in [0, 1]$ le niveau d'importance du critère i . Plus π_i est grand plus le critère est important. De plus on suppose la double normalisation $\exists i, \pi_i = 1$, et $\exists j, \pi_j = 0$.

Le minimum pondéré et le maximum pondéré sont des opérateurs d'agrégation qualitatifs simples [13]. Le premier mesure à quel niveau les critères importants sont satisfaits et correspond à l'expression $\bigwedge_{i=1}^n \pi_i \Rightarrow f_i$, tandis que le second $\bigvee_{i=1}^n \pi_i \wedge f_i$ est optimiste et demande seulement que au moins un critère important soit fortement satisfait. Ces opérateurs d'agrégation correspondent aux sommets **A** et **I** du cube de la Fig. 6.

La condition (i) de la face de devant entraîne $s \Rightarrow t = (1 - s) \vee t$ qui est la forte implication associée à $\wedge = \min$. On peut montrer facilement que le cube de la Fig. 6 est juste la contrepartie multi valuée du cube initial de la Fig. 2. La valeur de $MIN_\pi(f) = \bigwedge_{i=1}^n \pi_i \Rightarrow f_i$ est d'autant plus grande que tous les critères importants sont beaucoup satisfaits. L'opérateur $MIN_\pi^{neg}(f) = \bigwedge_{i=1}^n (1 - \pi_i) \Rightarrow (1 - f_i)$ tolère de faibles évaluations ($1 - f_i$ est grand) quand les critères ont une importance faible. Les agrégations de la face de devant du cube de la Fig. 6 sont des évaluations positives qui se focalisent sur les critères importants fortement satisfaits, tandis que les agrégations de la face de derrière sont négatives car elles prennent en compte l'absence de dissatisfactions pour les critères importants. Ceux sont deux points de vue complémentaires récemment proposés pour l'agrégation multi-critères [17]. Le cube de la Fig. 6 satisfait toutes les propriétés (i-vii) du cube des oppositions gradué if $\ast = \max(0, \cdot + \cdot - 1)$ est la conjonction de Łukasiewicz.

Les intégrales de Sugeno [31] [24] constituent une importante famille d'opérateurs d'agrégation qualitatifs qui inclut les minimum et maximum pondérés comme cas particuliers, et où les sous-ensembles de critères peuvent être pondérés (et pas seulement les critères) pour exprimer les synergies au sein de ces sous-ensembles. L'intégrale de Sugeno est définie par :

$$\oint_\gamma(f) = \bigvee_{A \subseteq C} \gamma(A) \wedge \bigwedge_{i \in A} f_i$$

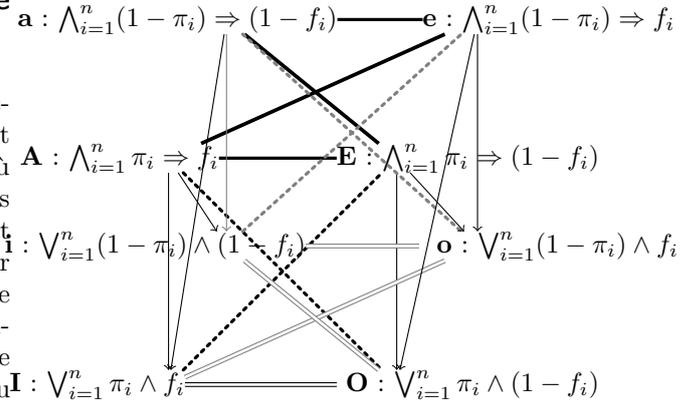


FIGURE 6 – Cube des agrégations pondérées qualitatives

où le niveau d'importance des sous-ensembles de critères A est représenté par une fonction d'ensemble, appelée capacité, qui est une application $\gamma : 2^C \rightarrow L$ telle que $\gamma(\emptyset) = 0$, $\gamma(C) = 1$, et si $A \subseteq B$ alors $\gamma(A) \leq \gamma(B)$. Les mesures de possibilité Π , ou les mesures de nécessité N sont des exemples de capacités. Notons que si f est la fonction caractéristique de l'ensemble $F \subseteq C$, alors $\oint_\gamma(f) = \gamma(F)$.

L'implication de **A** vers **I** dans le cube de la Fig. 5, qui exprime que $N(A) \leq \Pi(A)$ pour tout A , reflète le fait que N donne une évaluation pessimiste, tandis que $\Pi(A)$ est une évaluation optimiste. Afin de généraliser cette situation à toute capacité γ , on a besoin d'introduire la partie pessimiste γ_* et la partie optimiste γ^* de γ . Pour cela on a besoin de définir la conjuguée $\gamma^c(A)$ d'une capacité γ , i.e. la capacité $\gamma^c(A) = 1 - \gamma(\bar{A})$, $\forall A \subseteq C$, où \bar{A} est le complémentaire du sous-ensemble A . En raison de la dualité $N(A) = 1 - \Pi(\bar{A})$, N et Π sont conjuguées l'une de l'autre. Alors, soit $\gamma_*(A) = \min(\gamma(A), \gamma^c(A))$ et $\gamma^*(A) = \max(\gamma(A), \gamma^c(A))$. On a $\gamma_*(A) \leq \gamma^*(A)$. Notons que $\gamma_*(A) = 1 - \gamma^*(\bar{A})$ (γ_* et γ^* sont conjuguées). On peut alors construire la face de devant du cube de la Fig. 7, où γ_* est utilisée sur les sommets **A** et **E**, tandis que γ^* apparaît sur les sommets **I** et **O**. On peut montrer que cette face satisfait toutes les propriétés (i)-(iv) d'un carré des oppositions avec la conjonction de Łukasiewicz $\ast = \max(0, \cdot + \cdot - 1)$.

Le face de derrière du cube de la Fig. 7 est obtenue en changeant l'intégrale \oint_γ en une désintégrale [17] $\oint_\nu^\downarrow(f) = \oint_{1-\nu^c}(1-f)$, où ν est appelée une anti-capacité, car c'est une fonction d'ensemble décroissante (alors $1 - \nu^c$ est une capacité, où $\nu^c(A) = 1 - \nu(\bar{A})$). Etant donnée une capacité pessimiste γ , l'anti capacité associée ν est définie par $\nu(A) = \bar{\gamma}(\bar{A})$ à partir du complément pessimiste de la capacité $\bar{\gamma}$, lui même défini de la manière suivante. Tout

d'abord rappelons ce qu'est la transformée de Moebius qualitative $\gamma_{\#}$ d'une capacité $\gamma : \gamma_{\#}(E) = \gamma(E)$ si $\gamma(E) > \max_{B \subsetneq E} \gamma(B)$ et $\gamma_{\#}(E) = 0$ sinon. Alors $\gamma(A) = \max_{E \subset A} \gamma_{\#}(E)$ (c'est la contrepartie qualitative de la définition des fonctions de croyance à partir de la fonction de masse). On peut alors définir la complémentaire $\bar{\gamma}_{\#}$ de $\gamma_{\#}$ par $\bar{\gamma}_{\#}(\bar{E}) = \gamma_{\#}(E)$ pour tout E . A partir de la complémentaire on définit $\bar{\gamma}(A) = \max_{E \subset A} \bar{\gamma}_{\#}(E) = \max_{\bar{E} \subset \bar{A}} \gamma_{\#}(E)$, ceci rend évident que $\bar{\gamma}(\bar{A})$ généralise $\Delta(A)$ de la théorie des possibilités (en effet $\bar{\gamma}(\bar{A}) = \max_{A \subset E} \gamma_{\#}(E)$). Alors $\mathcal{f}_{\nu}^{\dagger}(f) = \mathcal{f}_{1-\nu^c}(1-f) = \mathcal{f}_{\bar{\gamma}}(1-f)$ car $1-\nu^c(A) = \nu(\bar{A}) = \bar{\gamma}(A)$.

Lorsque γ est une mesure de nécessité, $\nu(A)$ devient $\Delta(A) = N_{1-\pi}(\bar{A})$. Les désintégrales généralisent l'opérateur qualitatif MIN_{π}^{neg} qui se focalisent sur les dissatisfactions des objets lors du processus d'évaluation. Il est alors possible de vérifier que toutes les propriétés (i)-(vii) du cube des oppositions gradué pour la conjonction de Lukasiewicz $* = \max(0, \cdot + \cdot - 1)$ sont satisfaites dans le cube de la Fig. 7, qui généralise le cube Fig. 6.

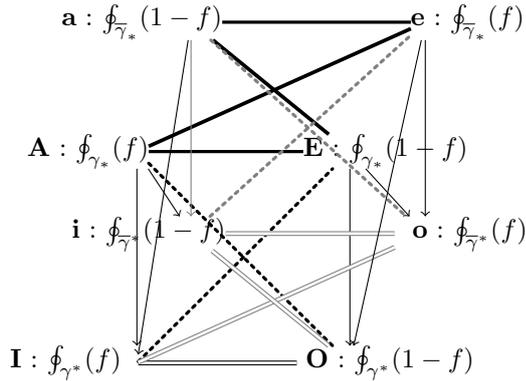


FIGURE 7 – Cube des oppositions induit par une intégrale de Sugeno

7 Cube des oppositions des fonctions de croyance

Dans la théorie de l'évidence de Shafer [30], une fonction de croyance et sa duale la fonction de plausibilité sont définies à partir d'une fonction de masse $m : Bel_m(A) = \sum_{E \subset A} m(E)$ pour $A \subseteq \mathcal{U}$. et $Pl_m(A) = 1 - Bel_m(\bar{A}) = \sum_{E \cap A \neq \emptyset} m(E)$. On suppose que $m(\emptyset) = 0$ et $\sum_E m(E) = 1$. La fonction de masse complémentaire \bar{m} est définie par $\bar{m}(E) = m(\bar{E})$ [12]. La condition de normalisation $\bar{m}(\emptyset) = 0$ impose que $m(\mathcal{U}) = 0$. La fonction de communalité Q et sa dualité \hat{O} sont définies par $Q_m(A) = \sum_{A \subset E} m(E) = Bel_{\bar{m}}(\bar{A})$ et $\hat{O}_m(A) = \sum_{\bar{E} \cap \bar{A} \neq \emptyset} m(E) = 1 - Q_m(\bar{A}) = Pl_{\bar{m}}(\bar{A})$. Il est facile de vérifier que la transformation $m \rightarrow \bar{m}$

revient à la transformation $\pi \rightarrow 1-\pi$ dans le cas d'éléments focaux emboîtés (i.e. les ensembles E tels que $m(E) > 0$). Cela montre que la théorie des possibilités et le cube Fig. 8 sont la contrepartie du cube Fig. 5. De plus, les fonctions de croyance s'étendent à des fonctions caractéristiques d'ensembles flous sous la forme $Bel_m(A) = \sum_E m(E) \cdot \min_{u \in E} \mu_A(u)$, ce qui est un cas particulier de l'intégrale de Choquet, comme la nécessité d'un événement flou $MIN_{\pi}(f)$ est un cas particulier de l'intégrale de Sugeno. Il est évident que dans le cube Fig. 8, les fonctions d'ensemble s'étendent à des événements flous, et que plus généralement on peut considérer son extension aux intégrales de Choquet.

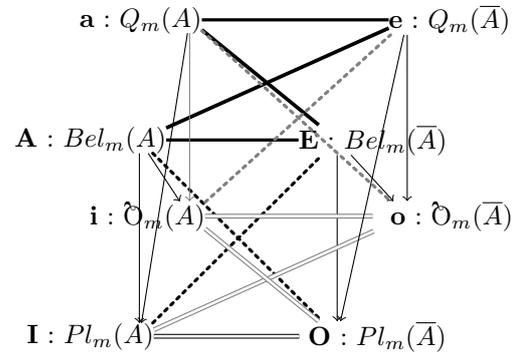


FIGURE 8 – Cube des oppositions des fonctions de croyance

8 Conclusion

Nous avons vu que le cube des oppositions fait sens pour une grande variété de situations incluant des cadres qualitatifs graduels et des cadres quantitatifs de modélisation. Les 4 entités associées aux faces latérales du cube ne sont que faiblement liées entre elles et sont toutes nécessaires pour décrire ou évaluer complètement une situation. L'existence de la face de derrière complète sur un mode renversé ce qui est à l'œuvre dans le carré des oppositions de devant. Prendre conscience de l'existence d'un cube des oppositions pour un cadre de modélisation donné, peut aider à introduire de nouvelles notions utiles de façon à avoir un cube complet. Le cube des oppositions peut aussi aider à faire de fructueux parallèles entre des théories, ou à les hybrider. L'auto-dualité des probabilités ($Prob(\bar{A}) = 1 - P(A)$) empêche leur association avec un carré et un cube des oppositions non dégénérés; la possibilité de l'introduction des probabilités supérieures et inférieures générales (au delà des fonctions de croyance) dans un cube d'opposition reste à étudier.

Références

- [1] L. Amgoud and H. Prade. Can AI models capture natural language argumentation? *Int. J. Cognit. Inform. and Natural Intellig.*, 6(3) :19–32, 2012.
- [2] L. Amgoud and H. Prade. A formal concept view of abstract argumentation. In L. C. van der Gaag, editor, *Proc. 12th Eur. Conf. Symb. and Quant. Appr. to Reas. with Uncert. (ECS-QARU'13)*, Utrecht, July 8-10, LNCS 7958, pages 1–12. Springer, 2013.
- [3] R. Belohlavek. *Fuzzy Relational Systems : Foundations and Principles*. Kluwer Aca. Pu., 2002.
- [4] J.-Y. Béziau. New light on the square of oppositions and its nameless corner. *Logical Investigations*, 10 :218–233, 2003.
- [5] J.-Y. Béziau. The power of the hexagon. *Logica Universalis*, 6 (1-2) :1–43, 2012.
- [6] R. Blanché. Sur l'opposition des concepts. *Theoria*, 19 :89–130, 1953.
- [7] R. Blanché. *Structures Intellectuelles. Essai sur l'Organisation Systématique des Concepts*. Vrin, Paris, 1966.
- [8] B. F. Chellas. *Modal Logic, an Introduction*. Cambridge University Press, Cambridge, 1980.
- [9] D. Ciucci, D. Dubois, and H. Prade. The structure of oppositions in rough set theory and formal concept analysis - Toward a new bridge between the two settings. In C. Beierle and C. Meghini, editors, *Proc. 8th Int. Symp. on Foundations of Information and Knowledge Systems (FoIKS'14)*, Bordeaux, Mar. 3-7, volume 8367 of LNCS, pages 154–173. Springer, 2014.
- [10] D. Dubois, P. Hájek, and H. Prade. Knowledge-driven versus data-driven logics. *J. of Logic, Language and Information*, 9(1) :65–89, 2000.
- [11] D. Dubois, E. Lorini, and H. Prade. Bipolar possibility theory as a basis for a logic of desires and beliefs. In W. Liu, V. S. Subrahmanian, and J. Wijsen, editors, *Proc. 7th Int. Conf. on Scalable Uncertainty Management - SUM'13 Washington, DC, Sept. 16-18*, volume 8078 of LNCS, pages 204–218. Springer, 2013.
- [12] D. Dubois and H. Prade. A set-theoretic view of belief functions. Logical operations and approximation by fuzzy sets. *Int. J. of General Systems*, 12(3) :193–226, 1986.
- [13] D. Dubois and H. Prade. Weighted minimum and maximum operations, An addendum to 'A review of fuzzy set aggregation connectives'. *Information Sciences*, 39 :205–210, 1986.
- [14] D. Dubois and H. Prade. Possibility theory : Qualitative and quantitative aspects. In D. M. Gabbay and Ph. Smets, editors, *Quantified Representation of Uncertainty and Imprecision*, volume 1 of *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, pages 169–226. Kluwer Acad. Publ., 1998.
- [15] D. Dubois and H. Prade. From Blanché's hexagonal organization of concepts to formal concept analysis and possibility theory. *Logica Univers.*, 6 :149–169, 2012.
- [16] D. Dubois and H. Prade. Possibility theory and formal concept analysis : Characterizing independent sub-contexts. *Fuzzy Sets and Systems*, 196 :4–16, 2012.
- [17] D. Dubois, H. Prade, and A. Rico. Qualitative integrals and desintegrals : How to handle positive and negative scales in evaluation. In S. Greco et al., editor, *Proc. 14th Int. Conf. on Inform. Process. and Mgmt. of Uncert. in Knowledge-Based Syst. (IPMU'12, Part III)*, Catania, July 9-13, volume 299 of CCIS, pages 306–316. Springer, 2012.
- [18] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2) :321–358, 1995.
- [19] P. E. Dunne, A. Hunter, P. McBurney, S. Parsons, and M. Wooldridge. Weighted argument systems : Basic definitions, algorithms, and complexity results. *Artificial Intelligence*, 175(2) :457–486, 2011.
- [20] L. Euler. Lettres cii-cviii. In *Lettres à une Princesse d'Allemagne sur Divers Sujets de Physique & de Philosophie*, vol 2. 1768.
- [21] J. A. Faris. The Gergonne relations. *J. of Symbolic Logic*, 20 (3) :207–231, 1955.
- [22] B. Ganter and R. Wille. *Formal Concept Analysis : Mathematical Foundations*. Springer-Verlag, Berlin, Heidelberg, 1999.
- [23] J. D. Gergonne. Essai de dialectique rationnelle. *Annales de Mathématiques Pures et Appliquées*, 7 :189–228, 1816.
- [24] M. Grabisch and Ch. Labreuche. A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid. *Annals of Oper. Res.*, 175 :247–286, 2010.
- [25] H. J. Levesque. All I know : A study in autoepistemic logic. *Artif. Intellig.*, 42(2-3) :263–309, 1990.

- [26] P. Marquis, O. Papini, and H. Prade. Some elements for a prehistory of artificial intelligence in the last four centuries. In T. Schaub, G. Friedrich, and B. O’Sullivan, editors, *Proc. 21st Europ. Conf. on Artificial Intelligence (ECAI’14), Prague, Aug. 18-22*, pages 609–614. IOS Press, 2014.
- [27] T. Parsons. The traditional square of opposition. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. 2008.
- [28] Z. Pawlak. *Rough Sets. Theoretical Aspects of Reasoning about Data*. Kluwer, 1991.
- [29] J. Piaget. *Logic and Psychology*. Manchester Univ. Press, 1953.
- [30] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.
- [31] M. Sugeno. Fuzzy measures and fuzzy integrals : a survey. In M. M. Gupta, G. N. Saridis, and B. R. Gaines, editors, *Fuzzy Automata and Decision Processes*, pages 89–102. North-Holland, 1977.

Une logique épistémique modeste basée sur les observations d'ordre supérieur

Andreas Herzig Emiliano Lorini Faustine Maffre

University of Toulouse, IRIT
118, Route de Narbonne
F-31062 Toulouse, France

{Andreas.Herzig, Emiliano.Lorini, Faustine.Maffre}@irit.fr

Résumé

Nous présentons une logique épistémique dynamique basée sur ce qu'un agent peut observer, y compris les observations jointes et l'observation de ce que les autres observent, généralisant ainsi les logiques ECL-PC(PO) et LRC de van der Hoek, Wooldridge et al. où les variables propositionnelles que chaque agent observe sont connaissance commune. Dans notre logique, les propositions ainsi que la visibilité de ces propositions peuvent être modifiées par des programmes d'affectation. Nous montrons comment les opérateurs épistémiques peuvent alors être interprétés. Nous proposons une axiomatisation adéquate et complète et prouvons que le problème de satisfiabilité est PSpace-complet. Enfin, nous montrons comment les annonces publiques et privées peuvent être exprimées dans notre logique.

1 Introduction

Dans son livre pionnier de 1962 [11], Hintikka proposa d'analyser les phrases de la forme "l'agent i sait φ dans un monde possible w " par " φ est vrai dans tous les mondes que i ne peut pas distinguer de w ". De telles logiques épistémiques ont recours à la sémantique des mondes possibles de Kripke. Elles ont été popularisées en informatique par Fagin et al. dans les années 90 [4].

Au cours des dernières années, plusieurs auteurs ont étudié comment la logique épistémique pourrait plutôt se baser sur la notion de visibilité (ou observabilité) de variables propositionnelles. Les approches les plus remarquables sont Epistemic Coalition Logic of Propositional Control with Partial Observability ECL-PC(PO) [17] et Logic of Revelation and Concealment LRC [16].

Ces logiques trouvent leur origine dans les langages utilisés dans des vérificateurs de modèle tels que MOCHA pour la description des systèmes distribués [2]. L'idée de base est que la phrase "l'agent i sait que le fait atomique p est vrai" peut être réduit à (1) p est vrai et (2) l'agent i observe la valeur de vérité de p . De la même façon, "l'agent i sait que p est faux" peut être réduit à (1) p est faux et (2) l'agent i observe la valeur de vérité de p . On suppose donc que chaque agent a un ensemble de faits atomiques (alias variables propositionnelles) qu'il peut observer, dans le sens où il connaît leurs valeurs de vérité. À l'inverse, toute combinaison des valeurs de vérité des variables non observées est possible pour l'agent. Ces informations d'observabilité permettent de reconstruire la sémantique de Hintikka : une relation d'accessibilité kripkéenne \sim_i entre les mondes possibles w et w' peut être associée à l'agent i par la définition suivante : $w \sim_i w'$ si et seulement si chaque variable observée par i dans w a la même valeur de vérité dans w et dans w' . Cela permet de donner la sémantique de phrases de la forme " i sait que φ " où φ est une formule complexe arbitraire (et pas seulement une variable propositionnelle ou sa négation).

À la fois ECL-PC(PO) et LRC, et plus généralement toutes les logiques épistémiques existantes basées sur la visibilité dans la littérature, supposent que ce que chaque agent peut voir est connaissance commune. C'est une hypothèse forte que nous allons relâcher dans ce papier : nous allons permettre des situations où l'agent i voit la variable p , mais l'agent j ne voit pas si i voit p . Ceci permet de modéliser de façon naturelle des scénarios tels que le problème de bavardage

où six amis échangent leurs secrets via des appels téléphoniques ou le problème des deux généraux où deux agents doivent se coordonner via une communication asynchrone non fiable. Dans notre approche, nous modélisons les informations de visibilité non seulement en termes de variables associées à des agents (comme cela est fait dans ECL-PC(PO) and LRC), mais plus généralement en termes de variables associées à des séquences d’agents. Syntactiquement, nous représentons ceci au moyen de formules atomiques que nous appelons des *atomes de visibilité*. Ils sont de la forme $S_{i_1} S_{i_2} \dots S_{i_n} p$, où p est une variable propositionnelle, et i_1, i_2, \dots, i_n sont des agents. Si $n=0$, alors on a seulement une variable propositionnelle. Si $n=1$, l’atome $S_{i_1} p$ se lit “l’agent i_1 voit la valeur de la variable p ”, et si $n=2$, l’observation du second ordre $S_{i_1} S_{i_2} p$ se lit “l’agent i_1 voit si i_2 voit la valeur de p ”; etc.

Notre sémantique est basée sur des *évaluations*, qui sont simplement des ensembles d’atomes de visibilité. Afin de garantir l’introspection positive et négative, nous devons veiller à ce que l’agent soit toujours conscient des variables propositionnelles qu’il voit : pour chaque agent i et chaque variable propositionnelle p , nous exigeons que $S_i S_i p$ soit dans chaque évaluation. Plus généralement, on considère que l’évaluation V est introspective si elle contient chaque atome de visibilité ayant deux S_i consécutifs. Ainsi, dans une évaluation introspective, l’agent est conscient de ce qu’il voit, et tout agent voit cela, et tout agent voit que tout agent voit cela, etc. Les informations de visibilité permettent d’interpréter les opérateurs épistémiques : pour les variables propositionnelles p , la formule $K_i p$ est vraie dans une évaluation V si V contient à la fois p et $S_i p$. Plus généralement, la condition de vérité pour $K_i \varphi$ est basée sur une relation entre les évaluations qui peut être construite grâce à nos atomes de visibilité : $V \sim_i V'$ si chaque atome que i voit dans V a la même valeur de vérité dans V et dans V' . Alors que les relations \sim_i sont toujours réflexives, elles sont symétriques et transitives — et sont donc des relations d’équivalence — sur l’ensemble des évaluations introspectives seulement.

Une autre nouveauté de notre approche par rapport aux logiques épistémiques existantes fondées sur la visibilité est que notre langage comprend une formule atomique spéciale pour l’attention conjointe, de la forme $JS p$, qui signifie “tous les agents voient conjointement la valeur de p ”. Métaphoriquement, l’attention conjointe peut être comprise comme un contact visuel entre les agents lors de l’observation de quelque chose. Tout comme la visibilité individuelle, nous permettons les observations jointes d’ordre supérieur, et ajoutons une contrainte sur les évaluations qui garantit l’introspection de la connaissance commune. Nous exigeons

en outre que la visibilité conjointe implique la visibilité individuelle en imposant que $S_i p \in V$ lorsque $JS p \in V$. Ceci permet d’interpréter un opérateur de connaissance commune CK modal de la même façon que celui de la connaissance individuelle.

Comme plusieurs propositions existantes, nous nous inspirons des logiques épistémiques dynamiques DEL [18] et ajoutons un côté dynamique à notre logique épistémique basée sur l’observation. Plus précisément, nous adaptons la logique LRC de van der Hoek et al. qui dispose de deux opérations de mise à jour modifiant la visibilité : révéler et cacher la valeur d’une variable à un agent. Ces deux primitives ne peuvent toutefois pas être utilisées en l’état car la mise à jour naïve d’une évaluation peut la rendre non introspective. Nous évitons ceci par une définition appropriée de la mise à jour d’une évaluation. Nous utilisons les mêmes programmes d’affectation que ceux de Dynamic Logic of Propositional Assignments DL-PA [10] [3], qui est un dialecte de Propositional Dynamic Logic PDL [6] où les programmes atomiques abstraits de PDL sont instanciés par des affectations de valeurs de vérité à des formules atomiques. L’avantage de ceci est d’obtenir une borne supérieure de la complexité PSPACE, à la fois pour le problème de satisfiabilité et celui de vérification de modèle. Nous montrons aussi comment les mises à jour de visibilité peuvent capturer les annonces publiques et privées des atomes et de leur négation.

Nous appelons notre logique DEL-PAO : Dynamic Epistemic Logic of Propositional Assignment and Observation.

Le papier est organisé comme suit : les sections 2 et 3 introduisent le langage et la sémantique de DEL-PAO. Les sections 4 et 5 contiennent une axiomatisation et le résultat de complexité. La section 6 illustre notre logique par deux applications : l’encodage des annonces et une modélisation du problème des généraux. La section 7 traite des travaux connexes et la section 8 conclut.¹

2 Langage

Soit $Prop$ un ensemble non vide dénombrable de variables propositionnelles et soit Agt un ensemble fini non vide d’agents. Les formules atomiques de notre langage sont des séquences d’opérateurs de visibilité suivies de variables propositionnelles. La définition formelle est la suivante.

L’ensemble des *opérateurs d’observabilité* est

$$OBS = \{S_i : i \in Agt\} \cup \{JS\},$$

où S_i représente de visibilité individuelle de l’agent i et JS représente la visibilité conjointe de tous les agents.

1. Une version complète de ce papier incluant les preuves (en anglais) est disponible à <http://tinyurl.com/o3q86ay>.

L'ensemble de toutes les séquences d'opérateurs de visibilité est noté OBS^* et l'ensemble de toutes les séquences non vides est noté OBS^+ . Nous utilisons σ, σ', \dots pour les éléments de OBS^* . Enfin, l'ensemble des formules atomiques est

$$ATM = \{\sigma p : \sigma \in OBS^*, p \in Prop\}.$$

Les éléments de cet ensemble sont également appelés *atomes de visibilité*, ou simplement atomes. Par exemple, $JS S_2 q$ signifie “tous les agents voient conjointement si l'agent 2 voit la valeur de q ”; en d'autres termes, il est attention conjointe dans le groupe de tous les agents concernant la visibilité sur q de 2. Nous utilisons $\alpha, \alpha', \dots, \beta, \beta', \dots$ pour les éléments de ATM .

Le langage de DEL-PAO est alors défini par la grammaire suivante :

$$\begin{aligned} \pi &::= +\alpha \mid -\alpha \mid \pi; \pi \mid \pi \sqcup \pi \mid \varphi? \\ \varphi &::= \alpha \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi \mid CK\varphi \mid [\pi]\varphi \end{aligned}$$

où α prend ses valeurs dans ATM et i dans Agt .

Nos programmes atomiques sont des affectations de valeurs de vérité à des atomes de ATM : $+\alpha$ rends α vrai et $-\alpha$ le rend faux. Les programmes complexes sont construits avec les opérateurs de PDL : $\pi; \pi'$ est la composition séquentielle, $\pi \sqcup \pi'$ est le choix non déterministe, et $\varphi?$ est le test. Comme dans PDL, la formule $[\pi]\varphi$ signifie “après chaque exécution de π , φ est vrai”. La formule $K_i\varphi$ signifie “ i sait que φ est vrai sur la base de ce qu'il observe”, et $CK\varphi$ signifie “tous les agents savent conjointement que φ est vrai sur la base de ce qu'ils observent conjointement”. Nos opérateurs épistémiques captent des formes de connaissances individuelles et communes qui sont respectivement obtenus par observation individuelle et l'observation conjointe des faits. Cela diffère donc conceptuellement des opérateurs classiques de la connaissance individuelle et commune comme étudié dans le domaine de la logique épistémique [4]. Nous reviendrons sur ce point dans la section 3.4.

Les autres opérateurs booléens $\top, \perp, \vee, \rightarrow$ and \leftrightarrow sont définis comme d'habitude, et $\neg K_i \neg \varphi$ est abrégé par $\widehat{K}_i \varphi$. Le programme $\top?$ est abrégé par *skip* (“ne rien faire”) et $\perp?$ par *fail*.

Nous notons $ATM(\varphi)$ (respectivement $ATM(\pi)$) l'ensemble des atomes apparaissant dans la formule φ (respectivement le programme π). Par exemple, pour $\varphi = [\pi]S_1 JS p \rightarrow q$ avec $\pi = q?; +S_2 p$, on a $ATM(\varphi) = \{q, S_2 p, S_1 JS p\}$ et $ATM(\pi) = \{q, S_2 p\}$. $JS p$ n'est donc pas un atome de φ . La longueur des formules φ et des programmes π , notée $length(\varphi)$ et $length(\pi)$, est le nombre de symboles utilisés pour les écrire, où nous ne comptons pas $[,]$ et les parenthèses et considérons que la longueur de JS, CK , les noms des agents et des variables propositionnelles

est de 1. Ainsi, les symboles S_i et K_i ont une longueur de 2. Par exemple, on a $length(S_2 S_2 p) = 5$ et $length([+S_2 p]JS p \rightarrow q) = 8$. On note que $ATM(\varphi) \leq length(\varphi)$.

3 Sémantique

Nous définissons les évaluations et précisons les contraintes qui sont motivées par le fait que les informations de visibilité devrait être introspectives et que la visibilité commune devrait impliquer la visibilité individuelle. Nous définissons ensuite les relations d'indistinguabilité entre évaluations ainsi que l'interprétation des formules et des programmes.

3.1 Évaluations introspectives

Definition 1. Une évaluation est un sous-ensemble de l'ensemble des atomes ATM . Une évaluation $V \in 2^{ATM}$ est introspective si et seulement si elle respecte les propriétés suivantes pour tout $\alpha \in ATM$ et $i \in Agt$:

$$S_i S_i \alpha \in V \quad (C1)$$

$$JS JS \alpha \in V \quad (C2)$$

$$JS S_i S_i \alpha \in V \quad (C3)$$

$$si JS \alpha \in V, alors S_i \alpha \in V \quad (C4)$$

$$si JS \alpha \in V, alors JS S_i \alpha \in V \quad (C5)$$

L'ensemble des évaluations introspectives est noté $INTR$.

(C1) concerne l'introspection de la vue individuelle : un agent voit toujours s'il voit la valeur de n'importe quel atome. (C2) requiert la même propriété pour la vue commune ; en effet, si $JS \alpha$ est vrai, alors $JS JS \alpha$ doit être vrai par introspection, et si $JS \alpha$ est faux alors tous les agents voient conjointement qu'au moins un d'entre eux a perdu le contact visuel. (C3) oblige la première propriété à être connaissance commune. (C4) garantit que la visibilité conjointe implique la visibilité individuelle. (C5) garantit que $JS \alpha \in V$ implique $JS \sigma \alpha \in V$ pour $\sigma \in OBS^*$.

Ensemble, les deux dernières contraintes assurent que lorsque $JS \alpha \in V$, alors $\sigma \alpha \in V$ pour $\sigma \in OBS^+$. Ceci motive la relation de *conséquence introspective* entre les atomes comme suit :

$$\alpha \rightsquigarrow \beta \text{ ssi soit } \alpha = \beta,$$

$$\text{soit } \alpha = JS \alpha' \text{ et } \beta = \sigma \alpha' \text{ pour } \sigma \in OBS^+$$

Ainsi, un atome de la forme $JS p$ a un nombre infini de conséquences : $JS p$ lui-même ; $S_i p$, $JS S_i p$ et $S_i JS p$ pour tout agent i ; $S_i S_j p$ pour tout i et j , et ainsi de suite. Les atomes qui ne sont pas de la forme $JS \alpha$ ont seulement comme conséquence eux-mêmes.

Nous pouvons caractériser évaluations introspectives comme les évaluations qui sont fermées par conséquence introspective.

Proposition 1. *Une évaluation $V \subseteq ATM$ est introspective si et seulement si, pour tout $\alpha \in ATM$ et $i \in Agt$:*

$$\sigma S_i S_i \alpha \in V \text{ pour tout } \sigma \in OBS^* \quad (1)$$

$$\sigma JS \alpha \in V \text{ pour tout } \sigma \in OBS^+ \quad (2)$$

$$si \alpha \in V \text{ et } \alpha \rightsquigarrow \beta \text{ alors } \beta \in V \quad (3)$$

Ainsi, tout atome contenant une séquence $S_i S_i$ est dans une évaluation introspective, ainsi que tout atome contenant un JS (en supposant qu'il n'est pas le premier élément). Enfin, toute conséquence est également présente. Ces trois éléments sont équivalents aux cinq contraintes (voir la preuve dans la version complète du papier), mais sont techniquement plus faciles à manipuler.

On dit qu'un atome $\alpha \in ATM$ est *valide dans $INTR$* si et seulement si α appartient à toute évaluation de $INTR$. Par la proposition 1, α est valide dans $INTR$ si et seulement s'il est de la forme soit $\sigma S_i S_i \alpha$ avec $\sigma \in OBS^*$ ou $\sigma JS \alpha$ avec $\sigma \in OBS^+$.

On observe que nous n'imposons pas la contrainte "si $\sigma \alpha \in V$ pour chaque $\sigma \in OBS^*$, alors $JS \alpha \in V$ ", qui correspond à la définition du plus grand point fixe de l'opérateur de connaissance commune. Nous commenterons ceci dans la section 3.4.

3.2 Relations d'indistinguabilité

Nous définissons les relations \sim_i entre évaluations, une par agent i . Deux évaluations sont liées par \sim_i si chaque α que i voit a la même valeur dans l'évaluation liée. De la même façon, nous définissons une relation \sim_{Agt} .

On note $V(\alpha) = V'(\alpha)$ lorsque soit $\alpha \in V$ et $\alpha \in V'$, ou $\alpha \notin V$ et $\alpha \notin V'$. On définit :

$$V \sim_i V' \text{ ssi } S_i \alpha \in V \text{ implique } V(\alpha) = V'(\alpha)$$

$$V \sim_{Agt} V' \text{ ssi } JS \alpha \in V \text{ implique } V(\alpha) = V'(\alpha)$$

Par exemple, supposons que l'ensemble des variables propositionnelles $Prop$ est le singleton $\{p\}$. Alors une évaluation contenant $S_i p$ et p est liée à toute évaluation contenant p , alors que l'évaluation avec $S_i p$ mais sans p est liée à toute évaluation ne contenant pas p .

Les relations \sim_i et \sim_{Agt} sont réflexives, mais ni transitives ni symétriques. Cependant, ces deux propriétés sont vérifiées sur les évaluations satisfaisant les contraintes (C1) et (C2) de la définition 1.

Proposition 2. *La relation \sim_{Agt} et chaque relation \sim_i sont des relations d'équivalence sur $INTR$.*

Lemma 1. *Soit $V \in INTR$, $V \sim_i V_1$ et $V \sim_{Agt} V_2$. Alors $V_1 \in INTR$ et $V_2 \in INTR$.*

Ce dernier lemme établit que, si nous sommes dans une évaluation introspective, alors une relation épistémique ne permet jamais d'accéder à une évaluation non introspective.

3.3 Conditions de vérité et validité

Étant donné une évaluation introspective V , nos opérations de mise à jour ajoutent ou enlèvent des atomes de V . Cela demande une certaine prudence : nous voulons que l'évaluation résultante soit introspective. Par exemple, quand V ne contient pas $S_i p$, alors $V \cup \{JS p\}$ viole (C4). Donc, lors de l'ajout d'un atome à V , nous devons aussi ajouter toutes ses *conséquences introspectives positives*. Symétriquement, lors du retrait d'un atome, il faut aussi de supprimer ses *conséquences introspectives négatives*. Nous définissons les éléments suivants :

$$Eff^+(\alpha) = \{\beta \in ATM : \alpha \rightsquigarrow \beta\}$$

$$Eff^-(\alpha) = \{\beta \in ATM : \beta \rightsquigarrow \alpha\}$$

Par exemple, $Eff^-(S_i S_j p) = \{JS p, JS S_j p\}$ car ces deux atomes impliquent $S_i S_j p$. Les variables propositionnelles n'ont pas de conséquence positive ou négative, à part eux-mêmes. De toute évidence, quand V est introspective alors à la fois $V \cup Eff^+(\alpha)$ et $V \setminus Eff^-(\alpha)$ le sont aussi, (sauf si α est valide).

Les conditions de vérité sont les suivantes :

$$V \models \alpha \quad \text{ssi } \alpha \in V$$

$$V \models \neg \varphi \quad \text{ssi } V \not\models \varphi$$

$$V \models \varphi \wedge \psi \quad \text{ssi } V \models \varphi \text{ et } V \models \psi$$

$$V \models K_i \varphi \quad \text{ssi } V' \models \varphi \text{ pour tout } V' \text{ tel que } V \sim_i V'$$

$$V \models CK \varphi \quad \text{ssi } V' \models \varphi \text{ pour tout } V' \text{ tel que } V \sim_{Agt} V'$$

$$V \models [\pi] \varphi \quad \text{ssi } V' \models \varphi \text{ pour tout } V' \text{ tel que } VR_\pi V'$$

où R_π est une relation binaire sur les évaluations définie (par récurrence mutuelle avec la définition de \models) par :

$$VR_{+\alpha} V' \quad \text{ssi } V' = V \cup Eff^+(\alpha)$$

$$VR_{-\alpha} V' \quad \text{ssi } V' = V \setminus Eff^-(\alpha)$$

and α n'est pas valide dans $INTR$

$$VR_{\pi_1; \pi_2} V' \quad \text{ssi il existe } U \text{ tel que}$$

$$VR_{\pi_1} U \text{ et } UR_{\pi_2} V'$$

$$VR_{\pi_1 \sqcup \pi_2} V' \quad \text{ssi } VR_{\pi_1} V' \text{ ou } VR_{\pi_2} V'$$

$$VR_{\varphi?} V' \quad \text{ssi } V = V' \text{ et } V \models \varphi$$

La relation R_π est définie comme dans PDL pour les opérateurs $;$, \sqcup et $?$. L'interprétation des affectations est conçu de telle manière que nous restons dans $INTR$: lors de l'exécution $+\alpha$, on ajoute toutes les

conséquences positives de α ; pour $-\alpha$, le programme devrait échouer si α est valide dans $INTR$ et sinon, enlever toutes les conséquences négatives de α . Par exemple, nous n'avons jamais $VR_{-S_1 S_1 p} V'$, i.e., le programme $-S_1 S_1 p$ échoue toujours. En revanche, le programme $-S_1 S_2 p$ réussit toujours, et on a

$$VR_{-S_1 S_2 p} (V \setminus \{S_1 S_2 p, JS S_2 p, JS p\}).$$

car $JS S_2 p \rightsquigarrow S_1 S_2 p$ et $JS p \rightsquigarrow S_1 S_2 p$. Donc $V \Vdash [-S_1 S_2 p] JS p$ quel que soit V .

Lemma 2. *Si $V \in INTR$ et $VR_{\pi} V'$ alors $V' \in INTR$.*

Proposition 3. *Pour tout $V \in INTR$, $i \in Agt$ et π , V est seulement liée à des évaluations de $INTR$ par \sim_i , \sim_{Agt} et R_{π} .*

Lorsqu'on a $V \models \varphi$, on dit que V est un *modèle* de φ . L'ensemble des modèles (pas nécessairement introspectifs) de φ est noté $\|\varphi\|$.

Une formule φ est *satisfiable dans $INTR$* si φ a un modèle introspectif, à savoir, si $\|\varphi\| \cap INTR \neq \emptyset$. Par exemple, $JS p \wedge \neg S_i p$ a un modèle, mais n'a pas de modèle introspectif et est donc insatisfiable dans $INTR$.

Une formule φ est *valide dans $INTR$* si $INTR \subseteq \|\varphi\|$. On dit aussi que φ est DEL-PAO-valide. Par exemple, $\neg[-S_1 S_2 p] JS p$ est valide dans $INTR$. On note que $\neg\beta \rightarrow [+ \alpha] \neg\beta$ est valide dans $INTR$ si et seulement si $\alpha \not\rightsquigarrow \beta$.

La proposition suivante exprime le fait que, quand φ ne contient aucun opérateur épistémique, alors la valeur de vérité de φ ne dépend que des atomes présents dans φ .

Proposition 4. *Soit φ sans opérateurs épistémiques. Soit $V, V' \in 2^{ATM}$ telles que $V(\alpha) = V'(\alpha)$ pour chaque atome $\alpha \in ATM(\varphi)$. Alors $V \models \varphi$ si et seulement si $V' \models \varphi$.*

Cette proposition sera déterminante pour le reste du papier. On remarque qu'elle n'est pas nécessairement correcte quand la formule contient des opérateurs épistémiques. Par exemple, considérons $K_i p$: sa valeur de vérité dépend de la valeur de vérité de l'atome de visibilité $S_i p$, qui ne fait pas partie $ATM(K_i p)$.

3.4 Discussion

Les opérateurs de la connaissance individuelle de DEL-PAO satisfont tous les principes de S5. Ils sont plus forts que ceux de la logique épistémique standard S5. On peut le voir avec la formule invalide de S5 $K_i(p \vee q) \rightarrow (K_i p \vee K_i q)$ qui est valide dans $INTR$ (voir la proposition 5). Pour donner un exemple, cela signifie que si l'on sait que le maître d'hôtel ou le jardinier est le meurtrier alors on sait lequel d'entre eux l'est ; elle implique aussi que l'on ne peut pas connaître une disjonction exclusive de deux variables propositionnelles

p et q . C'est un principe fort, qui est toutefois également validé par toutes les autres logiques épistémiques basées sur la visibilité.

Notre opérateur de connaissance commune satisfait l'axiome de point fixe de la connaissance commune

$$CKp \rightarrow p \wedge \left(\bigwedge_{i \in Agt} K_i CKp \right).$$

Cela correspond au fait que par des contraintes (C2) et (C4), $JS p \wedge p \rightarrow \left(\bigwedge_{i \in Agt} S_i JS p \right)$ est valide dans $INTR$.

Notre notion de connaissance commune est toutefois plus faible que la connaissance commune standard car l'axiome d'induction

$$\left(\varphi \wedge CK \left(\varphi \rightarrow \bigwedge_{i \in Agt} K_i \varphi \right) \right) \rightarrow CK \varphi$$

n'est pas valide dans $INTR$. Au-delà de la justification technique de ce choix (une telle contrainte ne peut pas être capturée par une formule finie constituée d'atomes de visibilité) nous nous alignons sur [14, 9] et supposons que ce principe est trop fort pour une logique de la connaissance commune.

4 Axiomatization

Nous présentons maintenant une axiomatisation adéquate et complète de DEL-PAO.

4.1 Axiomes de réduction pour les opérateurs modaux

La proposition suivante sera utilisée pour éliminer les opérateurs épistémiques d'une formule qui ne contient pas d'opérateurs dynamiques.

Proposition 5. *Les formules suivantes sont DEL-PAO-valides, avec A^+ et A^- des ensembles d'atomes tels que $A^+ \cap A^- = \emptyset$:*

$$K_i \alpha \leftrightarrow S_i \alpha \wedge \alpha \quad (Red_{K, \alpha})$$

$$CK \alpha \leftrightarrow JS \alpha \wedge \alpha \quad (Red_{CK, \alpha})$$

$$K_i \neg \alpha \leftrightarrow S_i \alpha \wedge \neg \alpha \quad (Red_{K, \neg})$$

$$CK \neg \alpha \leftrightarrow JS \alpha \wedge \neg \alpha \quad (Red_{CK, \neg})$$

$$K_i(\varphi \wedge \varphi') \leftrightarrow K_i \varphi \wedge K_i \varphi' \quad (Red_{K, \wedge})$$

$$CK(\varphi \wedge \varphi') \leftrightarrow CK \varphi \wedge CK \varphi' \quad (Red_{CK, \wedge})$$

$$K_i \left(\bigvee_{\alpha \in A^+} \alpha \vee \bigvee_{\alpha \in A^-} \neg \alpha \right) \leftrightarrow \left(\bigvee_{\alpha \in A^+} K_i \alpha \right) \vee \left(\bigvee_{\alpha \in A^-} K_i \neg \alpha \right) \quad (Red_{K, \vee})$$

$$CK \left(\bigvee_{\alpha \in A^+} \alpha \vee \bigvee_{\alpha \in A^-} \neg \alpha \right) \leftrightarrow \left(\bigvee_{\alpha \in A^+} CK \alpha \right) \vee \left(\bigvee_{\alpha \in A^-} CK \neg \alpha \right) \quad (Red_{CK, \vee})$$

La proposition suivante permet d'éliminer les opérateurs dynamiques de formules ne contenant pas d'opérateurs épistémiques.

Proposition 6. *Les formules suivantes sont DEL-PAO-valides.*

$$\begin{aligned}
 [\pi; \pi']\varphi &\leftrightarrow [\pi][\pi']\varphi && (\text{Red}_;) \\
 [\pi \sqcup \pi']\varphi &\leftrightarrow [\pi]\varphi \wedge [\pi']\varphi && (\text{Red}_{\sqcup}) \\
 [\varphi?]\varphi' &\leftrightarrow \varphi \rightarrow \varphi' && (\text{Red}_?) \\
 [+ \alpha]\neg\varphi &\leftrightarrow \neg[+ \alpha]\varphi && (\text{Red}_{+ \alpha, \neg}) \\
 [- \alpha]\neg\varphi &\leftrightarrow \begin{cases} \top & \text{si } \alpha \text{ est valide dans INTR} \\ \neg[- \alpha]\varphi & \text{sinon} \end{cases} && (\text{Red}_{- \alpha, \neg}) \\
 [+ \alpha](\varphi \wedge \varphi') &\leftrightarrow [+ \alpha]\varphi \wedge [+ \alpha]\varphi' && (\text{Red}_{+ \alpha, \wedge}) \\
 [- \alpha](\varphi \wedge \varphi') &\leftrightarrow [- \alpha]\varphi \wedge [- \alpha]\varphi' && (\text{Red}_{- \alpha, \wedge}) \\
 [+ \alpha]\beta &\leftrightarrow \begin{cases} \top & \text{si } \alpha \rightsquigarrow \beta \\ \beta & \text{sinon} \end{cases} && (\text{Red}_{+ \alpha}) \\
 [- \alpha]\beta &\leftrightarrow \begin{cases} \top & \text{si } \alpha \text{ est valide dans INTR} \\ \perp & \text{si } \alpha \text{ n'est pas valide dans INTR} \\ & \text{and } \beta \rightsquigarrow \alpha \\ \beta & \text{sinon} \end{cases} && (\text{Red}_{- \alpha})
 \end{aligned}$$

Theorem 1. *Pour toute formule φ , il existe une formule sans opérateurs modaux φ' telle que $\varphi \leftrightarrow \varphi'$ est DEL-PAO-valide.*

4.2 Axiomes d'introspection

Proposition 7. *Les formules suivantes sont DEL-PAO-valides, pour tout $i \in \text{Agt}$ et $\alpha \in \text{ATM}$:*

$$\begin{aligned}
 S_i S_i \alpha &&& (\text{Vis}_{C1}) \\
 JS JS \alpha &&& (\text{Vis}_{C2}) \\
 JS S_i S_i \alpha &&& (\text{Vis}_{C3}) \\
 JS \alpha \rightarrow S_i \alpha &&& (\text{Vis}_{C4}) \\
 JS \alpha \rightarrow JS S_i \alpha &&& (\text{Vis}_{C5})
 \end{aligned}$$

On appelle \mathcal{T}_{vis} la collection des formules ci-dessus, i.e.,

$$\begin{aligned}
 \mathcal{T}_{vis} = & \{S_i S_i \alpha : i \in \text{Agt}, \alpha \in \text{ATM}\} \\
 & \cup \{JS JS \alpha : \alpha \in \text{ATM}\} \\
 & \cup \{JS S_i S_i \alpha : i \in \text{Agt}, \alpha \in \text{ATM}\} \\
 & \cup \{JS \alpha \rightarrow S_i \alpha : i \in \text{Agt}, \alpha \in \text{ATM}\} \\
 & \cup \{JS \alpha \rightarrow JS S_i \alpha : i \in \text{Agt}, \alpha \in \text{ATM}\}
 \end{aligned}$$

Proposition 8. *Pour φ sans opérateurs modaux, φ est DEL-PAO-valide si et seulement si $\mathcal{T}_{vis} \models_{\text{CPL}} \varphi$, où \models_{CPL} est la conséquence logique en logique propositionnelle classique.*

4.3 Adéquation et complétude

L'axiomatisation de DEL-PAO est donnée par :

- les axiomes de CPL (logique propositionnelle classique) ;

- les axiomes de réduction de la proposition 5 ;
- les axiomes de réduction de la proposition 6 ;
- les axiomes d'introspection de la proposition 7 ;
- la règle du Modus Ponens et les règles d'équivalence pour K_i , CK et $[\pi]$:

$$\begin{aligned}
 & \frac{\varphi \leftrightarrow \varphi'}{K_i \varphi \leftrightarrow K_i \varphi'} \\
 & \frac{K_i \varphi \leftrightarrow K_i \varphi'}{\varphi \leftrightarrow \varphi'} \\
 & \frac{CK \varphi \leftrightarrow CK \varphi'}{\varphi \leftrightarrow \varphi'} \\
 & \frac{\varphi \leftrightarrow \varphi'}{[\pi]\varphi \leftrightarrow [\pi]\varphi'}
 \end{aligned}$$

Proposition 9. *Pour les formules φ sans opérateurs modaux, $\vdash_{\text{DEL-PAO}} \varphi$ si et seulement si $\mathcal{T}_{vis} \vdash_{\text{CPL}} \varphi$.*

Theorem 2. *L'axiomatisation de DEL-PAO est adéquate et complète.*

5 Complexité

Theorem 3. *Les problèmes de satisfiabilité et de vérification de modèle de DEL-PAO sont PSPACE-complets.*

Nous consacrons le reste de la section à la preuve de ce résultat. Nous commençons par observer que les problèmes de satisfiabilité et de vérification de modèle ont la même complexité.

Proposition 10. *Soit φ une formule de DEL-PAO sans opérateurs épistémiques telle que $\text{ATM}(\varphi) = \{\alpha_1, \dots, \alpha_n\}$. Alors φ est satisfiable dans INTR si et seulement si la formule*

$$\langle (+\alpha_1 \sqcup -\alpha_1); \dots; (+\alpha_n \sqcup -\alpha_n) \rangle \varphi$$

est valide dans INTR.

On voit que $(+\alpha_1 \sqcup -\alpha_1); \dots; (+\alpha_n \sqcup -\alpha_n)$ a une longueur linéaire dans $\text{length}(\varphi)$ car $\text{ATM}(\varphi) \leq \text{length}(\varphi)$.

Dans le reste de la section, nous établissons les bornes inférieures et supérieures du problème de vérification de modèle.

5.1 Borne inférieure

La borne inférieure du problème de vérification de modèle est établie au moyen d'un encodage de formules booléennes quantifiées (QBF), dont le problème de satisfiabilité est PSPACE-complet. Plus de détails sont disponibles dans la version complète du papier.

Il est un peu plus difficile d'établir la borne supérieure. Nous allons encoder notre logique dans le fragment sans étoile (*) de Dynamic Logic of Propositional Assignments DL-PA [10, 3], dont le problème de satisfiabilité est PSPACE-complet. D'abord, nous rappelons brièvement que cette logique.

5.2 Dynamic Logic of Propositional Assignments

Tout comme le langage de DEL-PAO, le langage de DL-PA est constitué de formules et de programmes. Ils sont définis par la grammaire suivante :

$$\begin{aligned}\pi &::= +\alpha \mid -\alpha \mid \pi; \pi \mid \pi \sqcup \pi \mid \varphi? \\ \varphi &::= \alpha \mid \neg\varphi \mid \varphi \wedge \varphi \mid [\pi]\varphi\end{aligned}$$

où α prend ses valeurs dans ATM et i dans Agt . Ce langage ne contient pas d'opérateurs épistémiques, et est constitué des mêmes atomes que DEL-PAO.

Les formules sont interprétées dans des évaluations $V \in 2^{ATM}$ exactement de la même manière que dans DEL-PAO, sauf que l'interprétation des programmes atomiques ne prend pas en compte les conséquences introspectives. Donc :

$$\begin{aligned}VR_{+\alpha}V' &\text{ ssi } V' = V \cup \{p\} \\ VR_{-\alpha}V' &\text{ ssi } V' = V \setminus \{p\}\end{aligned}$$

On a un équivalent de la proposition 4 dans DL-PA.

Proposition 11 ([3], Proposition 1). *Soit $V, V' \in 2^{ATM}$ telles que $V(\alpha) = V'(\alpha)$ pour tout atome $\alpha \in ATM(\varphi)$. Alors $V \models_{DL-PA} \varphi$ si et seulement si $V' \models_{DL-PA} \varphi$.*

5.3 Borne supérieure

Nous transformons d'abord nos opérateurs épistémiques en programmes équivalents ; il suffira alors de donner une borne supérieure pour le problème de vérification de modèle pour les formules sans opérateurs épistémiques. Nous établissons ensuite la complexité en traduisant nos formules et programmes en formules et programmes DL-PA. Ce faisant, nous prenons en compte les contraintes d'introspection de DEL-PAO sur les évaluations.

Tout d'abord, nous définissons les programmes :

$$\begin{aligned}\pi_{i,\alpha} &= S_i \alpha? \sqcup (\neg S_i \alpha?; (+\alpha \sqcup -\alpha)) \\ \pi_{Agt,\alpha} &= JS \alpha? \sqcup (\neg JS \alpha?; (+\alpha \sqcup -\alpha))\end{aligned}$$

Le premier programme vérifie si i voit α , et si ce n'est pas le cas, fait varier la valeur de vérité de α ; le second fait la même chose mais pour la visibilité conjointe. Étant donné un ensemble d'atomes $A = \{\alpha_1, \dots, \alpha_n\}$, on définit :

$$\begin{aligned}\pi_{i,A} &= \pi_{i,\alpha_1}; \dots; \pi_{i,\alpha_n} \\ \pi_{Agt,A} &= \pi_{Agt,\alpha_1}; \dots; \pi_{Agt,\alpha_n}\end{aligned}$$

où nous supposons que les atomes sont ordonnés de telle sorte que $k \leq l$ implique que $length(\alpha_k) \leq length(\alpha_l)$. De plus, nous supposons que le programme correspond à *skip* si l'ensemble A est vide. Par exemple,

$$\pi_{i,\{S_j p, p\}} = (S_i p? \sqcup (\neg S_i p?; (+p \sqcup -p)));$$

$$(S_i S_j p? \sqcup (\neg S_i S_j p?; (+S_j p \sqcup -S_j p)))$$

car $length(p) \leq length(S_j p)$.

Proposition 12. *Soit φ une formule de DEL-PAO sans opérateurs épistémiques. Alors les équivalences*

$$\begin{aligned}K_i \varphi &\leftrightarrow [\pi_{i,ATM(\varphi)}]\varphi \\ CK \varphi &\leftrightarrow [\pi_{Agt,ATM(\varphi)}]\varphi\end{aligned}$$

sont DEL-PAO-valides.

La proposition 12 peut être transformée en une procédure éliminant opérateurs épistémiques : il suffit d'appliquer plusieurs fois les équivalences, à commencer par les opérateurs les plus intérieurs.

Procédure 1. *Tant qu'il y a un opérateur épistémique dans φ :*

1. *s'il existe une sous-formule $K_i \varphi'$ telle que φ' ne contient pas les opérateurs épistémiques, remplacer φ par $[\pi_{i,ATM(\varphi)}]\varphi'$;*
2. *s'il existe une sous-formule $CK \varphi'$ telle que φ' ne contient pas les opérateurs épistémiques, remplacer φ par $[\pi_{Agt,ATM(\varphi)}]\varphi'$.*

Proposition 13. *Pour chaque formule DEL-PAO φ , il existe une formule DEL-PAO φ' sans opérateurs épistémiques telle que $\varphi \leftrightarrow \varphi'$ est DEL-PAO-valide. La longueur de φ' est polynomiale dans $length(\varphi)$.*

Il suffit donc de considérer le problème de vérification de modèle pour les formules sans opérateurs épistémiques. La prochaine étape est de traduire en temps polynomial les formules et les programmes de DEL-PAO en formules et programmes DL-PA. Nous allons en même temps capter les contraintes d'introspection par des affectations DL-PA.

Étant donné d'un atome α et un ensemble d'atomes pertinents $A \subseteq ATM$, soit $Eff^+(\alpha) \cap A = \{\beta_1, \dots, \beta_n\}$ et $Eff^-(\alpha) \cap A = \{\beta'_1, \dots, \beta'_m\}$. On traduit les affectations de α comme suit :

$$tr(+\alpha, A) = +\beta_1; \dots; +\beta_n$$

$$tr(-\alpha, A) = \begin{cases} fail & \text{si } \alpha \text{ est valide dans } INTR \\ -\beta'_1; \dots; -\beta'_m & \text{sinon} \end{cases}$$

Encore une fois, nous supposons que les atomes sont ordonnés de telle sorte que $k \leq l$ implique que $length(\beta_k) \leq length(\beta_l)$ et que le programme devient *skip* si l'ensemble $\{\beta_1, \dots, \beta_n\}$ est vide.

Nous étendons tr de façon homomorphique aux programmes et formules complexes, avec $tr(\alpha) = \alpha$ et

$$tr([\pi]\varphi) = [tr(\pi, ATM(\varphi))]tr(\varphi).$$

On note que $ATM(tr(\pi, A)) \subseteq A$ et $ATM(tr(\varphi)) \subseteq ATM(\varphi)$.

Proposition 14. *Soit φ une formule DEL-PAO sans opérateurs épistémiques. Alors $V \models_{DL-PA} \varphi$ si et seulement si $V \models_{DL-PA} tr(\varphi)$.*

La grande finale résulte de propositions 10, 13 et 14 et de l'observation que la longueur de $tr(\varphi)$ est polynomiale dans $length(\varphi)$.

Corollary 1. *Dans DEL-PAO, les problèmes de satisfiabilité et de vérification de modèle sont PSPACE-complet.*

6 Applications

Nous montrons maintenant comment exprimer les annonces publiques et privées des atomes et des formules épistémiques. Cela nous permet de formaliser le problème de bavardage et des deux généraux.

6.1 Annonces publiques et privées

Public Announcement Logic PAL [15] est une logique de la famille DEL étendant la logique épistémique standard avec un opérateur $[\psi!]$, de sorte que $[\psi!]\varphi$ signifie "après que le fait réel ψ ait été annoncé publiquement, φ est vrai". Ses validités sont axiomatisées au moyen des axiomes de réduction suivants :

$$\begin{aligned} [\psi!]p &\leftrightarrow \psi \rightarrow p \\ [\psi!]\neg\varphi &\leftrightarrow \psi \rightarrow \neg[\psi!]\varphi \\ [\psi!](\varphi \wedge \varphi') &\leftrightarrow [\psi!]\varphi \wedge [\psi!]\varphi' \\ [\psi!]K_i\varphi &\leftrightarrow \psi \rightarrow K_i[\psi!]\varphi \end{aligned}$$

Nous pouvons exprimer les annonces publiques de littéraux dans DEL-PAO comme suit :

$$\begin{aligned} p! &= p?; +JS p \\ \neg p! &= \neg p?; +JS p \end{aligned}$$

ainsi que celle de la connaissance des atomes :

$$K_i p! = K_i p?; +JS p$$

Il peut être vérifié qu'avec ces définitions, tous les axiomes de réduction de PAL sont valides dans notre logique lorsque ψ est soit un littéral soit la connaissance d'un atome (voir la version longue).

Un supplément intéressant est le fait que nous pouvons également modéliser facilement les annonces *privées* du même type de formules. Avec $j : \psi!$ signifiant " ψ est annoncé en privé à l'agent j ", on a :

$$\begin{aligned} j : p! &= p?; +S_j p \\ j : \neg p! &= \neg p?; +S_j p \\ j : K_i p! &= K_i p?; +S_j p; +S_j S_i p \end{aligned}$$

6.2 Le problème de bavardage

Six amis ont chacun un secret. Quand ils s'appellent les uns des autres, ils échangent tous les secrets qu'ils connaissent. Le problème est de trouver combien d'appels sont nécessaires pour diffuser tous les secrets

parmi tous les amis. Il a été prouvé [12] que le nombre minimal d'appels est 8 ; par exemple, si nous écrivons ij le fait que i appelle j (ou j appelle i), la séquence suivante propage tous les secrets : 12, 34, 56, 13, 45, 16, 24, 35. Une étude détaillée de ce problème peut être trouvée dans [18]. Le problème est également abordé dans [19] avec une logique spécifique aux réseaux de communication.

Nous modélisons ce problème avec des annonces privées. Avec $Agt = \{i : 1 \leq i \leq 6\}$ et $Prop = \{s_i : i \in Agt\}$ (s_i signifiant que i a le secret s_i), on définit le programme $Call_{ij}$, pour $i, j \in Agt$, par :

$$\begin{aligned} Call_{ij} = & \\ & (S_i s_1?; j : s_1! \sqcup \neg S_i s_1?); \dots; (S_i s_6?; j : s_6! \sqcup \neg S_i s_6?); \\ & (S_j s_1?; i : s_1! \sqcup \neg S_j s_1?); \dots; (S_j s_6?; i : s_6! \sqcup \neg S_j s_6?) \end{aligned}$$

En exécutant notre programme, i raconte tout ce qu'il sait à j , et inversement. Par conséquent, on a :

$$\begin{aligned} \{\alpha : \alpha \text{ est valide dans } INTR\} \cup Prop \cup \{S_i s_i : i \in Agt\} \models & \\ Call_{12}; Call_{34}; Call_{56}; Call_{13}; & \\ Call_{45}; Call_{16}; Call_{24}; Call_{35} \bigwedge_{i \in Agt} K_i \left(\bigwedge_{j \in Agt} s_j \right) & \end{aligned}$$

Notre évaluation contient tous les secrets, et le fait que chaque agent connaît son secret. Elle est introspective, puisque nous avons tous les atomes valides et nous n'ajoutons pas d'autre atome de la forme $JS \alpha$.

Intuitivement, chaque appel va ajouter la visibilité de chaque atome connu par l'un des deux agents aux deux ; une annonce de la forme $j : s_i!$ est égale, dans DEL-PAO, à $s_i?; +S_j s_i$, et puisque s_i est vrai dans notre évaluation, cela va se réduire à $+S_j s_i$. En fin de compte, $S_j s_i$ sera mis à vrai seulement si $S_i s_i$ est vrai. Donc $Call_{12}$ est réduit à $+S_2 s_1; +S_1 s_2$, et après son exécution, $S_1 s_1, S_2 s_2, S_1 s_2$ et $S_2 s_1$ sont tous vrais, et donc $K_1(s_1 \wedge s_2) \wedge K_2(s_1 \wedge s_2)$ est également vrai, ce qui est le résultat attendu. En continuant ainsi, et puisque cette séquence d'appels conduit à une solution dans le problème initial, la formule sera vraie à la fin du programme.

6.3 Le problème des deux généraux

Nous avons vu que dans le problème de bavardage, chaque ami finira par connaître chaque secret. Cela ne signifie pas toutefois qu'il y a connaissance commune de tous les secrets. La difficulté d'obtenir la connaissance commune par communication asynchrone non fiable est mis en évidence par le 'problème des deux généraux' [1] [7]. Sa généralisation est le célèbre problème de généraux byzantins [13].

En bref, deux généraux ont besoin de coordonner une attaque ; ils ne peuvent gagner que s'ils attaquent en même temps. Pour communiquer, ils doivent en-

voyer un messenger qui peut être capturé. Ils doivent décider du moment de l'attaque, se mettre d'accord sur ce moment, et chaque général doit savoir que l'autre est d'accord. Pour ce faire, un général envoie un message avec l'heure de l'attaque, mais comme il ne peut pas être sûr que le message ait été reçu, l'autre général doit envoyer un accusé de réception. Bien sûr, tout comme le message d'origine, l'accusé de réception peut être perdu, le premier général doit donc envoyer un accusé de réception, et ainsi de suite. Pour être parfaitement coordonnés, à savoir, pour garantir qu'il y a connaissance commune de l'heure de l'attaque, les généraux doivent envoyer un nombre infini de messages.

Formellement, avec $Agt = \{1, 2\}$ et $Prop = \{ta\}$, nous définissons l'atome α_n par :

- $\alpha_n = \underbrace{S_1 S_2 S_1 \dots S_2}_{n \text{ alternations}} ta$ si n est pair ;
- $\alpha_n = \underbrace{S_2 S_1 S_2 \dots S_2}_{n \text{ alternations}} ta$ si n est impair.

Ensuite, nous définissons le programme

$$\pi_n = +ta; +S_1 ta; +S_2 ta; +\alpha_2; +\alpha_3; \dots; +\alpha_n,$$

qui formalise le fait que le moment de l'attaque a été défini, que chaque agent le sait, que 1 sait que 2 le sait, et ainsi de suite jusqu'à n .

L'illustration du problème de deux généraux est alors le suivant. Pour chaque $n > 0$,

$$\{\alpha : \alpha \text{ est valide dans } INTR\} \models [\pi_n] \neg CK ta.$$

Intuitivement, pour avoir $V \models CK ta$, il faut avoir $ta \in V$ et $JS ta \in V$. Comme aucun d'entre eux est présent dans notre évaluation, les deux devraient être ajoutés par le programme π_n . En outre, et parce que $JS ta$ est une conséquence introspective de lui-même seulement, il devrait être ajouté explicitement par π_n . Cependant, alors que π_n ajoute ta quelle que soit la valeur de $n > 0$, l'atome $JS ta$ n'est jamais ajouté. Ainsi, $CK ta$ n'est jamais vrai quel que soit $n > 0$.

7 Travaux connexes

Ce travail est lié à plusieurs autres logiques développées au cours des dernières années. La logique *Epistemic Coalition Logic of Propositional Control with Partial Observability* ECL-PC(PO) [17] utilise le langage de la logique épistémique standard ; mais les formules sont interprétées sur des évaluations. La principale différence avec DEL-PAO est que la visibilité est représentée par un ensemble d'atomes pour chaque agent, contenant les variables dont l'agent peut percevoir la valeur. Par conséquent, ils n'est pas possible d'exprimer des perceptions d'ordre supérieur telles que "l'agent i voit si l'agent j perçoit p ". Au lieu de cela, et comme cela a déjà été mentionné, la perception de chaque agent devient connaissance commune parmi tous les agents. Par conséquent, nous conjecturons que ECL-PC(PO)

peut être encodé dans DEL-PAO.

Une autre logique proche de la nôtre est la *Logic of Revelation and Concealment* LRC [16]. Cette logique permet d'exprimer, par des programmes, qu'une variable est révélée ou cachée à un agent. Sémantiquement, les formules sont interprétées dans des modèles pointés avec un ensemble de visibilité pour chaque agent. Révéler une variable à un agent l'ajoute à son ensemble, alors que la lui cacher la supprime de son ensemble. Cependant, comme dans [17], les ensembles de visibilité sont connaissance commune parmi tous les agents.

Nous pouvons également citer la logique 'knowing whether' [5] qui ajoute un opérateur signifiant " i sait si φ " au langage de la logique épistémique standard, interprété comme " φ a la même valeur dans tous les mondes indistinguables pour i ". Cela peut être comparé à nos atomes de visibilité S_i qui expriment la même idée sur les atomes.

8 Conclusion

Nous avons introduit une logique épistémique dynamique avec affectations et observations DEL-PAO qui permet d'exprimer des observations d'ordre supérieur et conjointes ainsi que leurs mises à jour. Elle évite l'hypothèse forte des autres logiques épistémiques basées sur les observations, où qui voit quoi qui est connaissance commune. On remarque que l'ajout de l'ordre supérieur des observations et en particulier des observations communes se fait sans coût supplémentaire : la satisfiabilité et la vérification de modèle restent PSPACE-complet. Cela contraste avec les logiques standards de connaissance commune où la satisfiabilité est EXPTIME-difficile [8].

Une extension simple de notre logique serait de généraliser l'opérateur de connaissance commune CK à des sous-ensembles de Agt . Il faudrait ajouter des atomes de visibilité $JS_j \alpha$, un par groupe d'agents J . Une autre généralisation intéressante serait de considérer la croyance à la place de la connaissance. Un moyen d'y parvenir pourrait être de remplacer S_i par deux opérateurs O_i et C_i , signifiant respectivement que i a une opinion sur quelque chose et que l'opinion de i est correcte sur quelque chose. Cela nécessiterait d'autres contraintes sur les évaluations qui doivent correspondre aux propriétés de la croyance. D'autres extensions possibles concernent la partie dynamique : comme [10], on peut ajouter des atomes représentant le contrôle de i sur une variable propositionnelle p , dans le sens où i peut changer la valeur de vérité de p à volonté. On peut alors associer à chaque affectation un auteur, qui est l'agent exécutant l'affectation. Comme montré dans [10], cela permet d'intégrer la Coalition

Logic of Propositional Control [17]. Il reste à savoir comment ceci se combine avec les observations d'ordre supérieur.

Remerciements

Nous sommes reconnaissants aux relecteurs pour leur lecture approfondie et leurs commentaires.

Références

- [1] E. A. Akkoyunlu, K. Ekanadham, and R. V. Hurbert. Some constraints and tradeoffs in the design of network communications. In *Proceedings of the 5th ACM Symposium on Operating Systems Principles*, pages 67–74. ACM Press, 1975.
- [2] Rajeev Alur, Thomas A. Henzinger, Freddy Y. C. Mang, Shaz Qadeer, Sriram K. Rajamani, and Serdar Tasiran. MOCHA : modularity in model checking. In Alan J. Hu and Moshe Y. Vardi, editors, *Computer Aided Verification, 10th International Conference, CAV '98, Vancouver, BC, Canada, June 28 - July 2, 1998, Proceedings*, volume 1427 of *Lecture Notes in Computer Science*, pages 521–525. Springer, 1998.
- [3] Philippe Balbiani, Andreas Herzig, and Nicolas Troquard. Dynamic logic of propositional assignments : a well-behaved variant of PDL. In Orna Kupferman, editor, *Proceedings of the 28th Annual IEEE/ACM Symposium on Logic in Computer Science*, pages 143–152, 2013.
- [4] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning about Knowledge*. MIT Press, 1995.
- [5] Jie Fan, Yanjing Wang, and Hans van Ditmarsch. Knowing whether. *CoRR*, abs/1312.0, 2013.
- [6] Michael J. Fischer and Richard E. Ladner. Propositional dynamic logic of regular programs. *Journal of Computer and System Sciences*, 18(2) :194–211, 1979.
- [7] Jim Gray. Notes on data base operating systems. In *Operating Systems, An Advanced Course*, pages 393–481. Springer-Verlag, 1978.
- [8] Joseph Y. Halpern and Yoram Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54(3) :319–379, 1992.
- [9] Andreas Herzig. Logics of knowledge and action : critical analysis and challenges. *Journal of Autonomous Agents and Multi-Agent Systems*, pages 1–35, 2014. Online July 2, 2014, doi : 10.1007/s10458-014-9267-z.
- [10] Andreas Herzig, Emiliano Lorini, Nicolas Troquard, and Frédéric Moisan. A dynamic logic of normative systems. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 228–233, 2011.
- [11] Jaakko Hintikka. *Knowledge and belief*. Cornell University Press, Ithaca, N.Y., 1962.
- [12] Cor A. J. Hurkens. Spreading gossip efficiently. *Nieuw Archief voor Wiskunde*, 5/1(2) :208–210, 2000.
- [13] Leslie Lamport, Robert Shostak, and Marshall Pease. The Byzantine Generals Problem. *ACM Transactions on Programming Languages and Systems*, 4(3) :382–401, 1982.
- [14] Emiliano Lorini and Andreas Herzig. Direct and indirect common belief. In *Institutions, Emotions, and Group Agents*, pages 355–372. Springer Netherlands, 2014.
- [15] Jan Plaza. Logics of public communications. In M. L. Emrich, M. S. Pfeifer, M. Hadzikadic, and Z.W. Ras, editors, *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems*, pages 201–216. Oak Ridge National Laboratory, ORNL/DSRD- 24, 1989.
- [16] Wiebe van der Hoek, Petar Iliev, and Michael Wooldridge. A logic of revelation and concealment. In Wiebe van der Hoek, Lin Padgham, Vincent Conitzer, and Michael Winikoff, editors, *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems*, pages 1115–1122. IFAAMAS, 2012.
- [17] Wiebe van der Hoek, Nicolas Troquard, and Michael Wooldridge. Knowledge and control. In Liz Sonenberg, Peter Stone, Kagan Tumer, and Pinar Yolum, editors, *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems*, pages 719–726. IFAAMAS, 2011.
- [18] Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi. *Dynamic Epistemic Logic*. Springer Publishing Company, Incorporated, 1st edition, 2007.
- [19] Yanjing Wang, Floor Sietsma, and Jan van Eijck. Logic of information flow on communication channels. In Andrea Omicini, Sebastian Sardina, and Wamberto Weber Vasconcelos, editors, *DALT*, pages 130–147. Springer, 2010.

Raisonnement analogique pour la recommandation : premières expérimentations

Nicolas Hug Henri Prade Gilles Richard

Institut de Recherche en Informatique de Toulouse, 31400 Toulouse
{nicolas.hug, henri.prade, gilles.richard}@irit.fr

Résumé

Les systèmes de recommandation ont pour vocation de fournir des suggestions intéressantes à leurs utilisateurs. On distingue deux principaux types d'approche pour construire un système de recommandation : les méthodes dites "*content-based*" et les méthodes dites *collaboratives*. Encouragés par les bons résultats obtenus par les techniques de classification basées sur les proportions analogiques, nous nous intéressons ici à la conception de techniques de recommandation elles aussi basées sur les proportions analogiques. La qualité d'un système de recommandation peut être évaluée selon différents aspects, parmi lesquels la *précision*, qui mesure la capacité d'un système à évaluer avec justesse l'intérêt d'un utilisateur pour un objet donné. D'autres dimensions sont aussi à prendre en compte, notamment la *couverture*, ou la capacité du système à faire découvrir des articles à la fois intéressants et qui n'auraient pas pu être découverts autrement (*surprise*). Dans ce papier, nous comparons deux approches basées sur l'analogie avec d'autres approches classiques du problème de la recommandation.

Abstract

Recommender systems aim at providing suggestions of interest for end-users. Two basic types of approach underlie existing recommender systems, namely content-based methods and collaborative filtering. In this paper, encouraged by the good results obtained with analogical proportion-based classification techniques, we investigate the possibility of using analogy as the main underlying principle for implementing prediction algorithms. We study two variants of this idea. The quality of a recommender system can be estimated along diverse dimensions, among which the accuracy to predict user's rating for unseen items is clearly important. Still other dimensions like *coverage* and *surprise* are also of great interest. In this paper, we compare the proposed approach with well-known recommender systems.

1 Introduction

Dans un monde où le volume de données disponibles ne cesse de croître, les outils de filtrage automatique sont essentiels pour extraire efficacement de l'information et des connaissances. Dans le domaine du commerce électronique, les systèmes de recommandation (SR) jouent le rôle de moteur de recherche : ils filtrent les articles disponibles pour fournir aux clients des recommandations intéressantes. Avec l'essor du commerce en ligne, les systèmes de recommandation deviennent de plus en plus populaires [24, 1]. Ils aident à répondre à des questions aussi diverses que "quel film louer ?", "quel article acheter ?", ou encore "dans quel restaurant dîner ?". Pour ce faire, les SR fournissent aux utilisateurs une liste de recommandations. Evidemment, le système est d'autant meilleur que les recommandations sont personnalisées : un SR ne recommandant que les articles les plus populaires est vraisemblablement inutile, puisque les utilisateurs sont susceptibles de déjà connaître ces articles.

Un moyen très classique de fournir des recommandations pour un utilisateur cible est d'estimer son intérêt (ou son goût) pour les articles dont le système dispose. Le goût d'un utilisateur u pour un article i (i pour l'anglais *item*) est communément représenté par une note (ou score) que u donnerait à i . L'échelle des notes peut varier, mais l'intervalle entier $[1, 5]$ et l'échelle binaire *aime/n'aime pas* sont très communs dans les systèmes réels.

Une fois que ces estimations sont calculées, une option simple est de recommander à l'utilisateur les articles pour lesquels l'estimation de sa note est la plus haute. On utilise ici l'hypothèse implicite que l'utilisateur sera intéressé par les articles avec un score élevé.

Avoir une idée précise de la qualité globale d'un SR n'est pas une chose triviale, et plusieurs points de vue

doivent être pris en compte (voir [24] pour une étude exhaustive).

Le raisonnement par analogie est largement reconnu comme un élément clef de l'intelligence humaine [8, 11, 9, 16]. C'est un moyen puissant pour établir des parallèles entre des objets a priori non apparentés, et ainsi inférer des relations ou des propriétés sur la base des similarités/dissimilarités observées. Aussi, il est envisageable de concevoir un système de recommandation utilisant le raisonnement par analogie comme moyen d'inférence dans le but de proposer des recommandations pertinentes, et si possible surprenantes. Nous avons choisi d'explorer cette piste dans ce papier.

Les "proportions analogiques" permettent d'exprimer une relation analogique en faisant intervenir quatre éléments a, b, c, d , où l'on pose que " a est à b ce que c est à d ". Une telle proportion exprime le fait que les éléments a et b diffèrent de la même manière que les éléments c et d : cela permet une approche plus riche que de simplement considérer les plus proches voisins d'un élément, car on va chercher de l'information au-delà de leur voisinage. L'utilisation des proportions analogiques permet de concevoir des classificateurs très performants [2, 23, 5], mais l'analogie s'est aussi montrée efficace sur d'autres tâches d'intelligence artificielle [6, 21].

Les tâches de prédiction et de classification peuvent être considérées comme similaires, puisque les n notes possibles peuvent être vues comme autant de classes (à ceci près que dans le cas de la prédiction, ces "classes" sont ordonnées). En s'inspirant des travaux effectués en classification, mais aussi en essayant de s'adapter à ce qui semble être des spécificités propres aux SR, deux algorithmes basés sur l'analogie sont proposés dans ce papier, chacun reposant sur des définitions légèrement différentes du principe d'inférence analogique.

Notre papier est organisé comme suit. Dans la prochaine section, nous établissons une brève revue des techniques et outils pour la recommandation, ainsi que pour l'évaluation des SR. Dans la section 3, nous fournissons les prérequis nécessaires sur les proportions analogiques, et expliquons comment elles peuvent servir de principe d'inférence. Dans la section 4, nous nous intéressons à l'utilisation des proportions analogiques pour la conception d'un système de recommandation, en proposant deux implémentations. Dans la section 5, nous reportons les résultats obtenus sur une base de test et nous les comparons avec ceux de techniques existantes bien connues. Enfin, nous concluons et fournissons des pistes de recherche dans la section 6.

2 Prérequis sur les systèmes de recommandation

L'objectif d'un système de recommandation est de fournir à un utilisateur une liste personnalisée d'articles pertinents. Nous formalisons ici le problème.

2.1 Formalisation du problème de la recommandation

Soit U un ensemble d'utilisateurs et I un ensemble d'articles. Pour certaines paires $(u, i) \in U \times I$, on connaît la note r_{ui} , qui exprime l'intérêt que l'utilisateur u porte à l'article i .

Il est assez commun d'avoir $r_{ui} \in [1, 5]$, où 5 signifie un intérêt prononcé pour l'article i , 1 signifie un rejet et 3 une certaine indifférence, ou une note moyenne. Nous dénotons l'ensemble des notes connues du système par R . Bien sûr, dans les systèmes réels, le cardinal de R est très petit en comparaison du nombre total de notes possibles qui vaut $|U| \times |I|$, puisque de nombreuses notes sont inconnues. Nous notons U_i l'ensemble des utilisateurs qui ont noté l'article i , et symétriquement I_u représente l'ensemble des articles que u a notés.

Pour recommander des articles aux utilisateurs, un système de recommandation procède ainsi :

1. Avec un algorithme A , estimer les notes r_{ui} inconnues (c.à.d. $r_{ui} \notin R$). Cette estimation $A(u, i)$ est communément notée \hat{r}_{ui} .
2. Avec une stratégie de recommandation S et à la lumière des notes précédemment estimées, recommander des articles aux utilisateurs. Par exemple, une stratégie très basique mais assez courante est de suggérer à u les articles i pour lesquels \hat{r}_{ui} est le plus grand.

On distingue trois principales familles de systèmes de recommandation, chacune se distinguant par les techniques employées par l'algorithme de prédiction A : les systèmes basés sur le contenu (*content-based*), le filtrage collaboratif et les systèmes hybrides. Les méthodes hybrides sont en fait une combinaison des méthodes content-based et collaboratives. Nous nous intéresserons ici à ces deux dernières, sans trop nous attarder sur les méthodes content-based car elles se sont montrées moins performantes que les méthodes collaboratives.

2.2 Techniques content-based

Les algorithmes basés sur le contenu utilisent les métadonnées des utilisateurs et des articles pour estimer une note. Les métadonnées sont des informations externes qui peuvent être collectées, typiquement :

- pour les utilisateurs : le genre, l'âge, la profession, l'adresse, etc ;
- pour les articles : cela dépend bien sûr de la nature des articles, mais dans le cas de films on peut penser à leur genre, les acteurs principaux, le réalisateur, etc.

A partir de ces métadonnées, un système basé sur le contenu va essayer de trouver des articles similaires à ceux pour lequel l'utilisateur a déjà exprimé son intérêt (en donnant un score élevé par exemple). Cela suppose l'utilisation d'une mesure de similarité entre les articles. De nombreuses options sont envisageables pour une telle mesure. Elles ne seront pas discutées ici, car notre méthode est de nature collaborative.

Un inconvénient bien connu des techniques content-based est leur tendance à seulement recommander des articles que les utilisateurs connaissent déjà, et les recommandations sont ainsi peu surprenantes et manquent de diversité.

2.3 Le filtrage collaboratif

Les algorithmes de filtrage collaboratif reposent sur l'ensemble des notes connues R pour faire une prédiction : pour une prédiction r_{ui} ($r_{ui} \notin R$), A produira \hat{r}_{ui} en se basant sur un sous-ensemble de R soigneusement sélectionné. La différence principale entre les méthodes collaboratives et les méthodes content-based est que dans les premières les métadonnées ne sont pas utilisées, et dans les secondes les seules notes qui sont prises en compte sont celles de l'utilisateur cible. Deux approches collaboratives populaires sont l'approche basée sur le voisinage et l'approche par factorisation de matrice, que nous décrirons ici dans leur forme la plus simple.

L'approche par voisinage est une technique extrêmement basique, mais qui se montre néanmoins très efficace. Pour estimer la note d'un utilisateur u pour un article i , on sélectionne $N_i^k(u)$, l'ensemble des k utilisateurs les plus proches de u et qui ont noté i . Là aussi, on a besoin d'une mesure de similarité entre les utilisateurs. L'estimation de r_{ui} est calculée comme suit :

$$\hat{r}_{ui} = \operatorname{aggr}_{v \in N_i^k(u)} r_{vi},$$

où l'agrégat est habituellement une moyenne pondérée par la similarité entre u et v . Une prédiction un peu plus sophistiquée, popularisée par [3] est calculée ainsi :

$$\hat{r}_{ui} = b_{ui} + \operatorname{aggr}_{v \in N_i^k(u)} (r_{vi} - b_{vi}),$$

où b_{ui} est un biais lié à l'utilisateur u et à l'article i . Le biais caractérise la tendance de u à donner des notes au dessus ou en dessous de la moyenne générale μ et

la tendance des utilisateurs à noter i au dessus ou en dessous de μ . Comme elle utilise le voisinage des individus pour produire une prédiction, cette technique tend à modéliser les relations locales sous-jacentes aux données.

Notons qu'il est tout à fait possible d'adopter un point de vue centré sur les articles plutôt que sur les utilisateurs. En effet, plutôt que de chercher des utilisateurs similaires à u , on pourrait plutôt chercher des articles similaires à i , ce qui conduira à des formules duales de celles ci-dessus.

L'approche par factorisation de matrice est fortement inspirée par la décomposition en valeurs singulières : on suppose l'existence de f facteurs (ou critères) dont la nature n'est pas nécessairement connue et qui déterminent la valeur de chaque note r_{ui} . Un utilisateur u est modélisé par un vecteur $p_u \in \mathbb{R}^f$, où chaque composante de p_u caractérise l'importance du facteur correspondant pour u . D'une façon similaire, un article i est modélisé par un vecteur $q_i \in \mathbb{R}^f$, où chaque composante de q_i caractérise à quel point i satisfait le critère en question. A partir de là, une prédiction \hat{r}_{ui} est calculée comme le produit scalaire des deux vecteurs p_u et q_i :

$$\hat{r}_{ui} = p_u^t \cdot q_i.$$

Une fois le nombre de facteurs f fixé, le problème est ici d'estimer les vecteurs p_u et q_i pour tous les utilisateurs et tous les articles. Deux méthodes sont fréquemment utilisées pour cela : la régression des moindres carrés alternée et la descente de gradient stochastique, popularisée par [7].

Contrairement à la méthode de voisinage, la factorisation de matrice a tendance à modéliser et capturer les aspects globaux et généraux des données. Les deux méthodes ont été combinées avec succès dans les travaux de [13].

2.4 L'évaluation des systèmes de recommandation

Avoir une idée précise de la qualité globale d'un SR n'est pas une chose évidente, et plusieurs points de vue doivent être pris en compte (voir [24] pour une étude exhaustive). La précision des prédictions de l'algorithme est capitale, mais d'autres propriétés sont requises pour déployer un système efficace. Par exemple la *couverture*, qui mesure l'étendue des articles du système qui peuvent faire l'objet d'une recommandation, ou encore la *surprise* qui renseigne sur le caractère inattendu d'une recommandation.

Exactitude

La performance de l'algorithme de prédiction A est généralement évaluée du point de vue de l'exactitude

(*accuracy* en anglais), qui mesure à quel point les estimations de notes \hat{r}_{ui} sont proches de leurs vraies valeurs r_{ui} . Pour évaluer l'exactitude d'un algorithme de prédiction, on utilise généralement la technique de validation croisée bien connue en apprentissage artificiel : l'ensemble R des notes est divisé aléatoirement en deux ensembles disjoints R_{appr} et R_{test} , et l'algorithme A prédit les notes de R_{test} en se basant uniquement sur les notes de l'ensemble d'apprentissage R_{train} .

La racine de la moyenne du carré des erreurs (RMSE pour *Root Mean Squared Error*), est un indicateur très commun de l'exactitude d'un algorithme. On la calcule comme suit :

$$\text{RMSE}(A) = \sqrt{\frac{1}{|R_{test}|} \sum_{r_{ui} \in R_{test}} (\hat{r}_{ui} - r_{ui})^2}.$$

Un autre indicateur de l'exactitude est la moyenne des erreurs absolues (MAE pour *Mean Absolute Error*), où les grandes erreurs ne sont pas plus pénalisantes que les petites :

$$\text{MAE}(A) = \frac{1}{|R_{test}|} \sum_{r_{ui} \in R_{test}} |\hat{r}_{ui} - r_{ui}|.$$

Précision et rappel

La précision et le rappel permettent de mesurer la capacité d'un système à fournir des recommandations pertinentes. Dans ce qui suit, on notera I_S l'ensemble des articles que la stratégie S suggèrera à n'importe quel utilisateur, à partir des prédictions venant de l'algorithme A . Pour des notes appartenant à l'intervalle $[1, 5]$, une stratégie simple serait par exemple de recommander l'article i à l'utilisateur u si l'estimation \hat{r}_{ui} est supérieure à 4.

$$I_S = \{i \in I \mid \exists u \in U, \hat{r}_{ui} \geq 4\}.$$

Soit I_{pert} l'ensemble des articles qui sont *effectivement* pertinents pour les utilisateurs (c.à.d. l'ensemble des articles qui auraient été recommandés si toutes les prédictions de A étaient exactes) :

$$I_{pert} = \{i \in I \mid \exists u \in U, r_{ui} \geq 4\}.$$

La précision du système est définie comme la fraction d'articles recommandés qui sont pertinents pour les utilisateurs :

$$\text{Précision} = \frac{|I_S \cap I_{pert}|}{|I_S|},$$

et le rappel est défini comme la fraction d'articles recommandés qui sont pertinents, parmi tous les articles pertinents possibles :

$$\text{Rappel} = \frac{|I_S \cap I_{relev}|}{|I_{relev}|}.$$

Si des prédictions précises sont cruciales, il est largement admis que cela reste insuffisant pour déployer un système de recommandation efficace. D'autres mesures sont nécessaires [15, 10, 12].

Par exemple, on peut naturellement attendre d'un système de recommandation qu'il soit non seulement précis, mais aussi surprenant, et être capable de recommander le plus d'articles possible.

Couverture

La couverture, dans sa forme la plus simple, est utilisée pour mesurer la capacité d'un système à recommander une grande quantité d'articles. Il est assez commun en effet de tomber dans l'écueil classique qui consiste à ne recommander que des articles très populaires. Un tel système n'apporterait aucune information intéressante à ses utilisateurs.

La couverture peut être définie comme la proportion d'articles recommandés par rapport à tous les articles du système :

$$\text{Coverage} = \frac{|I_S|}{|I|}.$$

Surprise

Les utilisateurs attendent d'un système de recommandation qu'il soit surprenant, c'est à dire qu'il suggère des articles qu'ils n'auraient pas pu découvrir sans son aide. Selon les travaux de [12], la surprise d'un système de recommandation peut être évaluée avec l'aide de l'information mutuelle ponctuelle (PMI pour *Pointwise Mutual Information*). La PMI entre deux articles i et j est définie ainsi :

$$\text{PMI}(i, j) = -\log_2 \frac{p(i, j)}{p(i)p(j)} / \log_2 p(i, j),$$

où $p(i)$ et $p(j)$ représentent les probabilités pour les articles i et j d'être notés par n'importe quel utilisateur, et $p(i, j)$ est la probabilité que i et j soient notés ensemble : $p(i) = \frac{|U_i|}{|U|}$ et $p(i, j) = \frac{|U_i \cap U_j|}{|U|}$ (on rappelle que U_i est l'ensemble des utilisateurs qui ont noté l'article i). Pour estimer la surprise apportée par la recommandation d'un article i à un utilisateur u , nous avons deux choix :

- soit prendre le maximum des valeurs de PMI pour i et tous les autres articles notés par u :

$$\text{Surp}^{max}(u, i) = \max_{j \in I_u} \text{PMI}(i, j)$$

- soit prendre la moyenne de ces valeurs de PMI :

$$\text{Surp}^{avg}(u, i) = \frac{\sum_{j \in I_u} \text{PMI}(i, j)}{|I_u|}$$

Alors, la capacité globale du système à surprendre ses utilisateurs est la moyenne de toutes les valeurs de surprise sur l'ensemble des prédictions.

3 Prérequis sur les proportions analogiques

Une proportion analogique “ a est à b ce que c est à d ” exprime une relation analogique entre les paires (a, b) et (c, d) , ainsi qu’entre les paires (a, c) et (b, d) . On peut trouver de nombreux exemples naturels de telles proportions analogiques, comme par exemple “le veau est à la vache ce que le poulain est à la jument”, ou encore “le pinceau est au peintre ce que la craie est au professeur”. Cependant, ce n’est que récemment que des définitions formelles ont été proposées pour les proportions analogiques, dans des cadres différents [26, 14, 18]. Dans cette section, nous donnons un bref résumé de la formalisation des proportions analogiques qui nous aideront à décrire nos algorithmes par la suite. Pour une introduction générale, nous invitons le lecteur à se référer à [20, 19, 22].

3.1 Cadre formel

Il est entendu depuis l’époque d’Aristote qu’une proportion analogique T est une relation quaternaire qui satisfait les trois axiomes suivants :

1. $T(a, b, a, b)$ (reflexivité)
2. $T(a, b, c, d) \implies T(c, d, a, b)$ (symétrie)
3. $T(a, b, c, d) \implies T(a, c, b, d)$ (permutation centrale)

Il existe plusieurs modèles de proportions analogiques, dépendant du domaine de définition. Lorsque le domaine est défini, $T(a, b, c, d)$ s’écrit plus simplement $a : b :: c : d$. Dans ce papier, on s’intéresse à évaluer des proportions analogiques sur des notes, qui peuvent être soit booléennes (*aime/n’aime pas*), soit dans notre cas d’étude des valeurs entières (dans l’intervalle $[1, 5]$). L’idée de proportion analogique est similaire à celle de proportion numérique pour les nombres :

- Domaine \mathbb{R} :
 $a : b :: c : d \iff \frac{a}{b} = \frac{c}{d} \iff ad = bc$
(proportion géométrique).
- Domaine \mathbb{R} :
 $a : b :: c : d \iff a - b = c - d \iff a + d = b + c$
(proportion arithmétique).
- Domaine \mathbb{R}^n :
 $\vec{a} : \vec{b} :: \vec{c} : \vec{d} \iff \vec{a} - \vec{b} = \vec{c} - \vec{d}$. C’est simplement une extension de la proportion arithmétique aux vecteurs réels. Dans ce cas, les quatre vecteurs $\vec{a}, \vec{b}, \vec{c}, \vec{d}$ forment un parallélogramme.
- Domaine $\mathbb{B} = \{0, 1\}$:
 $a : b :: c : d \iff (a \wedge d \equiv b \wedge c) \wedge (a \vee d \equiv b \vee c)$.
- Domaine \mathbb{B}^n :
 $\vec{a} : \vec{b} :: \vec{c} : \vec{d} \iff \forall i \in [1, n], a_i : b_i :: c_i : d_i$.
C’est une extension directe du cas booléen.

D’autres définitions existent pour les matrices, les concepts formels, les treillis, etc. (voir [26, 14, 17]).

Résolution d’équation

A partir de trois éléments a, b, c , le problème de la résolution d’équation consiste à trouver un quatrième élément x de telle sorte que la proportion $a : b :: c : x$ soit valide.

Dans le cas des proportions booléennes, le problème n’est pas toujours soluble : bien qu’il soit facile de compléter $1 : 0 :: 1 : x$ pour construire une proportion valide, il n’existe pas de solution à l’équation $1 : 0 :: 0 : x^1$.

Inférence analogique

Le raisonnement analogique peut être vu comme un moyen d’inférer de nouvelles connaissances à partir de proportions analogiques observées. Nous formalisons ici ce raisonnement dans le cas où les éléments $\vec{a}, \vec{b}, \vec{c}, \vec{d}$ sont des vecteurs de \mathbb{R}^n ou de \mathbb{B}^n . Le *saut analogique* est un principe d’inférence non sûr qui postule que pour quatre vecteurs $\vec{a}, \vec{b}, \vec{c}, \vec{d}$ tels que la proportion est vraie sur certaines composantes, alors elle reste vraie aussi sur les composantes restantes. On peut formuler cela de la manière suivante (où $\vec{a} = (a_1, a_2, \dots, a_n)$ et $J \subset [1, n]$) :

$$\frac{\forall j \in J, a_j : b_j :: c_j : d_j}{\forall i \in [1, n] \setminus J, a_i : b_i :: c_i : d_i} \quad (\text{inférence analogique})$$

Ce principe amène à une règle de prédiction dans le contexte suivant :

- On a 4 vecteurs $\vec{a}, \vec{b}, \vec{c}, \vec{d}$ où \vec{d} n’est que partiellement connu : seules les composantes de \vec{d} dont les indices sont dans J sont connues.
- En utilisant l’inférence analogique, on peut prédire les composantes manquantes de \vec{d} en résolvant l’ensemble des équations (dans le cas où elles sont solubles)

$$\forall i \in [1, n] \setminus J, a_i : b_i :: c_i : d_i = 1.$$

Dans le cas où les éléments sont tels que leur dernière composante est une classe (ou label), appliquer ce raisonnement sur \vec{d} revient à prédire une classe candidate pour \vec{d} . Lorsque l’on peut trouver plusieurs triplets $\vec{a}, \vec{b}, \vec{c}$ qui satisfont la proportion analogique sur les indices dans J , il est possible d’obtenir plusieurs classes candidates et il est alors nécessaire de faire un choix (par une procédure de vote ou n’importe quelle technique d’agrégation), afin d’obtenir une seule et unique classe. Comme évoqué dans l’introduction, ce principe d’inférence a été appliqué avec succès à des problèmes de classification.

1. Dans le cadre booléen, la proportion $a : b :: c : d$ est vraie pour seulement six quadruplets (contre 2^4 possibles).

4 Recommandation analogique

Nous avons développé deux algorithmes pour la recommandation, chacun étant inspiré par deux vues différentes du principe d'inférence analogique. Dans les deux cas, l'idée principale est que si une proportion analogique est satisfaite entre trois utilisateurs a, b, c, d (c.à.d. que pour chaque article j qu'ils ont noté en commun la proportion analogique $r_{aj} : r_{bj} :: r_{cj} : r_{dj}$ est vraie), alors la proportion doit aussi être vraie pour un film i que a, b, c ont noté mais que d n'a pas noté : la valeur r_{di} est la composante manquante. Cela nous conduit à estimer r_{di} comme la solution \hat{r}_{di} de l'équation analogique suivante :

$$r_{ai} : r_{bi} :: r_{ci} : x$$

Etant donnée une paire (u, i) , telle que $r_{ui} \notin R$ (on ne connaît pas la note de u pour i) la procédure principale est la suivante :

1. Trouver l'ensemble des triplets d'utilisateurs a, b, c tels qu'une proportion analogique est vraie entre a, b, c , et u et tels que l'équation

$$r_{ai} : r_{bi} :: r_{ci} : x$$

est soluble.

2. Résoudre l'équation $r_{ai} : r_{bi} :: r_{ci} : x$, et considérer que la solution r est une note candidate pour r_{ui} .
3. Définir \hat{r}_{ui} comme un agrégat de toutes les notes candidates.

La première étape stipule simplement que dans l'espace vectoriel de dimension $|I_{abcu}|$, les utilisateurs a, b, c et u , considérés comme des vecteurs de notes sur I_{abcu} , forment un parallélogramme. Evidemment, cette condition est relativement stricte et il peut être nécessaire de la relaxer, en permettant au parallélogramme de se déformer. On peut donc remplacer la condition 3 par $\|(a-b) - (c-d)\| \leq \lambda$, où λ est un seuil adéquat et où $\|\cdot\|$ correspond à la norme euclidienne.

Une autre modification de l'algorithme serait de chercher les utilisateurs a, b et c uniquement dans un sous-ensemble de U . On peut penser à l'ensemble des k plus proches voisins de d , en utilisant l'hypothèse que ce sont les voisins de d qui sont les plus à même d'estimer correctement sa note pour i . Cela permet aussi de réduire la taille de l'espace de recherche des triplets et ainsi le temps d'exécution.

Evidemment, l'analogie peut être appliquée avec une approche basée sur les articles plutôt que sur les utilisateurs, comme c'est le cas pour les techniques classiques. Les deux approches seront étudiées dans l'expérimentation.

4.1 Algorithme utilisant les proportions analogiques

Ici, on utilise la définition de la proportion analogique basée sur la proportion arithmétique :

$$a : b :: c : d \iff a - b = c - d.$$

Etendue aux vecteurs, cette définition indique que $(\vec{a}, \vec{b}, \vec{c}, \vec{d})$ sont les sommets d'un parallélogramme. Une implémentation directe de la procédure ci-dessus mène à l'algorithme 1, où I_{abcu} désigne l'ensemble des articles j que les utilisateurs a, b, c et u ont conjointement notés ($r_{aj} \in R, r_{bj} \in R, r_{cj} \in R$ et $r_{uj} \in R$).

Algorithm 1

Entrée : Un ensemble de notes connues R , un utilisateur u et un article i tels que $r_{ui} \notin R$

Sortie : \hat{r}_{ui} : une estimation de r_{ui}

Initialisation :

$C = \emptyset$ // liste de solutions candidates

for all utilisateurs $\vec{a}, \vec{b}, \vec{c}$ voisins de u tels que

1. $r_{ai} \in R, r_{bi} \in R, r_{ci} \in R$

2. $r_{ai} - r_{bi} = r_{ci} - x$ est soluble

3. $\|(\vec{a}-\vec{b}) - (\vec{c}-\vec{d})\| \leq \lambda$ // l'analogie est presque vérifiée entre a, b, c et u

do

$x \leftarrow r_{ci} - r_{ai} + r_{bi}$

$C \leftarrow C \cup \{x\}$ // ajouter x comme une solution candidate

end for

$\hat{r}_{ui} = \text{aggr } x$
 $x \in C$

Nous avons considéré une condition stricte de résolubilité pour l'équation $r_{ai} - r_{bi} = r_{ci} - x$: comme le résultat exact $x = r_{ci} + r_{bi} - r_{ai}$ n'appartient pas nécessairement à l'intervalle $[1, 5]$, nous considérons l'équation soluble uniquement lorsque $r_{ai} = r_{bi}$ ou $r_{ai} = r_{ci}$. Cela assure que la solution sera bien dans $[1, 5]$.

Chaque utilisateur d'un triplet (a, b, c) est représenté comme un vecteur de notes de dimension m . Il est clair que cette dimension m varie en fonction du triplet considéré, puisqu'il est très probable que les articles conjointement notés par les utilisateurs de deux triplets distincts soient eux aussi différents. Aussi, il est nécessaire que le seuil λ soit une fonction de m , car l'étendue des valeurs que $\|\cdot\|$ peut prendre dépend de m . Nous avons observé expérimentalement que $\lambda = \frac{3}{2} \cdot \sqrt{m}$ permettait d'obtenir des résultats optimaux.

4.2 Algorithme avec principe d'inférence reposant sur des motifs

Une idée intuitive sous-jacente à la recommandation par filtrage collaboratif est que des utilisateurs ayant exprimé des goûts similaires sur plusieurs articles devraient, de même, avoir des goûts similaires sur de nouveaux articles. Cependant, on peut observer que dans un groupe d'utilisateurs ayant des goûts similaires, les avis peuvent diverger sur certains articles donnés.

Avec l'algorithme précédent, il peut se produire la situation (non désirée) suivante :

- i quatre utilisateurs a, b, c, u ont des goûts assez similaires sur tous les articles qu'ils ont conjointement notés (avec peut-être quelques divergences négligeables et permises par la déformation du parallélogramme), alors que
- ii l'équation à résoudre $r_{ai}, r_{bi}, r_{ci}, x$ où i est l'article que u n'a pas noté est telle que r_{ai} est à peu près égal à r_{ci} mais diffère fortement de r_{bi} (par exemple $1 : 4 :: 1 : x$, ou encore $5 : 2 :: 5 : x$).

Dans ce cas précis, le motif de l'équation à résoudre ne se retrouve pas dans les analogies construites par les articles conjointement notés, qui sont du type $1 : 1 :: 1 : 1$, $4 : 4 :: 4 : 5$, etc.

Or, il semble naturel de souhaiter éviter le genre de situation où le motif de l'équation à résoudre ne se retrouve pas (ou peu) parmi les composantes. En effet, dans ce cas, on peut considérer que l'on ne dispose pas suffisamment d'information pertinente pour appliquer l'inférence analogique puisque l'on n'a pas observé sur les composantes un motif identique à celui de l'équation à résoudre.

Cette observation amène à la conception d'un second algorithme où le principe d'inférence est légèrement modifié, comme expliqué ci-dessous.

Tout comme pour l'algorithme précédent, on cherche des triplets d'utilisateurs a, b, c tels que l'équation analogique $r_{ai} : r_{bi} :: r_{ci} : x$ soit soluble.

Pour chaque article $j \in I_{abcu}$, on assigne le quadruplet $(r_{aj}, r_{bj}, r_{cj}, r_{uj})$ à trois catégories :

1. *accord* : $(r_{aj}, r_{bj}, r_{cj}, r_{uj}) \in \text{Cat 1}$ ssi trois d'entre eux sont égaux et le quatrième diffère avec au plus un écart de 1.
Ex : $(4, 4, 4, 4) \in \text{Cat 1}$, $(4, 3, 4, 4) \in \text{Cat 1}$, mais $(4, 3, 4, 3) \notin \text{Cat 1}$
2. *Changement dans la même direction* :
 $(r_{aj}, r_{bj}, r_{cj}, r_{uj}) \in \text{Cat 2}$ ssi
 - (a) r_{aj} diffère de r_{bj} avec une différence d'au moins 2, et de même pour r_{cj} et r_{uj} ;
 - (b) le changement entre r_{aj} et r_{bj} est fait dans la même direction que celui entre r_{cj} et r_{uj} .

3. *autres* (Cat 3) : tous les quadruplets qui n'appartiennent pas aux deux premières catégories appartiennent à la troisième.

Il est clair que lorsque $|\text{Cat 3}|$ est négligeable par rapport à $|\text{Cat 1}| + |\text{Cat 2}|$, la proportion analogique est vraie (au moins approximativement) entre les quatre ensembles de notes.

Résolution d'équation et filtrage des solutions candidates

On observe trois cas différents, selon le motif de l'équation analogique $r_{ai} : r_{bi} :: r_{ci} : x$:

1. $a : a :: a : x \implies x = a$.
 x est candidat ssi $|\text{Cat 1}| \geq |\text{Cat 2}| + |\text{Cat 3}|$.
2. $a : b :: a : x, |a - b| \geq 2 \implies x = b$.
 x est candidat ssi $|\text{Cat 2}| > |\text{Cat 3}|$.
3. $a : b :: a : x, |a - b| = 1 \implies x = b$.
 x est candidat ssi $|\text{Cat 1}| \geq |\text{Cat 2}| + |\text{Cat 3}|$ ou $|\text{Cat 2}| > |\text{Cat 3}|$.

Là aussi, toutes les solutions candidates sont agrégées pour obtenir une seule et unique estimation \hat{r}_{ui} .

5 Expérimentations et résultats

Nous avons implémentés en Python et testés nos deux algorithmes, auxquels nous nous référons par *Paral* et par *Pattern*. Pour ces deux algorithmes, nous cherchons les utilisateurs a, b, c dans l'ensemble des k plus proches voisins de u avec $k = 20$. Ils sont comparés avec les algorithmes de voisinage décrits dans la section 2.3, auxquels nous nous référons par *k-NN* pour le modèle de base et *k-NN Biases* pour le modèle sophistiqué utilisant un biais. Pour ces deux algorithmes, nous avons choisi d'utiliser $k = 40$.

Pour chaque algorithme, nous avons calculé les mesures suivantes (décrites dans la section 2.4) :

- L'exactitude mesurée par la RMSE : une bonne exactitude correspond à une valeur de RMSE proche de 0.
- Précision et rappel : de bonnes valeurs correspondent à un pourcentage élevé.
- Couverture : idem.
- Surprise : ce sont en fait des mesures de non-surprise. Aussi, plus le nombre est faible, plus les recommandations sont surprenantes. Ces mesures sont notées S_{max} et S_{avg} .
- Le temps utilisateur nécessaire pour effectuer les recommandations (on ne prend pas en compte le temps d'apprentissage qui est le même pour les différents algorithmes).
- Pourcentage de prédictions impossibles.

Pour chaque algorithme, la stratégie de recommandation S est de recommander l'article i à l'utilisateur u si $\hat{r}_{ui} \geq 4$.

5.1 Jeu de données et protocole d'évaluation

Jeu de données

Nous avons comparé les algorithmes sur la base de données MovieLens-100K², qui contient 100.000 notes venant de 1000 utilisateurs pour 1700 films. Chaque note appartient à l'intervalle $[1, 5]$. La base de données propose des métadonnées sur les films et les utilisateurs, mais nous ne les avons pas utilisées.

Protocole d'évaluation

Afin d'obtenir des mesures légitimes, nous avons procédé à une campagne de validation croisée à cinq plis : pour chaque pli, l'ensemble des notes R est scindé aléatoirement en deux sous-ensembles disjoints R_{appr} et R_{test} , où R_{appr} contient cinq fois plus de notes que R_{test} . Les mesures reportées sont des moyennes sur les cinq plis.

5.2 Résultats et analyse

La table 5.2 illustre les performances des algorithmes. Des expériences similaires ont été menées en raisonnant sur des articles plutôt que sur des utilisateurs, montrant des résultats très semblables mais légèrement moins bons en RMSE (environ 5% plus grand).

	RMSE	Prec	Rap	Couv	S_{max}	S_{avg}	Time	Imp
Paral	.898	89.1	43.3	31.2	0.433	0.199	2h	1.1
Pattern	.927	84.9	50.0	47.7	0.433	0.198	5h	1.2
k -NN	.894	89.1	44.1	27.8	0.432	0.198	10s	0.1
k -NN <i>Biais</i>	.865	88.4	44.0	44.7	0.431	0.199	10s	0.1

TABLE 1 – Performance des algorithmes

Comme attendu, l'algorithme k -NN*Biais* est plus performant que l'algorithme de filtrage collaboratif basique k -NN. Cependant, ces deux algorithmes classiques sont supérieurs aux algorithmes analogiques proposés en ce qui concerne la RMSE. Il semble néanmoins qu'une certaine marge de progression soit envisageable pour ces deux nouveaux algorithmes, à la condition d'une analyse minutieuse de leur comportement. Pour les mesures autres que la RMSE, les résultats obtenus sont assez proches pour les quatre algorithmes. En ce qui concerne la surprise, qui est une notion délicate à quantifier, on peut se demander si la mesure utilisée ici est vraiment appropriée.

Des deux algorithmes analogiques, c'est étonnamment le premier qui donne les meilleurs résultats. Ceci

peut être dû au fait que le second est trop permissif dans la manière où les proportions analogiques sont satisfaites. De plus, le nombre de prédictions impossibles est supérieur, puisque les conditions de résolubilité des équations analogiques sont plus strictes. Une option envisageable serait d'hybrider les deux procédures.

Les deux algorithmes analogiques souffrent de leur complexité cubique, caractéristique de ce type d'approche. Dans le cas des systèmes de recommandation où des millions d'utilisateurs et d'articles entrent en jeu, cela devient un inconvénient non négligeable qui devra être contourné d'une manière ou d'une autre.

6 Conclusion et futures pistes

Nous en sommes clairement à un stade d'études préliminaires sur l'approche analogique pour la prédiction dans les systèmes de recommandation, qui est un sujet qui n'a jamais été exploré auparavant. Nos premiers résultats ne sont pas meilleurs que ceux obtenus avec les approches standards. Même s'ils ne semblent pas en être trop éloignés, les différences restent suffisamment grandes pour avoir un impact sur l'expérience utilisateur du système. Cependant, il est intéressant d'observer que des approches basées sur des idées assez différentes mènent à des résultats comparables.

De plus, il doit être noté que le problème de la prédiction pour la recommandation, bien que similaire à un problème de classification (pour lequel les classificateurs analogiques sont performants), présente des différences significatives, puisque les notes des articles qui jouent ici le rôle de composantes descriptives peuvent être assez redondantes et en quelque sorte incomplètes pour fournir un profil cohérent d'un utilisateur. Ceci pourrait expliquer le fait que l'application d'une approche analogique est moins performante pour la recommandation que pour la classification.

Des méthodes basées sur la proportion analogique ont aussi été utilisées récemment pour prédire des valeurs booléennes manquantes dans des bases de données [4]. Le problème de la recommandation peut aussi être vu comme un problème de valeurs manquantes, mais ici la proportion de données inconnues est considérable, et les données ne sont pas nécessairement booléennes. Cela aussi suggère que la tâche de recommandation est plus difficile.

Il reste cependant un certain nombre de pistes à explorer, tel que comprendre dans quelles situations une approche analogique serait plus performante que les approches classiques, et dans quelles autres situations une autre approche est préférable.

Enfin, l'exploitation du potentiel créatif de l'analogie pour proposer des articles jamais considérés par un utilisateur mais possédant des caractéristiques com-

2. <http://grouplens.org/datasets/movielens/>

munes avec des articles qu'il apprécie [25] ainsi que l'exploitation de son pouvoir explicatif pour suggérer pourquoi un article peut être intéressant, restent entièrement à explorer.

Références

- [1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems : a survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6) :734–749, June 2005.
- [2] S. Bayouhd, L. Miclet, and A. Delhay. Learning by analogy : A classification rule for binary and nominal data. *Proc. Inter. Joint Conf. on Artificial Intelligence IJCAI07*, pages 678–683, 2007.
- [3] R. M. Bell and Y. Koren. Lessons from the netflix prize challenge. *SIGKDD Explor. Newsl.*, 9(2) :75–79, December 2007.
- [4] W. Correa Beltran, H. Jaudoin, and O. Pivert. Estimating null values in relational databases using analogical proportions. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems - 15th International Conference, IPMU 2014, Montpellier, France, July 15-19, 2014, Proceedings, Part III*, pages 110–119, 2014.
- [5] Myriam Bounhas, Henri Prade, and Gilles Richard. Analogical classification : A new way to deal with examples. In *ECAI 2014 - 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic*, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pages 135–140. IOS Press, 2014.
- [6] W. Correa, H. Prade, and G. Richard. When intelligence is just a matter of copying. In *Proc. 20th Eur. Conf. on Artificial Intelligence, Montpellier, Aug. 27-31*, pages 276–281. IOS Press, 2012.
- [7] S. Funk. Netflix update : Try this at home. <http://sifter.org/~simon/journal/20061211.html>, 2006.
- [8] D. Gentner. Structure-mapping : A theoretical framework for analogy. *Cognitive Science*, 7(2) :155–170, 1983.
- [9] D. Gentner, K. J. Holyoak, and B. N. Kokinov. *The Analogical Mind : Perspectives from Cognitive Science*. Cognitive Science, and Philosophy. MIT Press, Cambridge, MA, 2001.
- [10] J. L. Herlocker, J. A. Konstan, Loren G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1) :5–53, 2004.
- [11] D. Hofstadter and M. Mitchell. The Copycat project : A model of mental fluidity and analogy-making. In D. Hofstadter and The Fluid Analogies Research Group, editors, *Fluid Concepts and Creative Analogies : Computer Models of the Fundamental Mechanisms of Thought*, pages 205–267, New York, NY, 1995. Basic Books, Inc.
- [12] M. Kaminskas and D. Bridge. Measuring surprise in recommender systems. In P. Adamopoulos et al., editor, *Procs. of the Workshop on Recommender Systems Evaluation : Dimensions and Design (Workshop Programme of the 8th ACM Conference on Recommender Systems)*, 2014.
- [13] Yehuda Koren. Factor in the neighbors : Scalable and accurate collaborative filtering. *ACM Trans. Knowl. Discov. Data*, 4(1) :1 :1–1 :24, January 2010.
- [14] Y. Lepage. De l'analogie rendant compte de la commutation en linguistique. *Habilit. à Diriger des Recher., Univ. J. Fourier, Grenoble*, 2003.
- [15] S. M. McNee, J. Riedl, and J. A. Konstan. Being accurate is not enough : how accuracy metrics have hurt recommender systems. In G. M. Olson and R. Jeffries, editors, *Extended Abstracts Proceedings of the 2006 Conference on Human Factors in Computing Systems, CHI 2006, Montréal, Québec, Canada, April 22-27, 2006*, pages 1097–1101, 2006.
- [16] E. Melis and M. Veloso. Analogy in problem solving. In *Handbook of Practical Reasoning : Computational and Theoretical Aspects*. Oxford Univ. Press, 1998.
- [17] L. Miclet, N. Barbot, and B. Jeudy. Analogical proportions in a lattice of sets of alignments built on the common subwords in a finite language. In H. Prade and G. Richard, editors, *Computational Approaches to Analogical Reasoning - Current Trends*. Springer, 2013.
- [18] L. Miclet and H. Prade. Handling analogical proportions in classical logic and fuzzy logics settings. In *Proc. 10th Eur. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'09), Verona*, pages 638–650. Springer, LNCS 5590, 2009.
- [19] H. Prade and G. Richard. Analogical proportions and multiple-valued logics. In L. C. van der Gaag, editor, *Proc. 12th Eur. Conf. on Symb. and Quantit. Appr. to Reasoning with Uncertainty (ECSQARU'13), Utrecht, Jul. 7-10*, LNAI 7958, pages 497–509. Springer, 2012.

- [20] H. Prade and G. Richard. From analogical proportion to logical proportions. *Logica Universalis*, 7(4) :441–505, 2013.
- [21] H. Prade and G. Richard. Picking the one that does not fit - a matter of logical proportions. In *Proc. 8th Conf. Europ. Soc. for Fuzzy Logic and Technology (EUSFLAT), Milano, Sept. 11-13*, 2013.
- [22] H. Prade and G. Richard. Homogenous and heterogeneous logical proportions. *IfCoLog J. of Logics and their Applications*, 1(1) :1–51, 2014.
- [23] H. Prade, G. Richard, and B. Yao. Classification by means of fuzzy analogy-related proportions. A preliminary report. In *Proc. 2d IEEE Int. Conf. Soft Comp. & Pattern Recog. (SocPar'10), Evry*. 2010, 297-302.
- [24] F. Ricci, L. Rokach, B. Shapira, and P.B. Kantor, editors. *Recommender Systems Handbook*. Springer, 2011.
- [25] T. Sakaguchi, Y. Akaho, K. Okada, T. Date, T. Takagi, N. Kamimaeda, M. Miyahara, and T. Tsunoda. Recommendation system with multi-dimensional and parallel-case four-term analogy. *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3137 – 3143, 2011.
- [26] F. Yvon and N. Stroppa. Formal models of analogical proportions. Technical report, Ecole Nationale Supérieure des Télécommunications, no D008, 2006.

Transformations probabilistes-possibilistes: conditionnement, inférence et modèles graphiques

Salem Benferhat Amélie Levray Karim Tabia

Univ Lille Nord de France, F-59000 Lille, France
UArtois, CRIL UMR CNRS 8188, F-62300 Lens, France
{benferhat,levray,tabia}@cril.univ-artois.fr

Résumé

Le problème de la représentation et du raisonnement dans l'incertain est un sujet important en Intelligence artificielle. Dans ce papier, nous étudions les transformations probabilistes-possibilistes (dans le sens des probabilités vers les possibilités) à travers plusieurs questions telles que l'inférence et les modèles graphiques. Les travaux qui traitent des transformations sont principalement centrés sur les propriétés intéressantes à satisfaire et proposent des transformations les satisfaisant. Peu de travaux ont été développés en ce qui concerne l'analyse du comportement de ces transformations face à des opérations comme le conditionnement et la marginalisation. Ce papier s'intéresse à la commutativité des transformations lors de tâches de raisonnement telles que marginaliser et conditionner. Un autre problème abordé dans ce papier est celui des transformations dans les modèles graphiques, plus précisément l'indépendance des événements et des variables.

Abstract

Representing and reasoning with uncertain information is a common topic in Artificial Intelligence. In this paper, we focus on probability-possibility transformations (from probability to possibility) in the context of changing operations and graphical models. Existing works mainly propose probability-possibility transformations satisfying some desirable properties. Regarding the analysis of the behavior of these transformations with respect to changing operations (such as conditioning and marginalization), only few works addressed such issues. This paper concerns the commutativity of transformations with respect to some reasoning tasks such as marginalization and conditioning. Another crucial issue addressed in this paper is the one of probability-possibility transformations in the context of graphical models, especially the independence of events and variables.

1 Introduction

De nombreux cadres existent pour représenter l'information en présence d'imprécision, d'incertitude et de contradictions, tels que la théorie des probabilités, la théorie des possibilités, les probabilités imprécises ou encore la théorie de Dempster-Shafer. La théorie des probabilités est la plus connue et par conséquent la plus développée mais lorsqu'on fait face à des cas d'ignorance totale ou partielle, ou à des informations incomplètes en général, cette théorie n'est peut-être pas la plus adaptée. C'est pour cela que des cadres de représentation alternatifs tels que la théorie des possibilités ont été proposés.

Les premiers travaux mêlant la théorie des probabilités et la théorie des possibilités ont essayé de trouver les connections entre ces théories en définissant des transformations pour passer d'un cadre à un autre. On peut voir plusieurs raisons pour lesquelles les transformations seraient utiles. Par exemple, pour fusionner des informations multi-sources, plusieurs agents peuvent fournir des informations mais dans différents formalismes. La question est de savoir comment fusionner ces informations. Une solution serait de transformer toutes les informations dans un seul cadre. Un autre exemple serait celui où on est en présence d'informations statistiquement pauvres. En effet, les probabilités sont appropriées dans un cadre empirique, mais cela requiert une quantité d'informations importante. Et quand ces informations ne sont pas disponibles en quantité suffisante, les possibilités peuvent combler ce manque comme dans [13]. On trouve dans [1] un exemple de propagation probabiliste (stochastique) et d'information possibiliste en analyse de risques. Une motivation supplémentaire aux transformations est de pouvoir utiliser des outils développés dans un cadre

(comme les algorithmes d'inférence, les logiciels) sans avoir à tout réimplémenter ou reprogrammer pour le cadre que l'on utilise. Par exemple, de nombreux outils existent pour faire de l'inférence dans les réseaux Bayésiens (comme Javabayes). Dans ce cas, il serait utile de pouvoir passer des possibilités aux probabilités, avec un minimum de perte d'information, pour utiliser ces outils.

Les transformations probabilistes-possibilistes ont fait l'objet de plusieurs travaux [20, 12, 15], mais ces derniers se sont plus concentrés à trouver de bonnes propriétés à satisfaire et les transformations les satisfaisant. Un exemple de telles propriétés est le principe de consistance dont le but est de préserver autant d'information que possible. Dans [20, 12, 15], plusieurs principes de consistance ont été définis. Parmi les travaux sur les transformations, l'auteur du papier [18] s'est intéressé, précisément, à la commutativité des tâches de raisonnement. Un autre papier [16] a étudié les transformations dans le cadre des modèles graphiques en transformant un réseau Bayésien en réseau possibiliste. Dans [5], les auteurs proposent des transformations des probabilités imprécises vers d'autres formalismes incertains. Un travail plus récent [19] utilise les transformations probabilistes-possibilistes pour passer d'un réseau Bayésien à une carte cognitive floue avec une application en analyse causale.

Dans le présent travail, nous étudions les transformations probabilistes-possibilistes et leur comportement lors de tâches de raisonnement et dans les modèles graphiques. Il est à noter que nous nous sommes intéressés uniquement aux transformations du cadre probabiliste vers le cadre possibiliste. Étant donnée une distribution encodant des informations incertaines, qu'elles soient probabilistes ou possibilistes, nous sommes supposés pouvoir raisonner à propos d'événements d'intérêt. Ce travail s'intéresse à des problématiques complémentaires telles que préserver la marginalisation, le conditionnement et les relations d'indépendance. Nous étudions ces questions lorsque les informations sont représentées à l'aide de distributions ou de modèles graphiques. Nous montrons qu'il n'existe pas de transformation, du cadre probabiliste vers le cadre possibiliste, qui préserve la plupart des tâches de raisonnement présentées dans ce travail. Par exemple, pour la préservation de la marginalisation, nous montrons qu'aucune transformation ne peut préserver l'ordre relatif des événements même si l'ordre relatif des interprétations est préservé. De même, l'ordre des interprétations ne peut être préservé lorsqu'on transforme les modèles graphiques probabilistes en modèles possibilistes. Rappelons, avant de présenter nos résultats, les principaux concepts des transformations.

2 Théorie des probabilités, théorie des possibilités et modèles graphiques

Les probabilités sont le cadre le plus connu et le plus ancien pour représenter et raisonner avec des informations incertaines. Le principal concept des probabilités est la notion de distribution de probabilités p qui associe à chaque état élémentaire (ou interprétation) ω_i du monde Ω , un degré de probabilité. La théorie des probabilités est régie par les axiomes de Kolmogorov (non-négativité, normalisation et additivité) et peut s'interpréter de deux manières (de manière fréquentiste ou de manière subjective).

Parmi les alternatives à la théorie des probabilités, il y a la théorie des possibilités [20, 8]. Cette théorie, très répandue, est basée sur la notion de distribution de possibilités π qui pour chaque état, de l'univers du discours, associe un degré dans l'intervalle $[0, 1]$ exprimant une connaissance partielle sur l'état du monde. Elle est aussi régie par trois axiomes qui sont la normalisation (il doit exister un état du monde qui soit totalement possible, c'est à dire avec un degré de 1), la non-négativité et la maxitivité. Par convention, $\pi(\omega_i)=1$ correspond à un état complètement possible alors que $\pi(\omega_i)=0$ correspond à un état impossible. Notons que nous pouvons interpréter les possibilités soit i) *qualitativement* (de manière ordinale) où seul l'ordre des interprétations a de l'importance, soit ii) *quantitativement* où les degrés prennent un sens sur l'intervalle comme en probabilités.

Une notion est présente dans ces deux théories, celle de mesure de probabilités (resp. de possibilités), elle est définie comme étant le degré de probabilités (resp. de possibilités) d'un ensemble d'interprétations (états du monde). Cette mesure, notée P en probabilités et Π en possibilités se calcule à l'aide de l'axiome d'additivité (en probabilités, où l'on somme les degrés des interprétations appartenant à l'ensemble) et avec l'axiome de maxitivité (en possibilités, où le degré de l'ensemble est le degré maximal des interprétations de l'ensemble). La différence principale entre la théorie des probabilités et la théorie des possibilités se trouve donc dans la façon de calculer le degré d'une disjonction de deux événements. En effet, $\forall \phi, \psi \subseteq \Omega$ avec $\phi \cap \psi = \emptyset$, $P(\phi \cup \psi) = P(\phi) + P(\psi)$ alors que $\Pi(\phi \cup \psi) = \max(\Pi(\phi), \Pi(\psi))$.

Le conditionnement est un important opérateur de changement de croyances qui consiste à mettre à jour les connaissances actuelles, encodées soit par une distribution de probabilités soit par une distribution de possibilités, lorsqu'un nouvel événement sûr est observé. Alors que le conditionnement quantitatif en possibilités et le conditionnement probabiliste se ressemblent (en effet, ils sont définis de manière similaire), le conditionnement qualitatif, quant à lui, diffère

de manière significative. Remarquons que les deux définitions de conditionnement possibiliste suivantes satisfont la condition : $\forall \omega \in \phi, \pi(\omega) = \pi(\omega|\phi) \otimes \Pi(\phi)$ où \otimes est soit le produit soit l'opérateur min. Dans le cadre quantitatif, la règle de conditionnement (aussi connue comme la règle de Dempster) est définie comme suit :

$$\pi(w_i|_p\phi) = \begin{cases} \frac{\pi(w_i)}{\Pi(\phi)} & \text{si } w_i \in \phi; \\ 0 & \text{sinon.} \end{cases} \quad (1)$$

Le conditionnement qualitatif est défini de la manière suivante [11] :

$$\pi(w_i|_m\phi) = \begin{cases} 1 & \text{si } \pi(w_i) = \Pi(\phi) \text{ et } w_i \in \phi; \\ \pi(w_i) & \text{si } \pi(w_i) < \Pi(\phi) \text{ et } w_i \in \phi; \\ 0 & \text{sinon.} \end{cases} \quad (2)$$

Travailler directement sur les distributions (de probabilités ou de possibilités) n'est pas pratique en terme de complexité (spatiale et temporelle). En effet, la taille de la distribution peut vite devenir trop volumineuse, nous avons donc besoin de représentations compactes. Les modèles graphiques [4] nous fournissent un outil efficace pour la représentation d'informations. Et parmi eux, on trouve les réseaux Bayésiens qui sont l'une des représentations compactes les plus utilisées.

Réseaux Bayésiens

Un réseau Bayésien [4] est défini par :

- **Une composante graphique** : Les variables et les arcs forment un graphe orienté acyclique (DAG). Chaque sommet du DAG représente une variable du problème modélisé et les arcs encodent les indépendances entre les variables.
- **Une composante numérique** : Chaque lien dans le DAG est quantifié par une distribution conditionnelle $p(A_i|par(A_i))$ pour chaque noeud A_i dans le contexte de ses parents $par(A_i)$. Pour les variables sans parents, on associe aussi une table locale a priori (sans parents).

Un réseau Bayésien code une distribution jointe qui se calcule avec la règle de chaînage.

$$p(A_1, \dots, A_n) = \prod_{i=1}^n p(A_i|par(A_i)) \quad (3)$$

Les réseaux Bayésiens ne servent pas uniquement à la représentation mais aussi au raisonnement (calculer des probabilités d'intérêt). Les modèles graphiques sont dotés de mécanismes d'inférence [4] pour répondre à des questions d'intérêt afin de réaliser certaines tâches comme l'explication, le diagnostic, la classification, etc.

Réseaux possibilistes

Un réseau possibiliste [3] est aussi spécifié par une composante graphique et une autre numérique mais où

les tables locales sont des distributions de possibilités. La règle de chaînage est définie comme suit :

$$\pi(A_1, \dots, A_n) = \otimes_{i=1..n} \pi(A_i | par(A_i)) \quad (4)$$

où \otimes est, selon le cadre possibiliste choisi, soit le produit soit l'opérateur min (autrement dit, $\otimes = *$ ou $\otimes = \min$).

A moins qu'il en soit précisé autrement, par la suite les résultats valent dans les deux cadres, quantitatif et qualitatif.

3 Transformations probabilistes - possibilistes

Dans cette section, nous présentons en premier les principaux principes des transformations probabilistes-possibilistes. En particulier, puisque la théorie des probabilités et la théorie des possibilités représentent différentes sortes d'incertitude, il faut se focaliser sur le principe de consistance défini par Zadeh [20] et redéfini par plusieurs auteurs comme Dubois et Prade [7].

3.1 Principes des transformations

Principe de consistance de Zadeh. Zadeh [20] décrit le principe de consistance entre probabilités et possibilités comme *"a high degree of possibility does not imply a high degree of probability, and a low degree of probability does not imply a low degree of possibility"*. Le degré de consistance entre une distribution de probabilités p et une distribution de possibilités π est défini par :

$$C_z(\pi, p) = \sum_{i=1..n} \pi_i * p_i \quad (5)$$

Cependant, Zadeh signale que $C_z(\pi, p)$ n'est pas une loi précise ou une relation entre les distributions de probabilités et de possibilités. *"Rather it is an approximate formalization of the heuristic observation that a lessening of the possibility of a event tends to lessen its probability, but not vice-versa"*.

En effet, le degré de consistance calculé est discutable [7, 12] puisqu'il est d'autant plus proche de 1 que ω_i a une probabilité non nulle ($\forall \omega_i \in \Omega$, si $p(\omega_i) > 0$ alors $\pi(\omega_i) = 1$). Nous pouvons donc avoir plusieurs distributions (différentes) mais *ayant le même degré de consistance*.

Principe de consistance de Dubois et Prade. Dubois et Prade ont défini le principe de consistance, différemment, utilisant ces trois postulats : [7]

- La *condition de consistance* : $P(\phi) \leq \Pi(\phi)$, $\forall \phi \subseteq \Omega$.

- La *préservation de la préférence* : $\forall(\omega_1, \omega_2) \in \Omega^2$, si $p(\omega_1) > p(\omega_2)$ alors $\pi(\omega_1) > \pi(\omega_2)$ et si $p(\omega_1) = p(\omega_2)$ alors $\pi(\omega_1) = \pi(\omega_2)$.
- Le principe du *maximum de spécificité* : Il faut chercher la distribution la plus spécifique qui satisfait les deux postulats précédents.

La notion de spécificité permet de sélectionner une distribution de possibilités en fonction de la quantité d'informations contenue dans cette distribution. Pour rappel, soient π_1 et π_2 deux distributions de possibilités, π_1 est dite plus spécifique que π_2 si $\forall \omega_i \in \Omega$, $\pi_1(\omega_i) \leq \pi_2(\omega_i)$.

3.2 Règles de transformations

Il existe plusieurs transformations dans la littérature. On peut citer : *Optimal transformation (OT)* [7], *Klir transformation (KT)* [12], *Symmetric transformation (ST)* [10] et *Variable transformation (VT)* [14]. La transformation dite "optimale" a été définie par Dubois et Prade [7] et garantit que la distribution de possibilités résultante est la plus spécifique. *OT* est définie par :

$$\pi_i = \sum_{j/p_j \leq p_i} p_j \quad (6)$$

Il est à noter qu'il existe des transformations du cadre possibiliste vers le cadre probabiliste [10, 12]. Il existe aussi des transformations vers d'autres cadres incertains [5], en particulier des transformations des probabilités imprécises vers d'autres théories, le but étant de diminuer la complexité des calculs. Dans cet article, nous ne nous focalisons pas sur une transformation en particulier mais sur les transformations des probabilités vers les possibilités en général. On parlera de transformations probabilistes-possibilistes pour désigner des probabilités aux possibilités.

4 Transformations et raisonnement

Le but de ce papier est d'étudier la commutativité des transformations sur les tâches de raisonnement. Un premier papier [18] a étudié cette question mais cherchait seulement à savoir si les distributions résultantes étaient *identiques*. Il a été montré qu'il n'existe pas de transformations qui donnent les mêmes distributions par rapport au conditionnement et à la marginalisation. Nous utilisons $\triangleright(p) = \pi$ pour désigner la transformation de la distribution de probabilités p en distribution de possibilités π satisfaisant le principe de préservation de préférence de Dubois et Prade. Dans la suite de ce papier, nous étudions la commutativité des transformations par rapport à i) l'ordre des événements quelconques et ii) deux opérations de raisonnement

qui sont la marginalisation et le conditionnement. Nous étudions ces questions particulières pour des raisons pratiques. En effet parmi les requêtes les plus utilisées dans les modèles probabilistes, on trouve les requêtes MPE (la recherche de l'explication (instanciation de toutes les variables) la plus probable étant donnée une observation) et les requêtes MAP (où étant donné une ou plusieurs observations, l'objectif est de trouver l'hypothèse la plus probable sur certaines variables) [4]. Pour être plus explicite, développons deux petits exemples.

Exemple 1. Soit $p(ABC)$ une distribution de probabilités sur trois variables binaires A, B et C . Ω , l'ensemble des états possibles, est le produit carthésien des domaines D_A, D_B, D_C . Etant donnée une observation, par exemple, $C = \neg c$, une requête MPE serait "quelle est l'instanciation des variables la plus probable étant donnée l'observation $C = \neg c$?". Ceci s'écrit en terme de probabilités : $\operatorname{argmax}_{\omega_i \in \Omega} (p(\omega_i, \neg c))$. En d'autres termes, parmi les états de Ω où $C = \neg c$, ie. l'ensemble $\{ab\neg c, a\neg b\neg c, \neg ab\neg c, \neg a\neg b\neg c\}$, quelle est l'interprétation qui a le plus haut degré ?

Exemple 2. Soit $p(ABC)$ une distribution de probabilités sur trois variables binaires A, B et C (l'ensemble des variables est noté \mathcal{A}). Etant donné un sous-ensemble de variables $E \subset \mathcal{A}$ qui représente l'observation, et e une instanciation de E . Etant donné un sous-ensemble de variables $Q \subset \mathcal{A} \setminus E$ qui représente les variables de requêtes. Alors, si $e = \neg c$, une requête MAP serait "quelle est l'instanciation (l'hypothèse) des variables de Q la plus probable étant donnée l'observation $e = \neg c$?". Ceci s'écrit en terme de probabilités : $\operatorname{argmax}_{q \in D_Q} (p(q|e))$. En d'autres termes, parmi les états du domaine de Q après conditionnement par l'évidence e , quelle est l'interprétation qui a le plus haut degré ?

Par conséquent, pour répondre à ces questions en utilisant les transformations probabilistes-possibilistes, il est nécessaire d'étudier la commutativité des transformations par rapport à la marginalisation et le conditionnement. En particulier, il est nécessaire d'analyser l'impact des transformations sur l'ordre des événements.

Nous considérons les opérations de changement sur les distributions comme décrit par la Figure 1. La première distribution de possibilités π'' est obtenue en appliquant l'opération (marginalisation, conditionnement...) puis en transformant par \triangleright . Et nous obtenons la seconde distribution de possibilités π' en transformant puis en appliquant l'opération. Notre objectif est de comparer ces deux distributions de possibilités et vérifier si elles encodent le même ordre entre les événements.

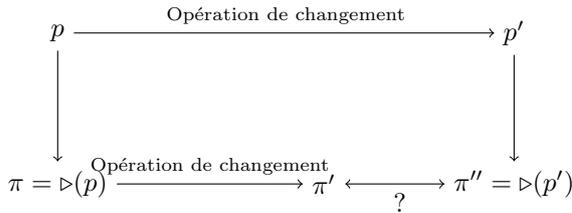


FIGURE 1 – Commutativité des tâches de raisonnement.

Considérons en premier l'opération de marginalisation qui consiste à construire les distributions marginales à partir d'une distribution jointe.

4.1 Transformations et marginalisation : préservation de l'ordre des événements

Dans la section précédente, nous avons vu que l'un des principes de Dubois et Prade requiert que l'ordre des interprétations doit être préservé, mais qu'en est-il de l'ordre des événements (ensemble d'interprétations)? Par exemple, est-ce que le principe de préservation de préférence suffit-il pour que la transformation préserve l'ordre d'événements arbitraires? Pour rappel, la marginalisation consiste à calculer le degré de probabilité ou de possibilité, d'un sous-ensemble de variables. Ainsi, soit $A = \{A_1 \dots A_k\}$ un sous-ensemble de variables de \mathcal{A} , la marginalisation en probabilité se calcule, grâce à l'axiome d'additivité, par : $P(A_1, \dots, A_k) = \sum_{A_{k+1} \dots A_n} (p(A_1 A_2 \dots A_n))$. Quant à la marginalisation en possibilités, elle se calcule via l'axiome de maxitivité. Ainsi, $\Pi(A_1, \dots, A_k) = \max_{A_{k+1} \dots A_n} (\pi(A_1 A_2 \dots A_n))$. Il s'agit de sommer¹ (ou maximiser) sur les variables restantes.

La Proposition 1 établit qu'il n'y a pas de transformation qui préserve l'ordre des événements.

Proposition 1. Soit \triangleright une transformation probabiliste-possibiliste². Alors il existe p une distribution de probabilités et $\triangleright(p) = \pi$ où il existe $\phi \subseteq \Omega$, $\psi \subseteq \Omega$, avec $\phi \neq \psi$ tel que :

$$P(\phi) < P(\psi) \not\Rightarrow \Pi(\phi) < \Pi(\psi)$$

La perte de l'ordre est due à la différence de comportement de l'axiome d'additivité du cadre probabiliste et de l'axiome de maxitivité du cadre possibiliste.

1. Si $p(ABC)$ est une distribution de probabilités conjointes aux trois variables binaires A , B et C , alors la distribution marginale $p(A)$ s'obtient depuis $p(ABC)$ en sommant pour obtenir $P(A=\neg a) = p(\neg abc) + p(\neg ab\neg c) + p(\neg a\neg bc) + p(\neg a\neg b\neg c)$. Et de manière similaire on obtient $P(A=a)$.

2. \triangleright est toujours supposée satisfaire les principes de préservation de préférence et de consistance de Dubois et Prade.

En conséquence de la Proposition 1, si l'univers du discours Ω est le produit cartésien de l'ensemble des domaines des variables, alors la marginalisation sur un sous-ensemble de variables ne préservera pas l'ordre relatif des événements après la transformation. Prenons un exemple afin d'illustrer notre propos.

Exemple 3. Soient deux variables binaires A et B . On considère la distribution de probabilités telle que représentée par la Table 1a. Nous pouvons marginali-

A	B	$p(A, B)$
a	b	0.4
a	$\neg b$	0.1
$\neg a$	b	0.3
$\neg a$	$\neg b$	0.2

(a)

A	B	$\pi(A, B)$
a	b	1
a	$\neg b$	x_1
$\neg a$	b	x_2
$\neg a$	$\neg b$	x_3

(b)

 TABLE 1 – Distribution de probabilités et sa transformée par \triangleright pour l'illustration de la Proposition 1

ser sur A ou B , ainsi $p(a) = 0.5$ et $p(\neg a) = 0.5$ et pour B , $p(b) = 0.7$ et $p(\neg b) = 0.3$. En transformant les marginales avec \triangleright , nous obtenons $\pi(a) = 1$ et $\pi(\neg a) = 1$ et pour B , $\pi(b) = 1$ et $\pi(\neg b) = y < 1$. Notons x_1, x_2, x_3 les degrés de possibilités (avec $x_2 > x_3 > x_1$) qui varient selon la transformation utilisée.

Maintenant, considérons la construction des distributions de possibilités marginales à partir de la transformation de la Table 1a, que l'on trouve dans la Table 1b. Les x_i sont ordonnées telles que : $x_1 < x_3 < x_2 < 1$. En marginalisant sur B , $\pi''(b) = 1$ et $\pi''(\neg b) = \max(x_1, x_3)$. En marginalisant sur A , $\pi''(a) = 1$ et $\pi''(\neg a) = \max(x_2, x_3)$. Par conséquent, en transformant les marginales, on a $\pi''(a) = \pi''(\neg a) = 1$ alors qu'en marginalisant la transformée, on a $\pi'(a) > \pi'(\neg a)$. Donc aucune transformation \triangleright préservant l'ordre des interprétations ne peut préserver la marginalisation.

4.2 Transformations et conditionnement

Dans cette partie, nous essayons de répondre à la question suivante : Étant donnée une distribution de probabilités p , une distribution de possibilités π (obtenue par transformation) et une observation, trouvons nous le même ordre après conditionnement en probabilités et en possibilités? La Proposition 2 montre que l'ordre des interprétations élémentaires est préservé après le conditionnement si la transformation utilisée préserve l'ordre des interprétations.

Proposition 2. Soit $\phi \subseteq \Omega$ une évidence, et \triangleright une transformation probabiliste-possibiliste. On suppose que p' est la distribution de probabilités obtenue en

conditionnant p par ϕ , $\pi'' = \triangleright(p')$ et π' est la distribution de possibilités obtenue en conditionnant $\pi = \triangleright(p)$ par ϕ . Alors, $\forall \omega_i, \omega_j \in \Omega$,

$$\pi'(\omega_i) < \pi'(\omega_j) \Leftrightarrow \pi''(\omega_i) < \pi''(\omega_j)$$

et

$$\pi'(\omega_i) = \pi'(\omega_j) \Leftrightarrow \pi''(\omega_i) = \pi''(\omega_j)$$

La Proposition 2 est valide dans les deux cadres possibilistes, quantitatif et qualitatif.

Idée de la preuve. Dans le cadre quantitatif comme dans le cadre qualitatif, la preuve est simple. Il suffit de considérer deux interprétations appartenant à l'ensemble des modèles de ϕ . Ainsi, soient deux interprétations $\omega_1, \omega_2 \in \phi$ (puisque si $\omega_i \notin \phi$ alors $p(\omega_i) = 0$, de même en possibilités) avec $p(\omega_1) < p(\omega_2)$, nous regardons après conditionnement l'ordre des deux interprétations, qui est préservé et puisque la transformation probabiliste-possibiliste \triangleright préserve l'ordre des interprétations, $\pi(\omega_1) < \pi(\omega_2)$. \square

Voyons-en un exemple dans le cadre quantitatif.

Exemple 4. Soit la distribution de probabilités de la Table 2, décrivant les croyances sur 3 variables A, B et C . Nous souhaitons conditionner par $\phi = \neg a$.

A	B	C	$p(A, B, C)$	$\pi(A, B, C)$
$\neg a$	$\neg b$	$\neg c$	0.216	α_1
$\neg a$	$\neg b$	c	0.27	1
$\neg a$	b	$\neg c$	0.144	α_2
$\neg a$	b	c	0.27	1
a	$\neg b$	$\neg c$	0.012	α_5
a	$\neg b$	c	0.048	α_3
a	b	$\neg c$	0.028	α_4
a	b	c	0.012	α_5

TABLE 2 – Distribution de probabilités et sa transformée par \triangleright où $1 > \alpha_1 > \alpha_2 > \alpha_3 > \alpha_4 > \alpha_5 > 0$.

On utilise la définition du conditionnement donnée dans la Section 2. Les résultats sont décrits par la Table 3, où on trouve dans la Table 3a, la distribution résultant du conditionnement puis de la transformation (ie. π''); Et dans la Table 3b, on trouve la distribution résultante de la transformation puis du conditionnement, autrement dit π' . Maintenant si l'on regarde la Table 3, nous pouvons voir que l'ordre est respecté puisqu'on a : $\alpha_1 < \alpha_2$ et $\beta_1 < \beta_2$.

Comme conséquence immédiate de la préservation de l'ordre des interprétations par l'opération de conditionnement, nous pouvons déduire de la Proposition 2 que pour les requêtes MPE, les réponses aux requêtes

$A B C$	$\pi''(ABC \neg a)$	$A B C$	$\pi'(ABC \neg a)$
$\neg a \neg b \neg c$	β_1	$\neg a \neg b \neg c$	α_1
$\neg a \neg b c$	1	$\neg a \neg b c$	1
$\neg a b \neg c$	β_2	$\neg a b \neg c$	α_2
$\neg a b c$	1	$\neg a b c$	1
$a \neg b \neg c$	0	$a \neg b \neg c$	0
$a \neg b c$	0	$a \neg b c$	0
$a b \neg c$	0	$a b \neg c$	0
$a b c$	0	$a b c$	0

(a)

(b)

TABLE 3 – Conditionnement quantitatif de la Table 2.

seront les mêmes en probabilités et en possibilités après la transformation, comme le stipule la proposition suivante :

Proposition 3. Soit p une distribution de probabilités et \triangleright une transformation préservant l'ordre des interprétations alors $\forall \phi \subseteq \Omega$,

$$\operatorname{argmax}_{\omega_i \in \Omega} (p(\omega_i, \phi)) = \operatorname{argmax}_{\omega_i \in \Omega} (\pi(\omega_i, \phi))$$

Idée de la preuve. L'idée de la preuve tient au fait que la réponse aux requêtes MPE dépend uniquement de l'ordre des interprétations qui est préservé par la transformation par hypothèse. \square

La Proposition 3 affirme que les requêtes MPE ont les mêmes réponses dans les deux cadres. Toutefois, lorsqu'on s'intéresse à des requêtes de type MAP, nous n'obtenons pas les mêmes réponses, cette perte est due à la marginalisation. En effet, les requêtes MAP sont évaluées en utilisant le conditionnement et la marginalisation. La réponse aux requêtes MAP n'est pas la même à cause de la marginalisation qui ne préserve pas l'ordre des événements (voir Proposition 1).

Proposition 4. Soit $\mathcal{A} = \{A_1, A_2, \dots, A_n\}$ un ensemble de variables discrètes. Soit $E \subset \mathcal{A}$, un sous-ensemble de variables observées et soit e la valeur observée pour E ($E=e$). Soit un autre sous-ensemble de variables $Q \subset \mathcal{A}$ tel que $Q \cap E = \emptyset$ et $Q \cup E \neq \mathcal{A}$. Il existe p une distribution de probabilités et π la distribution de possibilités résultant de la transformation \triangleright , tel que :

$$\operatorname{argmax}_{q \in D_Q} (p(q|e)) \neq \operatorname{argmax}_{q \in D_Q} (\pi(q|e))$$

avec D_Q le domaine associé aux variables de Q (produit cartésien des domaines des variables $A_i \in Q$).

4.3 Relations d'indépendance et transformations

La nature incomplète et incertaine des informations que l'on traite nous amène à nous intéresser à la notion

d'indépendance, qui est une notion très importante dans les modèles graphiques. Pour rappels, soient δ , ϕ et ψ trois événements, en probabilités (resp. en possibilités), ϕ est dit indépendant de ψ sachant δ si et seulement si $P(\phi|\psi\delta)=P(\phi|\delta)$ (resp. $\Pi(\phi|\psi\delta)=\Pi(\phi|\delta)$).

Cette section analyse si les relations d'indépendance entre événements sont préservées lors des transformations. Bien évidemment, le concept d'indépendance est lié à ceux du conditionnement et de la marginalisation. La Proposition 5 montre qu'il n'existe pas de transformation \triangleright préservant les relations d'indépendance.

Proposition 5. *Soient 3 événements ϕ, ψ et $\delta \subseteq \Omega$. Soit \triangleright une transformation probabiliste-possibiliste, alors il existe une distribution de probabilités p et $\pi = \triangleright(p)$ tel que,*

$$P(\phi|\psi\delta) = P(\phi|\delta) \not\Rightarrow \Pi(\phi|\otimes\psi\delta) = \Pi(\phi|\otimes\delta)$$

Dans la Proposition 5, $|\otimes$ représente soit le conditionnement quantitatif soit le conditionnement qualitatif. Les deux exemples suivants démontrent la perte des relations d'indépendance.

Exemple 5 (Cadre qualitatif). *Soit p une distribution probabiliste et $\pi = \triangleright(p)$ la distribution possibiliste. La Table 4 décrit les deux distributions où :*

A et B sont indépendants : $P(A|B)=P(A)$ et $P(B|A)=P(B)$.

A	B	$p(A, B)$	$\pi(A, B)$
a	b	0.12	β_4
a	$\neg b$	0.18	β_3
$\neg a$	b	0.28	β_2
$\neg a$	$\neg b$	0.42	1

TABLE 4 – Distribution de probabilités et la distribution de possibilités obtenue par transformation.

Puisque la transformation \triangleright préserve l'ordre des interprétations alors $1 > \beta_2 > \beta_3 > \beta_4$.

Il s'agit maintenant de vérifier si $\Pi(A|_{\min}B)=\Pi(A)$. En particulier, pour $\Pi(a|_{\min}b)$ et $\Pi(a)$ on trouve :

- $\Pi(a|_{\min}b)=\beta_4$
- $\Pi(a)=\beta_3$

On peut conclure que puisqu'il n'y a pas d'égalité, l'indépendance n'est pas préservée.

Exemple 6 (Cadre quantitatif). *Soit p une distribution probabiliste et $\pi = \triangleright(p)$ la distribution possibiliste. La Table 5 décrit les deux distributions où :*

- les événements $\phi = \{\omega_3, \omega_4, \omega_5\}$ and $\psi = \{\omega_2, \omega_3, \omega_4, \omega_5\}$ sont indépendants en probabilités. En effet, $P(\phi | \psi) = \frac{P(\phi \cap \psi)}{P(\psi)} = 0.3 = P(\phi)$.

ω_i	$p(\omega_i)$	$\pi(\omega_i)$
ω_1	0.42	1
ω_2	0.18	α_1
ω_3	0.12	α_2
ω_4	0.10	α_3
ω_5	0.08	α_4
ω_6	0.05	α_5
ω_7	0.05	α_5

TABLE 5 – Distribution de probabilités et distribution de possibilités obtenue par transformation.

Puisque la transformation \triangleright préserve l'ordre des interprétations alors $1 > \alpha_1 > \alpha_2 > \alpha_3 > \alpha_4 > \alpha_5$.

Le calcul de $\Pi(\phi | \psi)$ et de $\Pi(\phi)$ nous donne :

- $\Pi(\phi | \psi) = \frac{\Pi(\phi \cap \psi)}{\Pi(\psi)} = \frac{\alpha_2}{\alpha_1}$
- $\Pi(\phi) = \alpha_2$

Mais, $\frac{\alpha_2}{\alpha_1} \neq \alpha_2$ puisque $\alpha_1 < 1$. Alors l'équation $\Pi(\phi | \psi) = \Pi(\phi)$ ne tient pas, donc la préservation des événements non plus.

On peut déduire que puisque les indépendances d'événements ne sont pas préservées, alors les indépendances de variables ne sont pas préservées non plus.

La perte des relations d'indépendance entre les variables dans la distribution jointe pose un vrai problème en terme de perte d'informations, c'est pour cela que l'on s'intéresse maintenant à transformer directement un modèle graphique, le but étant de préserver ces relations d'indépendance.

5 Transformations dans les modèles graphiques

Transformer un modèle graphique probabiliste, ici un réseau Bayésien, vers un réseau possibiliste [16] consiste à transformer les tables locales et conserver la structure qui encode les relations d'indépendance conditionnelle.

Définition 1. *Soit un réseau Bayésien \mathcal{BN} sur un ensemble de variables $\mathcal{A}=\{A_1, \dots, A_n\}$, le réseau possibiliste \mathcal{PN} sur le même ensemble de variables \mathcal{A} est le réseau résultant de la transformation, il est composé de :*

- Une composante graphique qui est la même que le réseau \mathcal{BN} .
- Une composante numérique, qui est telle que l'ensemble des tables locales de \mathcal{PN} sont celles de \mathcal{BN} transformées par \triangleright .

L'avantage de transformer directement un réseau Bayésien en réseau possibiliste utilisant la Définition 1, est que l'on préserve les indépendances, et que calculatoirement cela est moins coûteux de transformer un

ensemble de tables locales que de transformer une distribution jointe. Le problème maintenant est qu'il n'y a pas de garantie que l'ordre des interprétations et des événements soient préservés dans le réseau possibiliste résultant et sa distribution jointe. La Figure 2 illustre la question de la transformation d'un réseau probabiliste (Bayésien) en réseau possibiliste.

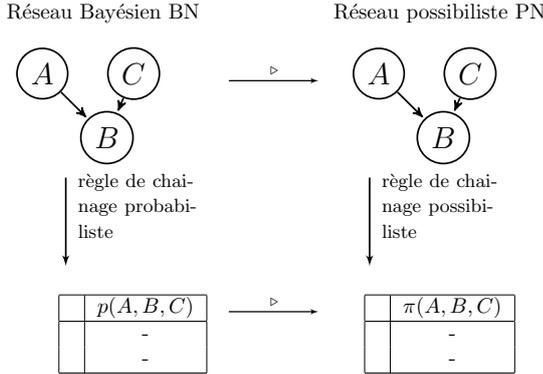


FIGURE 2 – Transformation de modèles graphiques

Nous nous intéressons donc à la différence entre l'ordre induit par la règle de chaînage probabiliste et l'ordre induit par la transformation puis le chaînage possibiliste. La Proposition 6 répond à la question : *Est-ce que l'ordre induit par la distribution $p_{\mathcal{BN}}$ (distribution jointe encodée par \mathcal{BN}) est préservé dans la distribution $\pi_{\mathcal{PN}}$ (la distribution jointe encodée par le réseau possibiliste \mathcal{PN}) ?*

Proposition 6. *Soit \triangleright une transformation probabiliste-possibiliste. Alors il existe un réseau Bayésien \mathcal{BN} tel que $\exists \omega_1 \in \Omega, \exists \omega_2 \in \Omega$ où :*

$$\pi'(\omega_1) < \pi'(\omega_2) \not\Rightarrow \pi''(\omega_1) < \pi''(\omega_2)$$

où : *i) $\pi'(\omega) = \triangleright(p(\omega))$, et p est la distribution jointe induite par \mathcal{BN} et ii) π'' est la distribution jointe induite par \mathcal{PN} en utilisant la Définition 1.*

Le contre-exemple qui suit prouve la Proposition 6 dans le cadre qualitatif.

Exemple 7. *Soit \mathcal{BN} le réseau Bayésien de la Figure 3 sur les variables A et B . Notons que la distribution de probabilités $p(A)$ dans \mathcal{BN} est une permutation³ de la distribution de probabilités $p(B)$. Ainsi, la transformation de $p(A)$ et $p(B)$ par \triangleright nous donne respectivement $\pi(A)$ et $\pi(B)$ où $\pi(B)$ est aussi une permutation de $\pi(A)$. Dans cet exemple, puisque \triangleright est supposé préserver l'ordre des interprétations, nous avons $1 > \alpha_1 > \alpha_2 > \alpha_3$.*

3. La propriété de permutation dans les transformations probabiliste-possibiliste est discutée dans [18].

A	$p(A)$	$\pi(A)$
a_1	0.4	1
a_2	0.2	α_2
a_3	0.25	α_1
a_4	0.15	α_3

(A)

B	$p(B)$	$\pi(B)$
b_1	0.15	α_3
b_2	0.2	α_2
b_3	0.25	α_1
b_4	0.4	1

(B)

FIGURE 3 – Exemple de transformation de réseaux Bayésiens-possibilistes.

Regardons de près les degrés de probabilités et de possibilités des interprétations a_1b_1 et a_2b_2 :

$$\begin{aligned} & - p(a_1b_1) = 0.4 * 0.15 = 0.06 \\ & - p(a_2b_2) = 0.2 * 0.2 = 0.04 \\ & \text{alors, } p(a_1b_1) > p(a_2b_2) \quad (*_1) \\ & - \pi(a_1b_1) = \alpha_3 \\ & - \pi(a_2b_2) = \alpha_2 \end{aligned}$$

alors, $\pi(a_1b_1) < \pi(a_2b_2)$ $(*_2)$
 A partir de $(*_1)$ et $(*_2)$ il est clair que l'ordre de ces deux interprétations est inversé peu importe la transformation qui sera utilisée.

De la même manière, dans le cadre quantitatif, nous pouvons prouver que l'ordre ne sera pas préservé quelque soit la transformation utilisée.

Pour résumer, si on transforme une distribution de probabilités jointe induite par \mathcal{BN} , on préserve l'ordre des interprétations mais cela est coûteux (la taille ie. le nombre d'interprétations de la distribution jointe est exponentielle dans le nombre de variables de \mathcal{BN}) et on perd les relations d'indépendance. Si on transforme directement un réseau Bayésien selon la Définition 1, on préserve les indépendances, c'est moins coûteux mais on risque de perdre l'ordre des interprétations.

6 Discussions et conclusions

Ce travail préliminaire traite des problèmes des transformations probabilistes-possibilistes, plus précisément des problèmes lors du raisonnement et des problèmes des transformations dans les modèles graphiques. Nous avons montré qu'il n'existe pas de transformation préservant l'ordre des interprétations qui puisse préserver l'ordre des événements. En plus de cela, nous avons montré que pour l'opération de marginalisation, nous ne pouvons pas trouver de transformation qui nous donne le même ordre sur les événements, lorsqu'on transforme puis marginalise ou l'inverse (que l'on marginalise puis que l'on transforme). Quant à la question des relations d'indépendance entre variables ou événements, le papier montre que les transformations ne préservent pas les indépendances. Enfin, dans la dernière partie, nous nous sommes penchés sur les transformations au sein des modèles graphiques. Malgré le fait que les relations d'indépen-

dance soient préservées, si nous regardons les distributions jointes, l'ordre des interprétations ne sera pas toujours le même.

Dans la littérature, il existe principalement deux travaux qui traitent des mêmes problèmes que dans le présent article. Dans le premier travail [16], les auteurs étudient les transformations de réseaux Bayésiens vers les réseaux possibilistes. Ils étendent la définition du principe de consistance, afin de préserver l'ordre des interprétations dans les distributions obtenues après transformation. Notons que ce travail s'intéresse plus aux transformations existantes (comme *OT* et *ST*) alors que nous avons essayé de travailler de manière générale en considérant toutes les transformations vérifiant le principe de préservation de préférences de Dubois et Prade. Le second travail [18], quant à lui, a étudié cette question des transformations à travers les tâches de raisonnement. Son objectif était de montrer que les distributions résultantes n'étaient pas identiques (au niveau des degrés de possibilités). Dans notre travail, nous nous sommes plus concentrés à montrer qu'en plus d'être différentes, les distributions n'induisent pas le même ordre entre les événements. Certaines de ces questions ont été étudiées dans le contexte de l'analyse d'intervalles flous [9].

Une question qui mérite d'être développée est de savoir s'il existe des formes particulières de distributions de probabilités qui pour n'importe quelles transformations \triangleright préserveraient l'ordre des événements (après la marginalisation, par exemple). L'idée la plus triviale concerne les distributions de probabilités uniformes. En effet, toute transformation \triangleright doit préserver la normalisation, ce qui donne une distribution de possibilités uniforme (où chaque état est associé à un degré de possibilité égal à 1). Par conséquent, les tâches de raisonnement appliquées aux réseaux ou aux distributions ne modifieront pas les connaissances (e.g. le sens des interprétations). Entre autres, les distributions marginales que l'on pourra calculer seront des distributions uniformes. Dans les modèles graphiques, l'ordre des interprétations dans les distributions jointes sera le même. Un autre type de distributions de probabilités est appelé : "atomic bound system" [17] ou encore "big-stepped" [6, 2] ou "lexicographic probability distributions", une telle distribution p est définie par : $\forall \omega_i \in \Omega, p(\omega_i) > \sum \{p(\omega_j) : \omega_j \in \Omega \text{ et } p(\omega_j) < p(\omega_i)\}$. Clairement, si p est une distribution "big-stepped" alors la transformation \triangleright préserve l'ordre entre les interprétations après marginalisation. Notons que pour les deux types de distributions décrits ci-dessus (uniforme et big-stepped), l'ordre entre les événements non élémentaires n'est pas préservé.

Il est connu que les transformations probabilistes-possibilistes souffrent d'une perte d'informations puis-

qu'on passe d'un cadre additif à un cadre qualitatif ou semi-qualitatif. Mais l'impact sur les tâches de raisonnement sur les distributions n'était pas complètement étudié. Les résultats obtenus confirment cette perte d'information à plusieurs niveaux. Ce n'est pas pour autant qu'on ne peut rien faire avec les transformations. En particulier, les réponses aux requêtes MPE ne sont pas affectées par les transformations, contrairement aux requêtes MAP. Nos travaux futurs étudieront en particulier l'inférence MAP dans les réseaux crédaux (basés sur les probabilités imprécises et connus pour leur très grande complexité en comparaison avec les réseaux Bayésiens ou possibilistes) en les transformant en réseaux possibilistes. Cette piste peut apporter une bonne et efficace approximation pour l'inférence MAP avec un meilleur coût de calcul. D'autres perspectives sont envisageables, notamment concernant la commutativité des transformations avec d'autres définitions de conditionnement et d'indépendances dans le cadre possibiliste.

Références

- [1] C. Baudrit, I. Couso, and D. Dubois. Joint propagation of probability and possibility in risk analysis : Towards a formal framework. *International Journal of Approximate Reasoning*, 45(1) :82 – 105, 2007.
- [2] S. Benferhat, D. Dubois, and H. Prade. Possibilistic and standard probabilistic semantics of conditional knowledge bases. *J. Log. Comput.*, 9(6) :873–895, 1999.
- [3] C. Borgelt, J. Gebhardt, and R. Kruse. Graphical models. In *In Proceedings of International School for the Synthesis of Expert Knowledge 98*, pages 51–68. Wiley, 2002.
- [4] A. Darwiche. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, 2009.
- [5] S. Destercke, D. Dubois, and E. Chojnacki. Transforming probability intervals into other uncertainty models. In *5th EUSFLAT Conference, Ostrava, Czech Republic, September 11-14, 2007, Volume 2 : Regular Sessions*, pages 367–373, 2007.
- [6] D. Dubois, H. Fargier, and H. Prade. Ordinal and probabilistic representations of acceptance. *J. Artif. Int. Res.*, 22(1) :23–56, July 2004.
- [7] D. Dubois, L. Foulloy, G. Mauris, and H. Prade. Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. *Reliable Computing*, 10(4) :273–297, 2004.
- [8] D. Dubois and H. Prade. *Possibility Theory : An Approach to Computerized Processing of Uncertainty*. Plenum Press, New York, 1988.

- [9] D. Dubois and H. Prade. Random sets and fuzzy interval analysis. *Fuzzy Sets and Systems*, 42(1) :87–101, 1991. DP165.
- [10] D. Dubois, H. Prade, and S. Sandri. On possibility/probability transformations. In *Fuzzy Logic*, pages 103–112. 1993.
- [11] E. Hisdal. Conditional possibilities independence and non interaction. *Fuzzy Sets and Systems*, pages 283–297, 1978.
- [12] G. J. Klir and J. F. Geer. Information-preserving probability- possibility transformations : Recent developments. In *Fuzzy Logic*, pages 417–428. 1993.
- [13] M.-H. Masson and T. Denoeux. Inferring a possibility distribution from empirical data. *Fuzzy Sets and Systems*, 157(3) :319–340, 2006.
- [14] P. Billaudel M.S Mouchaweh, M.S Bouguelid and B. Riera. Variable probability-possibility transformation. pages 417–428. September 2006.
- [15] H. Prade and D. Dubois. *Fuzzy Sets and Systems : Theory and Applications*. Academic Press, Inc., Orlando, FL, USA, 1997.
- [16] Y. B. Slimen, R. Ayachi, and N. B. Amor. Probability-possibility transformation : - application to bayesian and possibilistic networks. In *WILF*, volume 8256 of *Lecture Notes in Computer Science*, pages 122–130. Springer, 2013.
- [17] P. Snow. Standard probability distributions described by rational default entailment. 1996.
- [18] T. Sudkamp. On probability-possibility transformations. *Fuzzy Sets and Systems*, pages 73–81, 1992.
- [19] Y.Y. Wee, W.P. Cheah, S.C. Tan, and K. Wee. A method for root cause analysis with a bayesian belief network and fuzzy cognitive map. *Expert Syst. Appl.*, 42(1) :468–487, 2015.
- [20] L. A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 100 :9–34, 1999.

Typologie d’axiomes pour une logique modale pondérée

B. Legastelois¹ M.-J. Lesot¹ A. Revault d’Allonnes²

¹ Sorbonne Universités, UPMC Univ Paris 06, CNRS,
LIP6 UMR 7606, 4 place Jussieu 75005 Paris

² Université Paris 8 - EA 4383 - LIASD, FR-93526, Saint-Denis, France
Benedicte.Legastelois@lip6.fr

Résumé

Cet article se place dans le contexte des logiques modales pondérées et étudie la définition d’extensions pondérées des axiomes classiques. Il propose une sémantique reposant sur un comptage relatif des mondes, qui présente l’avantage de fournir des informations permettant de guider la définition des poids. Il discute la notion de valeurs de poids pertinentes, exploitant à la fois la structure des axiomes et leur interprétation sémantique, pour des cadres de Kripke dont la relation vérifie des contraintes comme la réflexivité ou la transitivité par exemple. Il établit une typologie d’axiomes pondérés, selon 4 catégories, dépendant de l’enrichissement apporté à la version classique de l’axiome ainsi que de l’éventuelle équivalence avec les contraintes sur la relation.

Abstract

In the framework of weighted modal logics, this paper studies the definition of weighted extensions for the classical modal axioms. It discusses the notion of relevant weight values, in a particular weighted Kripke’s semantics and exploiting the properties satisfied by the accessibility relation. Different generalisations of the classical axioms are constructed and, from these, a typology of weighted axioms is built, that distinguishes between four types depending on their relations to classic axioms and to the, possibly equivalent, frame conditions.

1 Introduction

Les extensions pondérées des logiques modales visent à en accroître l’expressivité et, par exemple, à permettre de raisonner sur des notions affinées comme les croyances graduelles. Elles rendent ainsi possible la représentation et la manipulation de degrés de croyance en des informations, pour déterminer à quel

point on les croit. Cette problématique clé s’inscrit dans le cadre de la représentation de croyances et de la modélisation du raisonnement sur celles-ci, mise en œuvre par exemple dans le cadre de nombreuses applications, notamment basées sur des agents intelligents. Le principe des logiques modales pondérées repose généralement sur l’enrichissement des opérateurs modaux, \Box et \Diamond , par l’intégration d’un degré à valeurs entières ou réelles. Les modalités pondérées ainsi proposées sont notées \Box_α et \Diamond_α , où α représente la valeur numérique du degré. Les variantes diffèrent principalement par le cadre théorique auquel les coefficients sont rattachés et leur interprétation, comme détaillé en section 2.

Dans ce cadre, cet article considère les axiomes des logiques modales classiques interprétés comme des règles de manipulation et combinaison des modalités : il en propose des extensions pondérées, établissant des relations entre des formules contenant ces modalités.

Ainsi, l’axiome classique (4), qui s’écrit $\vdash \Box\varphi \rightarrow \Box\Box\varphi$, établit une implication entre une simple occurrence et une répétition de \Box . De même, l’axiome (D), $\vdash \Box\varphi \rightarrow \Diamond\varphi$, induit une relation entre les deux opérateurs modaux. Cet article s’intéresse à la transposition de ces axiomes au cas de la logique modale pondérée, afin de proposer des règles de manipulation adaptées aux modalités \Box_α et \Diamond_α .

L’article propose d’abord une sémantique pour \Box_α et \Diamond_α dans le cadre de la sémantique de Kripke, reposant sur un comptage relatif des mondes, établissant ainsi un relâchement de la quantification universelle de la modalité classique \Box . Cette sémantique présente l’avantage d’être suffisamment informative pour guider la définition des axiomes pondérés. Ceux-ci sont

établis à partir des axiomes classiques, en remplaçant chaque modalité classique par une modalité pondérée, chacune ayant un poids qui lui est propre. Nous proposons alors une discussion sur les valeurs possibles de ces poids et leurs liens. Ce principe peut par exemple être illustré dans le cas de l’axiome (D) : sa version pondérée s’écrit $\vdash \Box_\alpha \varphi \rightarrow \Diamond_\beta \varphi$, la problématique est alors d’établir une valuation pertinente de β dépendant de celle de α .

Nous proposons de traiter cette problématique selon une approche sémantique, dans le cadre de la sémantique de Kripke étendue proposée. Cette approche permet d’identifier des liens entre les poids en fonction de la relation d’accessibilité et de ses propriétés. Réciproquement, nous nous intéressons ensuite à la caractérisation de la relation dans le cas où l’axiome est vérifié dans un modèle l’utilisant. Ainsi nous considérons la problématique des axiomes pondérés dans le contexte de la théorie de la correspondance. L’objectif n’est cependant pas de construire une axiomatisation à partir de la sémantique proposée pour la modalité pondérée mais d’étudier la transposition des axiomes classiques au cas pondéré, c’est-à-dire la conséquence de l’introduction des pondérations sur règles de manipulation des modalités.

Cet article établit une typologie des axiomes modaux pondérés et distingue quatre catégories selon l’enrichissement apporté par la pondération et les propriétés de la relation sur laquelle se base sa valuation. Le type (I) regroupe les axiomes pour lesquels le relâchement de la modalité classique ne peut pas exploiter les poids tels que définis dans la sémantique proposée. Le type (II) comporte les axiomes dont la variante pondérée préserve la propriété de la relation classiquement associée à l’axiome. Les types (III) et (IV) regroupent les axiomes pondérés pour lesquels la propriété classique doit être enrichie, respectivement avec ou sans conservation de la correspondance.

L’article est organisé comme suit : la section 2 présente une étude comparative de modèles existants pour les logiques modales pondérées. La section 3 présente la sémantique que nous proposons pour les modalités pondérées, qui est exploitée dans la section 4 pour construire les axiomes pondérés. La section 5 présente la typologie résultante des axiomes ainsi établis.

2 Modèles existants pour des logiques modales pondérées

Après une présentation des notations utilisées dans cet article, cette section décrit brièvement des extensions pondérées de la logique modale existantes. Les approches basées sur des modifications du modèle de Kripke sont d’abord exposées dans les sections 2.2

et 2.3 : la pondération porte soit sur la relation, soit sur les mondes possibles. La section 2.4 présente ensuite les approches fondées sur un comptage des mondes, qui préservent la définition d’un modèle de Kripke mais modifient la quantification utilisée dans l’interprétation sémantique des modalités classiques.

2.1 Notations

Selon la notation usuelle (voir [1] par exemple), un cadre $F = \langle W, R \rangle$ est un couple composé d’un ensemble non vide W de mondes possibles et d’une relation binaire d’accessibilité entre les mondes, R . Un modèle de Kripke $\mathcal{M} = \langle F, s \rangle$ est un couple composé d’un cadre F et d’une valuation s qui, pour chaque monde de W , affecte une valeur de vérité à tous les atomes.

Étant donné un modèle \mathcal{M} et un monde w de W , on note R_w l’ensemble des mondes accessibles depuis w :

$$R_w = \{w' \in W \mid wRw'\} \quad (1)$$

De plus, selon la définition usuelle du symbole de conséquence sémantique \models , pour toute formule φ nous définissons l’ensemble $R_w(\varphi)$ des mondes accessibles depuis w et dans lesquels φ est vraie :

$$R_w(\varphi) = \{w' \in W \mid \mathcal{M}, w' \models \varphi\} \quad (2)$$

L’interprétation des formules $\Box\varphi$ et $\Diamond\varphi = \neg\Box\neg\varphi$ est basée sur la quantification, respectivement universelle et existentielle, des mondes accessibles validant φ . En utilisant les notations précédentes, cette interprétation est notée :

$$\mathcal{M}, w \models \Box\varphi \Leftrightarrow \forall w' \in R_w, \mathcal{M}, w' \models \varphi \quad (3)$$

$$\Leftrightarrow R_w(\varphi) = R_w$$

$$\mathcal{M}, w \models \Diamond\varphi \Leftrightarrow \exists w' \in R_w, \mathcal{M}, w' \models \varphi \quad (4)$$

$$\Leftrightarrow R_w(\varphi) \neq \emptyset$$

2.2 Pondération de la relation d’accessibilité

Un premier ensemble d’extensions pondérées de la logique modale est basé sur une modification des modèles de Kripke : la relation d’accessibilité R est remplacée par un ensemble de relations indexées R^α , avec $\alpha \in [0, 1]$. Cet ensemble permet de définir les modalités pondérées \Box_α , respectivement associées à chaque relation R^α , et interprétées selon les équations (3) et (4). Trois approches sont distinguées selon le cadre formel d’interprétation du degré α , qui peut être la théorie des probabilités, la théorie des possibilités [17] ou la théorie des sous-ensembles flous [16].

Dans le cas probabiliste [13], l’interprétation associée à la pondération de la relation d’accessibilité repose sur la probabilité d’une transition entre mondes.

La combinaison de poids se fait alors selon les propriétés de la théorie des probabilités.

Dans le cas flou [2], le poids sur la relation représente son intensité, exprimant qu’un monde est *plus ou moins* accessible depuis un autre : il décrit une imprécision quant à l’accessibilité entre les mondes. Il faut noter que de telles relations d’accessibilité imprécises correspondent, dans le formalisme flou, aux α -coupes d’une relation floue. Elles satisfont donc une propriété d’inclusion : en notant $R_w^\alpha = \{w' \in W \mid wR^\alpha w'\}$, on a

$$\forall \alpha, \beta \in [0, 1], w \in W, \text{ si } \alpha \geq \beta, \text{ alors } R_w^\alpha \subseteq R_w^\beta$$

Par conséquent, au niveau logique, les modalités pondérées dans le cas flou vérifient une propriété que nous appelons gradualité décroissante :

$$\forall \alpha, \beta \in [0, 1], \text{ si } \alpha \geq \beta \text{ alors } \models \Box_\alpha \varphi \rightarrow \Box_\beta \varphi \quad (5)$$

L’interprétation floue du degré α conduit donc à une logique multi-modale dont les modalités sont interdépendantes.

Dans le cas possibiliste [7], le poids sur la relation représente l’incertitude concernant l’accessibilité entre les mondes. Cela revient à exprimer un doute sur l’existence de cette accessibilité tandis que l’imprécision du modèle flou traduit une information sur son intensité. Au contraire du cas flou, l’approche possibiliste [7] induit un ensemble de modalités indépendantes.

2.3 Pondération des mondes

Une deuxième catégorie de logiques modales pondérées existantes fait porter la pondération sur les mondes accessibles plutôt que sur la relation. Il faut noter que, par conséquent, les poids ont un effet global sur le modèle : ils ne dépendent pas d’un monde de référence, ni de ses mondes accessibles. Au contraire, une relation pondérée produit un effet local car les poids sont spécifiques à chaque couple de mondes.

Les modèles de pondération possibiliste des mondes [6, 3] enrichissent les modèles de Kripke à l’aide d’une distribution de possibilités [17] sur l’ensemble W , notée π : les mondes sont alors considérés comme plus ou moins possibles.

π peut être généralisée aux formules, en définissant $\Pi(\varphi) = \max_{w \in W} \{\pi(w) \mid \mathcal{M}, w \models \varphi\}$ [6]. Ce modèle permet essentiellement de généraliser la logique possibiliste et en propose une interprétation épistémique.

La distribution de possibilités π peut aussi être utilisée pour définir la relation d’accessibilité [3] :

$$R_w = \{w' \in W \mid \pi(w) \leq \pi(w')\}$$

La sémantique de \Box et \Diamond est alors identique au cas classique, rappelé par les équations (3) et (4), utilisant cette relation d’accessibilité. Par conséquent, la

formule $\Box\varphi$ est vraie dans un monde w si et seulement si φ est vraie dans tous les mondes au moins aussi possibles que w . Il faut souligner que cette extension pondérée de la sémantique de Kripke n’introduit pas de pondération des modalités. On peut également noter que la relation d’accessibilité induite par π est nécessairement anti-symétrique, transitive et réflexive, ce qui contraint l’expressivité des modalités engendrées.

Les poids associés aux mondes peuvent également être considérés à un niveau expressif, reposant sur le sens que l’on veut leur donner et non explicitement sur un formalisme théorique, par exemple sous la forme de degrés d’exceptionnalité [11] : ils estiment à quel point un monde est inhabituel, ou non représentatif. Un degré d’exceptionnalité est alors défini pour chaque formule logique : $except(\varphi) = \min_{w \in W} \{except(w) \mid \mathcal{M}, w \models \varphi\}$. L’interprétation de la modalité pondérée induite ne conserve pas la définition classique de l’équation (3) mais est donnée par :

$$\mathcal{M}, w \models \Box_\alpha \varphi \Leftrightarrow except(\neg\varphi) \geq \alpha$$

Selon cette définition, plus la contradiction de φ est exceptionnelle, plus la pondération de $\Box_\alpha \varphi$ est élevée. On peut souligner deux propriétés qui découlent de cette définition. D’abord, la valeur de vérité d’une formule modale est globale et ne dépend pas du monde de référence w . En effet, $\mathcal{M}, w \models \Box_\alpha \varphi \Leftrightarrow \mathcal{M} \models \Box_\alpha \varphi$. De plus, en raison de l’inégalité sur laquelle la sémantique de \Box_α s’appuie, les modalités sont dépendantes : plus précisément, la propriété de gradualité décroissante donnée dans l’équation (5) s’applique également pour ce modèle.

2.4 Approche par comptage

L’approche par comptage ne propose pas de variante enrichie des cadres de Kripke : ni les mondes possibles, ni la relation d’accessibilité ne sont pondérés mais l’interprétation de la modalité est modifiée [10, 9, 8, 15]. Contrairement aux modèles présentés précédemment, les poids considérés ici sont à valeur entière et notés n .

L’approche par comptage modifie la quantification de mondes accessibles sur laquelle se base la sémantique classique des modalités. Ainsi, l’interprétation de \Diamond_n s’appuie sur un durcissement du quantificateur existentiel de l’équation (4) : il n’est plus imposé qu’au moins un monde accessible valide la formule mais qu’au moins n mondes le fassent. Formellement, cette approche propose d’interpréter la modalité \Diamond_n , et par dualité \Box_n , pour tout $n \in \mathbb{N}$, par :

$$\mathcal{M}, w \models \Diamond_n \varphi \Leftrightarrow |R_w(\varphi)| \geq n \quad (6)$$

$$\mathcal{M}, w \models \Box_n \varphi \Leftrightarrow |R_w(\neg\varphi)| < n \quad (7)$$

On peut noter que le poids n de \Box_n correspond au nombre maximum de mondes accessibles invalidant la

formule. Par conséquent, n peut être interprété comme une mesure de contradiction.

Cette définition s'appuie sur un comptage absolu des mondes là où la logique majoritaire [12] s'intéresse à un cas particulier de comptage relatif. Pour celle-ci en effet l'opérateur modal induit exprime qu'une formule est vraie dans plus de la moitié des mondes accessibles. Elle repose donc sur une approche par comptage relatif mais définit des modalités non pondérées. Elle aborde également la question d'une sémantique dans laquelle l'ensemble des mondes W est infini [12].

Contrairement aux approches décrites dans la section précédente, basées sur une modification des modèles de Kripke, l'approche par comptage est de plus axiomatisée, tant dans le cas absolu que relatif [5, 12] : ces modèles proposent un ensemble de règles de manipulation de la modalité pondérée.

3 Sémantique proposée

Cette section présente la sémantique que nous proposons pour une logique modale pondérée, un outil essentiel de l'extension des axiomes au cas pondéré, discutée en sections 4 et 5.

Syntaxiquement, étant donné $p \in \mathbb{P}$, l'ensemble des variables propositionnelles, et $\alpha \in [0, 1]$, un coefficient numérique, on considère l'ensemble des formules du langage de la logique modale pondérée :

$$F := p \mid \neg F \mid F \wedge F \mid F \vee F \mid F \rightarrow F \mid \Box_\alpha F \mid \Diamond_\alpha F$$

3.1 Définition

La sémantique proposée s'appuie sur le même principe que l'approche par comptage (cf. section 2.4), mais est basée sur un comptage relatif des mondes validants : elle est définie pour un ensemble W fini selon une interprétation fréquentiste, le cardinal des mondes validants est normalisé. Malgré une limitation aux ensemble finis de mondes, cette normalisation permet d'enrichir l'information portée par la modalité pondérée. Ainsi, la nouvelle modalité et sa sémantique permettent d'établir des variantes pondérées des axiomes classiques, discutées en sections 4 et 5. De plus, l'utilisation de cette proportion induit l'indépendance entre la valeur du poids considéré et la connexité du monde de référence : la valeur de vérité d'une formule $\Box_\alpha \varphi$ dans un monde w n'est pas biaisée par la valeur de $|R_w|$, le nombre de mondes accessibles depuis w .

Formellement, nous proposons d'interpréter la modalité pondérée \Box_α , pour tout $\alpha \in [0, 1]$, par :

$$\begin{cases} \mathcal{M}, w \models \Box_\alpha \varphi \Leftrightarrow \frac{|R_w(\varphi)|}{|R_w|} \geq \alpha & \text{si } R_w \neq \emptyset \\ \mathcal{M}, w \models \Box_\alpha \varphi & \text{sinon} \end{cases} \quad (8)$$

Cette définition relâche le quantificateur universel utilisé pour la modalité classique \Box , en imposant seulement qu'une proportion de mondes accessibles valident la formule φ , plutôt que l'ensemble de ces mondes.

Par dualité, la modalité \Diamond_α est interprétée, pour tout $\alpha \in [0, 1]$, par :

$$\begin{cases} \mathcal{M}, w \models \Diamond_\alpha \varphi \Leftrightarrow \frac{|R_w(\varphi)|}{|R_w|} > 1 - \alpha & \text{si } R_w \neq \emptyset \\ \mathcal{M}, w \not\models \Diamond_\alpha \varphi & \text{sinon} \end{cases} \quad (9)$$

La modalité \Diamond_α nécessite d'avoir plus qu'une proportion $1 - \alpha$ de mondes accessibles validant la formule φ plutôt qu'au moins un monde accessible. De même que les approches par comptage présentées en section 2.4, cette sémantique induit un durcissement du quantificateur existentiel. On peut noter que plus la valeur de α est élevée, plus la contrainte sur l'ensemble des mondes validants est faible.

3.2 Propriétés

Cette section présente et discute quelques propriétés satisfaites par la modalité pondérée proposée.

Cas limite

La proposition suivante traite les modalités pondérées pour les valeurs aux bornes du domaine de α . Pour le cas où $\alpha = 1$, les modalités pondérées correspondent aux modalités classiques. Pour le cas où $\alpha = 0$, $\Box_\alpha \varphi$ est une tautologie tandis que $\Diamond_\alpha \varphi$ est une contradiction. Formellement, pour toute formule φ , on a :

$$\begin{array}{ll} \text{Proposition 1.} & \Box_1 \varphi = \Box \varphi & \Diamond_1 \varphi = \Diamond \varphi \\ & \models \Box_0 \varphi & \models \neg \Diamond_0 \varphi \end{array}$$

La preuve de cette proposition découle de façon immédiate des équations (8) et (9) et est donc omise.

Du fait de cette propriété, le cas où $\alpha = 0$ est considéré comme trivial et non informatif ; il est donc exclu. Cependant, le cas où $\alpha = 0$ est la seule valeur pour laquelle la formule modale est valide exprime une information importante. En effet, pour un monde $w \in W$ tel que $R_w \neq \emptyset$ et $|R_w(\varphi)|/|R_w| = 0$, on a $\mathcal{M}, w \models \Box_1 \neg \varphi$.

Gradualité décroissante

En raison de la transitivité de l'inégalité sur laquelle repose \Box_α , on a la même relation de gradualité que le modèle de pondération floue de la relation :

Proposition 2. *La définition de \Box_α donnée en équation (8) satisfait la propriété de gradualité décroissante définie dans l'équation (5).*

De même que pour la proposition précédente, la preuve découlant directement de la définition des modalités est omise.

Cette proposition implique que vraie pour un degré, la formule est aussi vraie pour tous les degrés inférieurs. On peut noter que cette propriété apporte une autre justification à l'absence d'information apportée par \Box_0 : s'il existe $\alpha > 0$ tel que $\Box_\alpha \varphi$ est satisfait, alors $\Box_0 \varphi$ l'est aussi, par gradualité. Plus généralement, il apparaît que la valeur la plus informative pour α est la plus élevée pour laquelle la formule modale est vraie, les valeurs inférieures pouvant en être inférées. Cette propriété est essentielle pour l'établissement des extensions pondérées des axiomes modaux, comme discuté en section 4.

Par dualité, on peut montrer un résultat similaire pour la modalité \Diamond_α , qui respecte la propriété de gradualité croissante. Aussi, plus la valeur de α est faible, plus $\Diamond_\alpha \varphi$ est informative.

Relations entre \Box_α et \Diamond_α

Il faut noter que malgré la relation de dualité préservée entre les deux opérateurs modaux, $\Box_\alpha \varphi = \neg \Diamond_\alpha \neg \varphi$, l'équivalence, qui pourrait être attendue, entre \Box_α et $\Diamond_{1-\alpha}$ n'est pas garantie. En effet, en raison de la définition de \Box_α qui se base sur une inégalité large et celle de \Diamond_α qui se base sur une inégalité stricte, on peut montrer que l'implication n'est vraie que dans un sens. Le cas où $\alpha = 0$ est traité par la Proposition 1.

Proposition 3.

$$\begin{aligned} \forall \alpha \in]0, 1] \quad & \models \Diamond_\alpha \varphi \rightarrow \Box_{1-\alpha} \varphi \\ & \not\models \Box_\alpha \varphi \rightarrow \Diamond_{1-\alpha} \varphi \end{aligned}$$

Démonstration. Soit un modèle $\mathcal{M} = \langle \langle W, R \rangle, s \rangle$ et $w \in W$. On sait que

$$\begin{aligned} \mathcal{M}, w \models \Diamond_\alpha \varphi & \Leftrightarrow R_w \neq \emptyset \text{ et } \frac{|R_w(\varphi)|}{|R_w|} > 1 - \alpha \\ & \Rightarrow R_w \neq \emptyset \text{ et } \frac{|R_w(\varphi)|}{|R_w|} \geq 1 - \alpha \\ & \Rightarrow \mathcal{M}, w \models \Box_{1-\alpha} \varphi \end{aligned}$$

En revanche, l'implication $\Box_\alpha \varphi \rightarrow \Diamond_{1-\alpha} \varphi$ n'est pas une tautologie : on peut le montrer par le contre-exemple de la figure 1. Dans ce modèle de Kripke, on a $w \models \Box_{2/3} \varphi$ mais $w \not\models \Diamond_{1/3} \varphi$, car $|R_w(\varphi)|/|R_w| = 1/3$ ne satisfait pas l'inégalité stricte. ■

Une autre proposition établit une équivalence entre la modalité classique \Diamond et \Box_α :

Proposition 4. *Pour tout modèle $\mathcal{M} = \langle \langle W, R \rangle, s \rangle$ et pour tout monde $w \in W$,*

$$\mathcal{M}, w \models \Diamond_1 \varphi \Leftrightarrow R_w \neq \emptyset \text{ et } \mathcal{M}, w \models \Box_{\frac{1}{|R_w|}} \varphi$$

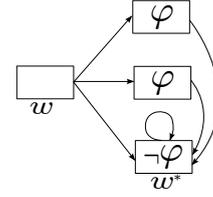


FIGURE 1 – Modèle de Kripke pour prouver la Prop.3, puis le Th. 3 pour $\alpha = \frac{2}{3}$

Démonstration. Soit un modèle $\mathcal{M} = \langle \langle W, R \rangle, s \rangle$ et $w \in W$. On a :

$$\begin{aligned} \mathcal{M}, w \models \Diamond_1 \varphi & \Leftrightarrow \exists w' \in R_w, \mathcal{M}, w' \models \varphi \\ & \Leftrightarrow R_w \neq \emptyset \text{ et } \frac{|R_w(\varphi)|}{|R_w|} \geq \frac{1}{|R_w|} \end{aligned}$$

■

4 Principes de l'extension pondérée des axiomes modaux

Les axiomes de la logique modale classique [1] peuvent être vus comme des règles de manipulation des opérateurs modaux \Box et \Diamond . Ils établissent ainsi des relations entre une répétition de modalités, une modalité simple ou une combinaison de modalités.

Nous proposons d'étudier l'extension pondérée de ces axiomes, pour lesquels les modalités contenues dans les formules considérées sont remplacées par des modalités pondérées. Nous obtenons un ensemble de formules modales correspondant à la variante pondérée de chaque axiome et dans lesquelles les modalités portent chacune leur propre poids. Ainsi, l'axiome classique (4) qui s'écrit $\vdash \Box \varphi \rightarrow \Box \Box \varphi$ conduit à l'extension pondérée $\Box_\alpha \varphi \rightarrow \Box_\beta \Box_\gamma \varphi$.

Nous nous intéressons aux dépendances entre les valeurs de ces poids, selon l'approche sémantique basée sur l'interprétation de la modalité pondérée proposée dans la section précédente. La méthode que nous choisissons consiste en l'identification de dépendance entre les poids, soit de façon générique, soit pour des cadres de Kripke dont la relation vérifie certaines propriétés.

Cette section présente les principes de valuation des poids dans les axiomes pondérés, d'abord selon leur structure puis selon des propriétés de la relation du cadre de Kripke.

4.1 Contraintes liées à la structure des axiomes

La majorité des axiomes des logiques modales (voir [4] par exemple) prennent la forme d'une implication. Lorsqu'ils sont interprétés comme des composants d'un système d'inférence, ils sont d'autant plus

intéressants que leur prémisse est facile à satisfaire et leur conclusion informative.

En relation avec la propriété de gradualité décroissante établie dans la Proposition 2, ce principe permet de guider l’établissement de poids pertinents pour les axiomes pondérés : la position de la modalité par rapport à l’implication influe sur la définition de valeur pertinente et la structure de l’axiome joue sur celle-ci.

Ainsi, lorsque la modalité \Box_α est dans la conclusion de l’implication, la valeur de α doit être maximale. En effet, comme pour toute valeur β inférieure, \Box_β peut en être inférée, la valeur la plus haute est la plus informative.

Au contraire, si \Box_α est dans la prémisse de l’implication, la valeur de α la plus pertinente est la valeur minimale : c’est la plus petite valeur qui définit la prémisse la plus facile à satisfaire. En effet, toujours en exploitant la propriété de gradualité décroissante, si $\Box_\beta\varphi$ est établie pour une valeur de β supérieure à celle de l’axiome, elle en permet l’utilisation par modus ponens avec l’équation (5).

Par dualité, pour l’opérateur \Diamond_α qui satisfait une propriété de gradualité croissante, la définition de valuation pertinente est inversée : la valeur de α doit donc être minimale pour \Diamond_α dans la conclusion, et maximale dans la prémisse.

On peut alors qualifier l’extension pondérée d’enrichissement, relâchement ou altération de l’axiome modal d’origine, selon la position de la modalité pondérée et la valuation du poids : si la pondération de l’axiome est établie pour \Box_α avec un α très élevé en conclusion, l’axiome induit est riche car il permet d’inférer des éléments informatifs. Il faut noter que cette configuration n’a d’intérêt que si la prémisse contient elle-même une modalité pondérée : dans le cas contraire, on peut appliquer l’axiome classique concluant à \Box_1 , qui induit \Box_α par gradualité décroissante.

Lorsque l’axiome pondéré présente une modalité pondérée \Box_α dans sa prémisse, il peut être considéré comme un relâchement de l’axiome classique correspondant. En effet, il autorise une inférence même lorsque la forte hypothèse \Box_1 n’est pas vérifiée.

Enfin, des variations plus complexes peuvent conduire à un axiome pondéré que l’on peut seulement qualifier d’altération de l’axiome d’origine, comme discuté dans la section 5.

4.2 Exploitation des propriétés sur la relation

Un deuxième principe de valuation des poids se base sur la propriété de la relation associée à l’axiome classique dans la théorie de la correspondance [14]. En effet, le point de vue sémantique de l’axiomatisation de la logique modale s’appuie sur des cadres de Kripke dont la relation d’accessibilité satisfait des contraintes

spécifiques, comme la réflexivité ou la transitivité. Or, l’interprétation de la modalité pondérée dépend des mondes accessibles et caractériser la relation permet d’avoir des garanties sur l’ensemble des mondes accessibles. Il est donc pertinent de se baser sur les propriétés de la relation pour valuer les poids.

La première étape de l’interprétation sémantique de l’extension pondérée d’un axiome modal considère des cadres de Kripke satisfaisant la propriété classiquement correspondante. En effet, le cas limite où tous les poids intégrés prennent la valeur 1 impose de préserver le cas classique.

On peut alors évaluer la dépendance entre les valeurs des poids dans l’axiome selon l’hypothèse sur la relation. Ce principe garantit la compatibilité avec le cas limite où tous les poids intégrés valent 1.

Une deuxième étape peut devoir imposer d’autres contraintes sur la relation pour pondérer les axiomes de manière pertinente : il est possible que l’on obtienne un axiome qui n’exclut pas le cas où la modalité de la conclusion est pondérée par $\alpha = 0$. Or, comme discuté en section 3.2, cette valeur n’est pas informative.

La modalité \Box_α , qui propose une interprétation plus expressive que \Box , est aussi moins informative. En effet, savoir qu’une proportion de mondes accessibles valide la formule ne donne pas d’information sur la validité de cette formule dans tous les mondes considérés. La modalité \Box_α donne une indication globale et induit une incertitude quant à la validité de la formule dans un monde en particulier. L’approche que nous proposons alors consiste à enrichir l’hypothèse sur la relation afin d’exclure les cadres n’admettant que la valeur nulle.

Enfin, une fois la valuation pertinente établie avec une hypothèse enrichie sur la relation, nous nous intéressons à la proposition inverse : elle consiste à vérifier si, étant donné un axiome pondéré d’après une hypothèse enrichie sur la relation, tous les cadres validant l’axiome satisfont cette hypothèse. Cette étude peut être considérée comme un premier pas vers la définition d’une théorie de la correspondance pondérée.

5 Typologie des axiomes pondérés

Cette section présente les résultats obtenus, c’est-à-dire les extensions pondérées des axiomes de la logique modale après application des principes de pondération présentés dans la section précédente. Pour l’interprétation sémantique des modalités, nous utilisons la définition proposée en section 3.

Les axiomes pondérés obtenus sont organisés en quatre types récapitulés dans le tableau 1, numérotés de (I) à (IV) et définis dans la section 5.1 ci-dessous. Les sections suivantes détaillent successivement les quatre types d’axiomes.

5.1 Définition des quatre types d'axiomes

Les quatre types d'axiomes identifiés sont définis en fonction du lien entre l'axiome pondéré et les propriétés de la relation :

- (I) Axiomes non pondérés
- (II) Axiomes pondérés avec correspondance classique
- (III) Axiomes pondérés sans correspondance
- (IV) Axiomes pondérés avec correspondance enrichie

Le type (I) regroupe les axiomes pour lesquels le relâchement de la modalité classique ne peut pas exploiter les poids tels que définis dans la sémantique proposée. Le type (II) comporte les axiomes dont la variante pondérée préserve la propriété de la relation classiquement associée à l'axiome. Les types (III) et (IV) regroupent les axiomes pondérés pour lesquels la propriété classique doit être enrichie, et distinguent les cas où la correspondance est, ou non, conservée.

Il faut souligner que, d'un axiome classique, peuvent découler plusieurs extensions pondérées, selon l'hypothèse sur la relation considérée.

Les quatre sections suivantes présentent les quatre types d'axiomes. Pour des raisons de place, un seul exemple est détaillé par type.

5.2 Type (I) : axiomes non pondérés

Le premier groupe d'axiomes regroupe les axiomes pour lesquels la seule possibilité de pondération est le cas limite où $\alpha = 1$: le relâchement de l'axiome classique ne peut s'appliquer.

Cette catégorie peut être illustrée par l'axiome (M), dont la forme pondérée s'écrit $\Box_\alpha \varphi \rightarrow \varphi$. Par compatibilité avec le cas classique, cette formule doit être valide pour $\alpha = 1$ dans tous les cadres dont la relation est réflexive. Cependant, si $\Box_\alpha \varphi$ est établi pour $\alpha < 1$ dans un monde donné, la contrainte de réflexivité ne garantit pas que le monde de référence ne soit pas parmi les mondes accessibles invalidant la formule, on ne peut donc pas établir φ . Formellement, on peut montrer le théorème suivant par construction.

Théorème 1. $\forall \alpha \in [0, 1[$, il existe un modèle $\mathcal{M} = \langle \langle W, R \rangle, s \rangle$ avec R réflexive et $w \in W$ tel que $\mathcal{M}, w \models \Box_\alpha \varphi$ mais $\mathcal{M}, w \not\models \varphi$.

Démonstration. Soit $\alpha \in [0, 1[$. Il suffit d'un contre-exemple pour montrer ce théorème. Soit $F = \langle W, R \rangle$ un cadre contenant n mondes, où n est tel que $n/(n+1) \geq \alpha$ et, étant donné un monde $w \in W$, R est réflexive et $R_w = W$. Soit s la valuation telle que (i) $x \models \varphi$ pour tout $x \in W \setminus \{w\}$ et (ii) $w \not\models \varphi$. On a bien $w \models \Box_\alpha \varphi$ mais $w \not\models \varphi$. ■

	Axiome pondéré	Propriétés de R	Type
(K $_\alpha$)	$\Box_\alpha(\varphi \rightarrow \psi) \rightarrow (\Box_\beta \varphi \rightarrow \Box_\gamma \psi)$		(II)
(CD $_\alpha$)	$\Diamond_\alpha \varphi \rightarrow \Box_{1-\alpha} \varphi$		(IV)
(D $_\alpha$)	$\Box_\alpha \varphi \rightarrow \Diamond_{1-\alpha+\varepsilon} \varphi$	sérielle	(II)
(M $_\alpha$)	$\Box_1 \varphi \rightarrow \varphi$	réflexive	(I)
(B $_\alpha$)	$\varphi \rightarrow \Box_1 \Diamond_1 \varphi$	symétrique	(I)
(□M $_\alpha$)	$\Box_1(\Box_1 \varphi \rightarrow \varphi)$	shift-réflexive	(I)
(4 $_\alpha$)	$\Box_\alpha \varphi \rightarrow \Box_1 \Box_\alpha \varphi$	transitive & euclidienne	(III)
(5 $_\alpha$)	$\Diamond_\alpha \varphi \rightarrow \Box_1 \Diamond_{\alpha+\varepsilon} \varphi$	transitive & euclidienne	(III)
(C4 $_\alpha$)	$\Box_\alpha \Box_1 \varphi \rightarrow \Box_\alpha \varphi$	dense & shift-réflexive	(IV)
(C4 $_\alpha$)	$\Box_1 \Box_\beta \varphi \rightarrow \Box_\beta \varphi$	dense & transitive	(IV)
(C4 $_\alpha$)	$\Box_1 \Box_\beta \varphi \rightarrow \Diamond_1 \varphi$	dense & transitive	(IV)
(C4 $_\alpha$)	$\Box_\alpha \Box_\beta \varphi \rightarrow \Box_\beta \varphi$	transitive & euclidienne	(IV)

TABLE 1 – Axiomes pondérés obtenus avec l'hypothèse sur la relation associée, pour tout $\alpha, \beta \in [0, 1]$, $\varepsilon \in]0, \alpha]$ et avec $\gamma = \max(0, \alpha + \beta - 1)$

Un traitement similaire peut être appliqué aux axiomes (B) et (□M), dont les extensions pondérées ne s'appliquent que pour $\alpha = 1$ (cf. tableau 1).

5.3 Type (II) : axiomes pondérés avec correspondance classique

Le deuxième type d'axiomes pondérés relâche l'axiome classique afin d'en donner une version pondérée pour laquelle la valeur des poids peut être inférieure à 1. De plus, la propriété de la relation classiquement associée à l'axiome est conservée et suffit à valuer les poids de manière pertinente. L'axiome classique (K), qui n'impose aucune propriété, illustre cette catégorie :

Théorème 2. (K $_\alpha$)

$\forall \alpha, \beta \in [0, 1]$

$$\models \Box_\alpha(\varphi \rightarrow \psi) \rightarrow (\Box_\beta \varphi \rightarrow \Box_\gamma \psi)$$

avec $\gamma = \max(0, \alpha + \beta - 1)$

Démonstration. Soit $\alpha, \beta \in [0, 1]$, un cadre $F = \langle W, R \rangle$, une valuation s et un monde $w \in W$. Si $R_w = \emptyset$, alors w valide trivialement les trois formules modales de (K $_\alpha$) et leurs implications. Si $|R_w| > 0$, on peut appliquer le modus ponens dans les mondes accessibles qui valident à la fois $\varphi \rightarrow \psi$ et φ . Aussi on a $R_w(\varphi \rightarrow \psi) \cap R_w(\varphi) \subseteq R_w(\psi)$.

Ainsi, par définition du cardinal de l'intersection : $|R_w(\varphi \rightarrow \psi) \cap R_w(\varphi)| = |R_w(\varphi \rightarrow \psi)| + |R_w(\varphi)| -$

$|R_w(\varphi \rightarrow \psi) \cup R_w(\varphi)|$. Comme $|R_w| \geq |R_w(\varphi \rightarrow \psi) \cup R_w(\varphi)|$, on a : $|R_w(\psi)| \geq |R_w(\varphi \rightarrow \psi) \cap R_w(\varphi)| \geq |R_w(\varphi \rightarrow \psi)| + |R_w(\varphi)| - |R_w|$. Donc : $\frac{|R_w(\psi)|}{|R_w|} \geq \alpha + \beta - 1$. ■

Une extension pondérée de l'axiome (D) peut être construite de manière similaire lorsque l'on considère des cadres de Kripke à relation sérielle. De plus, on montre réciproquement que, dans tout cadre à relation sérielle, (D_α) est vérifié. Cette variante pondérée (D_α) , dont la définition est donnée dans le tableau 1, complète la Proposition 3 qui relie les deux opérateurs modaux \Box_α et \Diamond_α .

5.4 Type (III) : axiomes pondérés sans correspondance

Cette section présente les axiomes pour lesquels la propriété de la relation classiquement correspondante doit être enrichie pour définir des poids pertinents.

Elle est illustrée par l'axiome (4) qui s'écrit $\vdash \Box\varphi \rightarrow \Box\Box\varphi$ et est associée à la transitivité de la relation. La variante pondérée de cet axiome s'écrit sous la forme $\vdash \Box_\alpha\varphi \rightarrow \Box_\beta\Box_\gamma\varphi$: la question ici étant de déterminer des valeurs appropriées pour β et γ selon la valeur de α . On montre que la seule condition de transitivité de la relation ne suffit pas pour établir des valeurs de poids strictement positives :

Théorème 3. $\forall \alpha \in [0, 1[$, il existe un modèle $\mathcal{M} = \langle \langle W, R \rangle, s \rangle$ avec R transitive et un monde $w \in W$ tel que $\mathcal{M}, w \models \Box_\alpha\varphi$ et $\mathcal{M}, w \not\models \Box_1\Box_0\varphi$.

Il faut noter que, d'après la Proposition 1, $\Box_1\Box_0\varphi = \Box_1\Box_1\neg\varphi$, ce qui ne constitue pas une conclusion informative sur φ .

Démonstration. La preuve de ce théorème consiste à construire un tel modèle \mathcal{M} . Étant donné $\alpha < 1$, soient $m, q \in \mathbb{N}^*$ tels que $\alpha \leq m/(m+q)$. Soit W un ensemble de $1+q+m$ mondes, $w \in W$, R une relation binaire entre les mondes et s une valuation telle que (i) $|R_w(\varphi)| = m$, (ii) $|R_w(\neg\varphi)| = q$, (iii) $\forall w_p \in R_w(\varphi)$, $\exists w_n \in R_w(\neg\varphi)$ tel que $w_p R w_n$ et (iv) $\forall w_n \in R_w(\neg\varphi)$, $w_n R w_n$. Un tel modèle est illustré par la figure 1 pour $\alpha = \frac{2}{3}$, avec $m = 2$ et $q = 1$. Par définition, R est transitive et on a $\mathcal{M}, w \models \Box_{m/(m+q)}\varphi$. Par conséquent, en utilisant la propriété de gradualité décroissante : $\mathcal{M}, w \models \Box_\alpha\varphi$.

De plus, comme $\forall u \in R_w$, $\mathcal{M}, u \models \Box_1\neg\varphi$, on a $\mathcal{M}, w \models \Box_1\Box_1\neg\varphi$: aussi, il n'existe pas de β tel que $\mathcal{M}, w \models \Box_\beta\Box_\gamma\varphi$ pour $\gamma > 0$. ■

Aussi, la transitivité n'est pas une condition suffisante pour avoir des garanties sur les valeurs de β

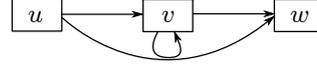


FIGURE 2 – Contre-exemple montrant la non-réciprocité du Théorème 4

et γ : elle n'empêche pas l'existence des mondes notés w_n dans la preuve ci-dessus et illustrés par w^* sur la figure 2, que l'on peut appeler mondes « puits ». Il est donc nécessaire d'enrichir l'hypothèse sur la relation en ajoutant une autre contrainte. Le Théorème 4 considère le cas de l'euclidianité, en conservant la transitivité par souci de compatibilité avec le cas classique, obtenu lorsque les poids sont égaux à 1.

Théorème 4. (4_α)

$\forall F = \langle W, R \rangle$,

R est transitive et euclidienne

$\Rightarrow \forall \alpha \in [0, 1], F \models \Box_\alpha\varphi \rightarrow \Box_1\Box_\alpha\varphi$

Démonstration. Cette preuve repose sur le fait que, pour tout cadre $\langle W, R \rangle$ dont la relation R est transitive et euclidienne, on a pour tout $w \in W : \forall w' \in R_w$, $R_{w'} = R_w$ (preuve omise). Ainsi, pour toute valuation s et pour tout $\alpha \in [0, 1]$, si $w \models \Box_\alpha\varphi$, alors tous les mondes accessibles $w' \in R_w$ valident aussi $w' \models \Box_\alpha\varphi$, c'est-à-dire $w \models \Box_1\Box_\alpha\varphi$. ■

Toutefois, la réciproque n'est pas vérifiée : la figure 2 représente un cadre $F = \langle W, R \rangle$ avec $W = \{u, v, w\}$, tel que $\langle F, s \rangle, x \models \Box_\alpha\varphi \rightarrow \Box_1\Box_\alpha\varphi$ pour toute valuation s et pour tout monde $x \in W$ mais R n'est pas euclidienne.

Il faut souligner que l'axiome établi dans le Théorème 4 est fort : la première modalité en conclusion vaut 1, le degré maximal, et la seconde vaut α , qui est exactement la valeur affectée à la modalité en prémisses. De ce fait, on ne peut améliorer l'extension pondérée de l'axiome (4) car aucune autre version ne peut être plus forte.

Des résultats identiques, listés dans le Tableau 1 mais non détaillés ici, peuvent être prouvés pour l'extension classique de l'axiome (5) ; il possède la même structure que l'axiome (4), avec un seul opérateur, \Diamond_α , dans la prémisses et une combinaison des deux modalités dans la conclusion.

5.5 Type (IV) : axiomes pondérés avec correspondance enrichie

Les axiomes pondérés appartenant au type (IV) constituent également des extensions des axiomes classiques pour lesquels la propriété de la relation correspondante doit être enrichie. La différence avec le type

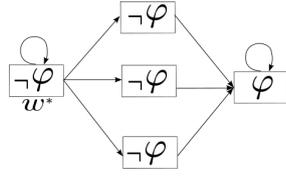


FIGURE 3 – Contre-exemple prouvant le Théorème 5 pour $\alpha = \frac{3}{4}$

(III) vient du fait que la correspondance entre l'axiome pondéré et l'hypothèse enrichie est établie.

Nous illustrons cette catégorie avec le cas de l'axiome (C4) : son écriture classique est $\vdash \Box\Box\varphi \rightarrow \Box\varphi$, et il est associé à la densité de la relation. La version pondérée de cet axiome prend la forme $\vdash \Box_\alpha\Box_\beta\varphi \rightarrow \Box_\gamma\varphi$. Comme l'indique le Théorème 5, la densité seule ne suffit pas à garantir des valuations pertinentes pour les poids. Pour toutes les valeurs de α et β , on peut construire un modèle pour lequel $\gamma = 0$:

Théorème 5. $\forall \alpha \in [0, 1[, \forall \beta \in [0, 1]$, il existe un modèle \mathcal{M} avec R dense et $w \in W$ tel que $\mathcal{M}, w \models \Box_\alpha\Box_\beta\varphi$ et $\mathcal{M}, w \models \Box_1\neg\varphi$

Démonstration. À nouveau, la preuve du théorème se fait par construction d'un tel modèle \mathcal{M} . Étant donné $\alpha \in [0, 1[$ et $\beta \in [0, 1]$, soit $n \in \mathbb{N}$ tel que $n \geq \alpha/(1-\alpha)$. Soit W un ensemble de $n+2$ mondes, w^* et w' deux mondes distincts dans W , R et s tel que (i) $\mathcal{M}, w' \models \varphi$, (ii) $\forall w \in W \setminus \{w'\}$, $\mathcal{M}, w \models \neg\varphi$, (iii) $R_{w^*} = W \setminus \{w'\}$ et (iv) $\forall w \in W \setminus \{w^*\}$, $R_w = \{w'\}$. Par construction, R est dense. Un tel contre-exemple est représenté par la Figure 3 pour $\alpha = 0,75$.

Alors $\forall w \in W \setminus \{w^*\}$, $\mathcal{M}, w \models \Box_1\varphi$. Ceci implique par gradualité décroissante que $\mathcal{M}, w \models \Box_\beta\varphi$. Par conséquent

$$\begin{aligned} |R_{w^*}(\Box_\beta\varphi)| = n &\Rightarrow \frac{|R_{w^*}(\Box_\beta\varphi)|}{|R_{w^*}|} = \frac{n}{n+1} \geq \alpha \\ &\Rightarrow \mathcal{M}, w^* \models \Box_\alpha\Box_\beta\varphi \end{aligned}$$

Toutefois $\mathcal{M}, w^* \models \Box_1\neg\varphi$ donc $\nexists \gamma > 0$ tel que $\mathcal{M}, w^* \models \Box_\gamma\varphi$. ■

En conséquence de ce théorème, il est nécessaire d'enrichir l'hypothèse sur la relation afin d'avoir des garanties de positivité sur la valeur de γ . L'ensemble des choix de propriétés ajoutées est résumé dans le Tableau 1. Les quatre extensions pondérées de l'axiome (C4) sont classées par durcissement croissant de l'hypothèse sur la relation. Cette hypothèse influe sur l'informativité de la conclusion et le niveau de contrainte de la prémisse.

Nous donnons la preuve pour la version pondérée la plus forte :

Théorème 6. (C4 $_\alpha$)

$\forall F = \langle W, R \rangle$,

R est transitive et euclidienne

$$\Rightarrow \forall \alpha, \beta \in]0, 1], \beta \in [0, 1], F \models \Box_\alpha\Box_\beta\varphi \rightarrow \Box_\beta\varphi$$

Démonstration. Soit $\langle W, R \rangle$ avec R transitive et euclidienne, soit un monde $w \in W$ et $\alpha, \beta \in]0, 1]$ (si $\beta = 0$ alors $\Box_\beta\varphi$ est vraie, donc l'implication l'est aussi).

Si $w \not\models \Box_\alpha\Box_\beta\varphi$, alors $w \models \Box_\alpha\Box_\beta\varphi \rightarrow \Box_\beta\varphi$.

Si $w \models \Box_\alpha\Box_\beta\varphi$, alors une proportion α de mondes accessibles depuis w valide $\Box_\beta\varphi$, soit u un tel monde, on peut écrire :

$$\frac{|R_u(\varphi)|}{|R_u|} \geq \beta$$

Or, comme R est transitive et euclidienne, on sait que $\forall w' \in R_w$, $R_{w'} = R_w$. En particulier, $R_u = R_w$. On a donc :

$$\frac{|R_w(\varphi)|}{|R_w|} \geq \beta$$

Donc $w \models \Box_\beta\varphi$. ■

La réciproque est prouvée par contraposée : si la relation n'est pas transitive et euclidienne, alors l'axiome (C4 $_\alpha$) n'est pas valide. Pour cela, nous construisons un cadre de Kripke dont la relation est non transitive ou non euclidienne et nous proposons des valeurs pour α et β ainsi qu'une valuation telles que l'axiome n'est pas vérifié dans un monde du cadre.

Le principe de cette preuve est illustré par sa première partie : nous montrons que pour tout cadre $\langle W, R \rangle$, si la relation R n'est pas transitive, alors il existe α, β et une valuation telle qu'il existe un monde $w \in W$ tel que $w \not\models (C4_\alpha)$. Formellement,

Théorème 7.

$\forall F = \langle W, R \rangle$,

R est non transitive

$$\Rightarrow \exists \alpha, \beta \in [0, 1], \exists s, \exists w \in W,$$

$$\langle \langle W, R \rangle, s \rangle, w \not\models \Box_\alpha\Box_\beta\varphi \rightarrow \Box_\beta\varphi$$

Démonstration. Soit W un ensemble fini de mondes et R une relation non transitive : il existe $u, v, w \in W$ tels que $uRv \wedge vRw \wedge \neg uRw$. Posons $\alpha = \frac{1}{|R_u|}$, $\beta = \frac{1}{|R_v|}$ et soit s une valuation telle que (i) $w \models \varphi$ et (ii) $\forall x \in W \setminus \{w\}$, $x \not\models \varphi$. La figure 4 présente un exemple d'un tel modèle. On a :

- $u \models \Box_1\neg\varphi$ car le seul monde qui valide φ n'est pas accessible à partir de u : $w \notin R_u$. Aussi, en particulier $u \not\models \Box_\beta\varphi$
- $v \models \Box_\beta\varphi$ car $w \in R_v$ et $w \models \varphi$

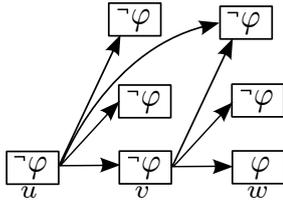


FIGURE 4 – Modèle de Kripke tel que R est non transitive

— $u \models \Box_\alpha \Box_\beta \varphi$ car $v \in R_u$ et $v \models \Box_\beta \varphi$
 Donc $\exists \alpha, \beta$ tels que $u \models \Box_\alpha \Box_\beta \varphi$ mais $u \not\models \Box_\beta \varphi$, ce qui implique que $u \not\models (C4_\alpha)$. ■

De la même manière, on peut montrer que si la relation n'est pas euclidienne alors il existe un modèle qui ne vérifie pas l'axiome $(C4_\alpha)$.

6 Conclusion et travaux futurs

Cet article étudie des règles de combinaison entre les opérateurs modaux pondérés, proposant une extension des axiomes classiques de la logique modale. Il établit une typologie des axiomes pondérés, mis en perspective avec la propriété de la relation classiquement associée à chacun des axiomes.

Les perspectives incluent l'étude d'autres types de règles de manipulation, combinant par exemple la modalité et d'autres connecteurs logiques, comme la conjonction et la disjonction. D'autres types de propriétés de la relation peuvent également être considérés : nous avons centré nos propositions sur les plus répandues, il serait intéressant d'en étudier des versions relâchées, comme par exemple une α -symétrie, afin d'examiner leurs effets sur la valuation des poids d'un axiome pondéré.

La mise en correspondance entre propriétés de la relation et axiomes soulève de façon plus générale les problèmes d'adéquation et de complétude, en regard de l'interprétation proposée des modalités pondérées dans la sémantique de Kripke. Les axiomes de type (III) requièrent à ce titre une étude approfondie.

Les perspectives incluent également une spécification de la logique modale pondérée que nous proposons dans un contexte purement doxastique : d'un point de vue axiomatique, la variante pondérée de l'axiomatique KD45 et ses propriétés peut être considérée à partir des axiomes établis dans cet article. Une telle variante doit fournir des outils de manipulation et de raisonnement à partir de croyances graduelles, pouvant par exemple être mis en œuvre dans des applications basées sur des agents intelligents.

Références

- [1] P. Blackburn, M. De Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2001.
- [2] F. Bou, F. Esteva, L. Godo, and R. Rodriguez. Characterizing fuzzy modal semantics by fuzzy multimodal systems with crisp accessibility relations. In *Proc. of the Joint 2009 IFSA World Congress and EUSFLAT Conference*, pages 1541–1546, 2009.
- [3] C. Boutilier. Modal logics for qualitative possibility theory. *International Journal of Approximate Reasoning*, 10(2) :173–201, 1994.
- [4] B. F. Chellas. *Modal logic : an introduction*, volume 316. Cambridge Univ Press, 1980.
- [5] F. De Caro. Graded modalities II. *Studia Logica*, 47(2) :1–10, 1988.
- [6] D. Dubois, H. Prade, and S. Schockaert. Stable models in generalized possibilistic logic. In *13th Int. Conf. on the Principles of Knowledge Representation and Reasoning*, 2012.
- [7] L. Fariñas del Cerro and A. Herzig. A modal analysis of possibility theory. In *Fundamentals of Artificial Intelligence Research*, pages 11–18. Springer, 1991.
- [8] M. Fattorosi-Barnaba and C. Cerrato. Graded modalities III. *Studia Logica*, 47 :99–110, 1988.
- [9] M. Fattorosi-Barnaba and F. De Caro. Graded modalities I. *Studia Logica*, 44(2) :197–221, 1985.
- [10] K. Fine. In so many possible worlds. *Notre Dame Journal of Formal Logic*, 13(4) :516–520, 1972.
- [11] N. Laverny. Logique doxastique graduelle. Technical report, Journées Francophones sur les Modèles Formels de l'Interaction (MFI), 2007.
- [12] E. Pacuit and S. Salame. Majority logic : Axiomatization and completeness. *Studia Logica*, 2006.
- [13] A. Shirazi and E. Amir. Probabilistic modal logic. In *Proc. of AAAI*, volume 7, pages 489–495, 2007.
- [14] J. Van Benthem. *Correspondence theory*. Springer, 1984.
- [15] W. van der Hoek and J.-J. Meyer. *Graded modalities in epistemic logic*. Springer, 1992.
- [16] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8 :338–353, 1965.
- [17] L.A. Zadeh. Fuzzy sets as the basis for a theory of possibility. *Fuzzy Sets and Systems*, 1 :3–28, 1978.

Révision des croyances dans une clôture propositionnelle de contraintes linéaires

Jean Lieber

LORIA UMR 7503 (Université de Lorraine, CNRS) et Inria Nancy Grand Est
54506 Vandœuvre-lès-Nancy, France
Jean.Lieber@loria.fr

Résumé

Réviser des croyances par d'autres croyances consiste à modifier les premières pour qu'elles soient cohérentes avec les secondes. Certains opérateurs de révision de croyances s'appuient sur des distances entre interprétations (qui servent à « mesurer » les modifications). La révision des croyances a été étudiée dans plusieurs formalismes, notamment en logique propositionnelle et dans le formalisme des conjonctions de contraintes linéaires. Cet article étudie la révision dans la clôture propositionnelle des contraintes linéaires, qui étend les deux formalismes précédents. La difficulté principale tient au fait qu'un opérateur s'appuyant sur une distance classique sur les n -uplets de réels, telle que la distance de Manhattan, ne vérifiera pas les postulats classiques de la révision. La solution proposée ici consiste à utiliser une distance à ensemble de valeurs discret. Un opérateur de révision est ainsi décrit et étudié, et un algorithme pour cet opérateur est présenté, qui s'appuie sur une mise sous forme normale disjonctive et sur des optimisations linéaires.

Abstract

Revising beliefs by other beliefs consists in modifying the formers so that they become consistent with the latters. Some revision operators are based on distance functions between interpretations (that enable to “measure” the modifications). Belief revision has been studied in several formalisms, including propositional logic and the formalism of conjunctions of linear constraints. This paper studies revision in the propositional closure of linear constraints, which extends the two previous formalisms. The main difficulty is related to the fact that an operator using a classical distance function on tuples of real numbers, such as the Manhattan distance, does not satisfy the classical postulates of revision. The solution proposed here consists in using a distance function with a discrete set of va-

lues. Such a revision operator is described and studied, and an algorithm for it is presented, based on a transformation into disjunctive normal form and on linear optimizations.

1 Introduction

Le raisonnement à partir de cas (RÀPC) est un mode de raisonnement s'appuyant sur la réutilisation d'un ensemble de cas, un cas étant la représentation d'un épisode de résolution de problème [19]. Dans beaucoup de travaux, raisonner à partir de cas consiste dans un premier temps à sélectionner un cas dans une base de cas qui soit proche du cas représentant le problème cible (étape de remémoration), puis à modifier ce cas afin de résoudre le problème cible (étape d'adaptation).

Une approche développée ces dernières années a consisté à utiliser un opérateur de révision des croyances pour effectuer l'adaptation [7]. Le RÀPC peut ainsi bénéficier des travaux visant à implanter des opérateurs de révision dans divers formalismes, par exemple dans des logiques de descriptions [2, 4, 6], dans la logique propositionnelle \mathcal{L}_P ou des fragments de \mathcal{L}_P [10, 20, 18], ou dans des algèbres qualitatives [14, 13]. Certains de ces travaux ont été motivés par le développement d'un système de RÀPC, TAAABLE, qui résout des problèmes culinaires [9], les opérateurs de révision étant réunis dans la bibliothèque RÉVISOR (<http://revisor.loria.fr>) : RÉVISOR/PL et RÉVISOR/PLAK en \mathcal{L}_P [18], RÉVISOR/CLC dans le formalisme \mathcal{L}_{CCL} des conjonctions de contraintes linéaires [5] et dans \mathcal{L}_{AQ} , une algèbre qualitative [13] (en pratique, les algèbres qualitatives étudiées étaient Allen, INDU et RCC8). Dans TAAABLE, ces opérateurs ont été utilisés de la façon suivante : RÉVISOR/PL et RÉVISOR/PLAK permettent de proposer des substitutions de noms d'ingrédients, RÉVISOR/CLC permet d'adapter

les quantités d'ingrédients (par exemple, le remplacement d'un fruit par un fruit moins sucré peut être accompagné par une augmentation de la quantité de sucre en poudre), RÉVISOR/QA permet d'adapter les préparations.

Le système TAAABLE fait donc appel à différents opérateurs de révision dans différents formalismes. Cependant, il n'y a pas interaction entre ces trois inférences, en dehors de leurs exécutions par le système. Pour une interaction qui soit gérée au sein d'un mécanisme d'inférences, il faudrait un formalisme combinant ces formalismes. Une révision dans une première « combinaison » de formalisme a été étudiée : le cadre formel étant celui de la clôture propositionnelle d'une algèbre qualitative [12]. Cet article présente une étude de la révision dans une nouvelle combinaison de formalismes : \mathcal{L}_{CPCL} , la clôture propositionnelle de \mathcal{L}_{CCL} .

Après les préliminaires (section 2), la logique \mathcal{L}_{CPCL} est définie en section 3. Puis, une définition d'un opérateur de révision dans \mathcal{L}_{CPCL} est donnée (section 4). Cet opérateur s'appuiera sur une distance entre n -uplets de réels. On verra que l'utilisation des distances usuelles (telles que la distance de Manhattan ou la distance euclidienne) pose problème, pour la satisfaction des postulats AGM des opérateurs de révision. Une autre famille de distance sera utilisée pour contourner ce problème. Puis, la section 5 présentera un algorithme pour calculer la révision de ψ par μ où ψ et μ sont des ensembles de croyances représentés dans \mathcal{L}_{CPCL} . \mathcal{L}_{CPCL} s'appuie sur des variables à valeurs réelles. La section 6 étudie comment l'approche présentée s'étend pour les contraintes avec des variables qui peuvent être à valeurs entières ou réelles. La section 7 conclura et mettra en évidence des perspectives de recherches.

Le rapport [16] est une version longue du présent article : la différence est principalement au niveau des preuves, plus détaillées dans ce rapport.

2 Préliminaires

Ensembles de nombres. \mathbb{N} , \mathbb{Z} , \mathbb{Q} et \mathbb{R} sont, respectivement, les ensembles des entiers naturels, des entiers relatifs, des rationnels et des réels. On considérera (par plongements) qu'on a la chaîne des inclusions $\mathbb{N} \subseteq \mathbb{Z} \subseteq \mathbb{Q} \subseteq \mathbb{R}$. \mathbb{Q}_+ est l'ensemble des rationnels $x \geq 0$. \mathbb{R}_+ est l'ensemble des réels $x \geq 0$. Si $\varepsilon \in \mathbb{R}$, on note $\varepsilon\mathbb{N}$ l'ensemble $\{\varepsilon n \mid n \in \mathbb{N}\} = \{0, \varepsilon, 2\varepsilon, \dots\}$.

Parties entières d'un réel. Étant donné $x \in \mathbb{R}$, on note $\lfloor x \rfloor$ et $\lceil x \rceil$ sa partie entière et sa partie entière supérieure : $\lfloor x \rfloor = \max\{n \in \mathbb{Z} \mid x \geq n\}$ et $\lceil x \rceil = \min\{n \in \mathbb{Z} \mid x \leq n\}$. On notera les propriétés suivantes (pour

$\alpha, \beta \in \mathbb{R}$) :

$$\lceil \alpha + \beta \rceil \leq \lceil \alpha \rceil + \lceil \beta \rceil \quad (1)$$

$$\lceil \alpha \rceil \leq \beta \quad \text{ssi} \quad \alpha \leq \lfloor \beta \rfloor \quad (2)$$

$$\alpha \leq \lceil \alpha \rceil < \alpha + 1 \quad (3)$$

Distances et espaces métriques. Soit E un ensemble. Une distance sur E est une fonction d qui à $(x, y) \in E^2$ associe un réel positif $d(x, y)$ et est tel que, pour tout $x, y, z \in E$, (i) $d(x, y) = 0$ ssi $x = y$, (ii) $d(x, y) = d(y, x)$ (symétrie) et (iii) $d(x, z) \leq d(x, y) + d(y, z)$ (inégalité triangulaire).

Un espace métrique est un ensemble E muni d'une distance d et est noté (E, d) .

Si A et B sont des sous-ensembles de E et que y est un élément de E , on note $d(A, y) = \inf_{x \in A} d(x, y)$ et $d(A, B) = \inf_{x \in A, y \in B} d(x, y)$. On a donc l'implication suivante, pour A et A' , deux sous-ensembles de E :

$$\text{Si } A \subseteq A' \text{ alors } d(A, y) \geq d(A', y) \quad (4)$$

Par convention, on pose $\inf \emptyset = +\infty$. D'où $d(\emptyset, y) = +\infty$.

Soit (E, d) un espace métrique, $\lambda \in \mathbb{R}_+$ et $A \subseteq E$. On appelle *dilatation* de A à distance λ pour la distance d l'ensemble

$$G_\lambda^d(A) = \{y \in E \mid d(A, y) \leq \lambda\}$$

Le terme dilatation est emprunté à la morphologie mathématique [3]¹. En particulier, l'*adhérence* d'un ensemble A dans (E, d) est $G_0^d(A)$. On dira que A est *fermé* dans (E, d) s'il est égal à son adhérence ($G_0^d(A) = A$). Par ailleurs, on notera que si A, B et C sont des sous-ensembles de E , $y \in E$ et $\lambda \in \mathbb{R}_+$, on a :

$$G_\lambda^d(A \cup B) = G_\lambda^d(A) \cup G_\lambda^d(B) \quad (5)$$

$$G_\lambda^d(A \cap B) \subseteq G_\lambda^d(A) \cap G_\lambda^d(B) \quad (6)$$

$$\text{si } A \text{ et } B \text{ sont fermés alors } A \cap B \text{ est fermé} \quad (7)$$

$$d(G_0^d(C), y) = d(C, y) \quad (8)$$

$$G_\lambda^d(C) = G_\lambda^d(G_0^d(C)) \quad (9)$$

Soit w_j ($j \in \{1, 2, \dots, n\}$) des constantes réelles strictement positives. Un exemple de distance sur \mathbb{R}^n (où $n \in \mathbb{N}$) est défini, pour $x, y \in \mathbb{R}^n$ par :

$$d(x, y) = \sum_{j=1}^n w_j |y_j - x_j| \quad (10)$$

Cette distance est appelée *distance de Manhattan*.

1. En fait, ce qui est appelé ici dilatation de A à distance λ est la dilatation de A par une boule de rayon λ dans (E, d) .

La révision des croyances est une opération de changement de croyances. Intuitivement, étant donné un ensemble de croyances ψ d'un agent à propos d'un monde statique, elle consiste à considérer les changements de ses croyances en présence d'un nouvel ensemble de croyances μ , en supposant μ prioritaire sur ψ pour l'agent. L'ensemble résultant des croyances de l'agent est noté $\psi * \mu$ et dépend du choix de l'opérateur de révision $*$. Dans [1], le principe du changement minimal a été affirmé et peut être formulé ainsi : ψ est changé minimalement en ψ' tel que la conjonction de ψ' et μ soit cohérent, et le résultat de la révision est cette conjonction. Il y a donc en général plusieurs opérateurs $*$ possibles, étant donné que la définition de $*$ dépend de la manière dont le changement de croyances est « mesuré ». Plus précisément, le principe du changement minimal a été formalisé par un ensemble de postulats, connus sous le nom de postulats AGM (d'après les noms des auteurs de [1]). [17] présente un état de l'art détaillé de la révision à un niveau général (pour tout formalisme satisfaisant certaines propriétés, comme le fait d'être clos pour la conjonction), incluant des théorèmes de représentation et des discussions sur certaines problématiques proches (autres opérateurs de changements de croyances, etc.).

Dans [15], la révision a été étudiée dans le cadre de la logique propositionnelle \mathcal{L}_P (avec un nombre fini de variables). Les postulats AGM ont été traduits dans ce formalisme comme suit ($\psi, \psi_1, \psi_2, \mu, \mu_1, \mu_2$ et φ sont des formules propositionnelles) :

- (*1) $\psi * \mu \models \mu$.
- (*2) Si $\psi \wedge \mu$ est cohérent alors $\psi * \mu \equiv \psi \wedge \mu$.
- (*3) Si μ est cohérent alors $\psi * \mu$ est cohérent.
- (*4) Si $\psi_1 \equiv \psi_2$ et $\mu_1 \equiv \mu_2$ alors $\psi_1 * \mu_1 \equiv \psi_2 * \mu_2$.
- (*5) $(\psi * \mu) \wedge \varphi \models \psi * (\mu \wedge \varphi)$.
- (*6) Si $(\psi * \mu) \wedge \varphi$ est cohérent alors $\psi * (\mu \wedge \varphi) \models (\psi * \mu) \wedge \varphi$.

Dans la suite, $\mathcal{M}(\varphi)$ est l'ensemble des modèles de la formule φ . Il permet de définir la relation $\models : \varphi_1 \models \varphi_2$ si $\mathcal{M}(\varphi_1) \subseteq \mathcal{M}(\varphi_2)$.

Toujours dans [15], une famille d'opérateurs de révision est définie, fondées sur des distances d sur Ω , où Ω est l'ensemble des interprétations. Pour simplifier les notations, pour deux formules ψ et μ et une interprétation y , $d(\psi, y) = d(\mathcal{M}(\psi), y)$ et $d(\psi, \mu) = d(\mathcal{M}(\psi), \mathcal{M}(\mu))$. La révision de ψ par μ selon $*^d$ ($\psi *^d \mu$) est telle que

$$\mathcal{M}(\psi *^d \mu) = \{y \in \mathcal{M}(\mu) \mid d(\psi, y) = d^*\} \quad (11)$$

avec $d^* = d(\psi, \mu)$

Intuitivement, d^* est une mesure de la modification minimale de ψ en ψ' nécessaire pour rendre $\psi' \wedge \mu$ cohérent.

Dans [15], la preuve que $*^d$ vérifie les postulats (*1–6) est donnée².

Une définition équivalente de $*^d$ peut être donnée en utilisant la notion de dilatation :

$$\mathcal{M}(\psi *^d \mu) = G_{d^*}^d(\mathcal{M}(\psi)) \cap \mathcal{M}(\mu) \quad (12)$$

Dans le cas particulier où ψ est incohérent, $\psi *^d \mu \equiv \mu$, puisque $d^* = +\infty$ et $G_{+\infty}^d = \Omega$. Dans le cas particulier où μ est incohérent, $\mathcal{M}(\psi *^d \mu) = \emptyset = \mathcal{M}(\mu)$ et on a également $d^* = +\infty$. En résumé :

- si l'une au moins des formules ψ et μ est incohérente
- alors $\psi *^d \mu \equiv \mu$ et $d(\psi, \mu) = +\infty$ (13)

Cette approche peut être étendue à d'autres formalismes ayant une sémantique en théorie des modèles et telle qu'une distance sur l'ensemble Ω des interprétations peut être définie. Cependant, pour certains formalismes, cela suppose des contraintes additionnelles :

- (a) L'opérateur $*^d$ ainsi défini doit satisfaire les postulats (*1–6), ce qui n'est pas le cas pour toute logique et toute distance sur Ω ;
- (b) L'ensemble d'interprétations $\mathcal{M}(\psi *^d \mu)$ doit être représentable dans le formalisme, i.e., il existe une formule ϱ telle que $\mathcal{M}(\varrho)$ soit cet ensemble (contrainte de représentabilité).

On verra que ces questions se posent pour $\mathcal{L}_{\text{CPCL}}$.

3 La logique ($\mathcal{L}_{\text{CPCL}}, \models$)

3.1 Syntaxe

Soit n un entier naturel non nul fixé. Soit \mathcal{V} , un ensemble de n symboles, appelées *variables* : $\mathcal{V} = \{x_1, x_2, \dots, x_n\}$. Une *contrainte* est une expression de la forme $a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b$ où $b \in \mathbb{Q}$ et $(a_1, a_2, \dots, a_n) \in \mathbb{Q}^n \setminus \{(0, 0, \dots, 0)\}$. On notera parfois cette contrainte $\sum_{j=1}^n a_jx_j \leq b$.

L'ensemble des formules $\mathcal{L}_{\text{CPCL}}$ est le plus petit ensemble tel que :

- Toute contrainte est une formule ;
- Si $\varphi \in \mathcal{L}_{\text{CPCL}}$ alors $\neg\varphi \in \mathcal{L}_{\text{CPCL}}$;
- Si $\varphi_1, \varphi_2 \in \mathcal{L}_{\text{CPCL}}$ alors $\varphi_1 \wedge \varphi_2 \in \mathcal{L}_{\text{CPCL}}$ et $\varphi_1 \vee \varphi_2 \in \mathcal{L}_{\text{CPCL}}$.

Le formalisme \mathcal{L}_{CCL} est un sous-ensemble de $\mathcal{L}_{\text{CPCL}}$ réduit aux conjonctions de contraintes linéaires (i.e., $\varphi \in \mathcal{L}_{\text{CCL}}$ si $\varphi \in \mathcal{L}_{\text{CPCL}}$ et les connecteurs \neg et \vee n'apparaissent pas dans φ).

2. La classe des opérateurs de révision définis dans [15] est celle qui respecte les opérateurs (*1–6) et la classe des opérateurs $*^d$ en est une sous-classe particulière. En fait, dans [15] est étudié le cas particulier de $*^d$ où d est la distance de Hamming — pour montrer qu'il coïncide avec l'opérateur défini dans [11] — mais la généralisation à un opérateur $*^d$ pour n'importe quelle d est immédiate.

Un littéral est soit une contrainte (littéral positif) soit une formule de la forme $\neg c$ où c est une contrainte (littéral négatif). Un littéral négatif $\neg(a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b)$ sera aussi noté $a_1x_1 + a_2x_2 + \dots + a_nx_n > b$. De la même façon, on notera parfois $a_1x_1 + a_2x_2 + \dots + a_nx_n \geq b$ pour $-a_1x_1 - a_2x_2 - \dots - a_nx_n \leq -b$ et $a_1x_1 + a_2x_2 + \dots + a_nx_n < b$ pour $-a_1x_1 - a_2x_2 - \dots - a_nx_n > -b$. Par ailleurs, l'expression $a_1x_1 + a_2x_2 + \dots + a_nx_n = b$ est une abréviation pour $(a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b) \wedge (a_1x_1 + a_2x_2 + \dots + a_nx_n \geq b)$ et $a_1x_1 + a_2x_2 + \dots + a_nx_n \neq b$ est une abréviation pour $\neg(a_1x_1 + a_2x_2 + \dots + a_nx_n = b)$. Enfin, le terme $0x_j$ pourra être omis d'une somme apparaissant dans une formule.

3.2 Sémantique

Soit $\Omega = \mathbb{R}^n$. Soit $y = (y_1, y_2, \dots, y_n) \in \Omega$. À toute formule $\varphi \in \mathcal{L}_{\text{CPCL}}$ on associe le sous-ensemble $\mathcal{M}(\varphi)$ de Ω de la façon suivante :

- Si φ est la contrainte $a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b$ alors $\mathcal{M}(\varphi)$ est l'ensemble des $y = (y_1, y_2, \dots, y_n) \in \Omega$ tels que $a_1y_1 + a_2y_2 + \dots + a_ny_n \leq b$;
- Si $\varphi = \neg\psi$ alors $\mathcal{M}(\varphi) = \Omega \setminus \mathcal{M}(\psi)$;
- Si $\varphi = \varphi_1 \wedge \varphi_2$ alors $\mathcal{M}(\varphi) = \mathcal{M}(\varphi_1) \cap \mathcal{M}(\varphi_2)$;
- Si $\varphi = \varphi_1 \vee \varphi_2$ alors $\mathcal{M}(\varphi) = \mathcal{M}(\varphi_1) \cup \mathcal{M}(\varphi_2)$.

Étant donnés, $\varphi, \varphi_1, \varphi_2 \in \mathcal{L}_{\text{CPCL}}$, φ est dit cohérent si $\mathcal{M}(\varphi) \neq \emptyset$, $\varphi_1 \models \varphi_2$ si $\mathcal{M}(\varphi_1) \subseteq \mathcal{M}(\varphi_2)$ et $\varphi_1 \equiv \varphi_2$ si $\mathcal{M}(\varphi_1) = \mathcal{M}(\varphi_2)$. y est un modèle de φ si $y \in \mathcal{M}(\varphi)$.

Remarque : La syntaxe s'appuie sur \mathbb{Q} et non sur \mathbb{R} : le langage $\mathcal{L}_{\text{CPCL}}$ doit être au plus dénombrable. En revanche, la sémantique s'appuie sur l'ensemble des réels : l'étude des contraintes linéaires se fait d'habitude sur \mathbb{R}^n (en particulier, l'optimisation linéaire qui sera utilisée en section 5).

3.3 Propriétés de $\mathcal{L}_{\text{CPCL}}$

Proposition 1 (Mise sous DNF). *Toute formule $\varphi \in \mathcal{L}_{\text{CPCL}}$ est équivalente à une disjonction de conjonctions de littéraux (forme normale disjonctive ou DNF).*

Démonstration. Cette opération de mise sous DNF est la même que celle effectuée en logique propositionnelle, en utilisant les contraintes à la place des variables propositionnelles. \square

Proposition 2. *Soit d la distance de Manhattan sur \mathbb{R}^n (cf. équation (10)). On supposera que les coefficients w_j sont rationnels : $w_j \in \mathbb{Q}$, $w_j > 0$. Soit $\varphi \in \mathcal{L}_{\text{CPCL}}$ et soit $\lambda \in \mathbb{Q}_+$.*

L'ensemble $G = G_\lambda^d(\mathcal{M}(\varphi))$ est représentable dans $\mathcal{L}_{\text{CPCL}}$: il existe $\varphi' \in \mathcal{L}_{\text{CPCL}}$ tel que $\mathcal{M}(\varphi') = G$. De plus, si $\varphi \in \mathcal{L}_{\text{CCL}}$ alors G est représentable dans \mathcal{L}_{CCL} .

Démonstration.

Cas où $\varphi \in \mathcal{L}_{\text{CCL}}$. La formule φ s'écrit donc $\varphi = \bigwedge_{i=1}^p \sum_{j=1}^n a_{ij}x_j \leq b_i$. Par définition de G et de la distance de Manhattan, G est l'ensemble des $y = (y_1, y_2, \dots, y_n) \in \mathbb{R}^n$ tels que $(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n)$ est solution du système d'équations :

$$\begin{cases} \sum_{j=1}^n a_{ij}x_j \leq b_i & \text{pour } i \in \{1, 2, \dots, p\} \\ \sum_{j=1}^n w_j|y_j - x_j| \leq \lambda \end{cases}$$

Ce système n'est pas linéaire (il contient des inégalités qui ne sont pas affines, à cause des valeurs absolues). Pour le « rendre » linéaire, on notera que la contrainte $\sum_{j=1}^n w_j|y_j - x_j| \leq \lambda$ équivaut à la conjonction des contraintes $\sum_{j=1}^n w_j s_j (y_j - x_j) \leq \lambda$ où $s = (s_1, s_2, \dots, s_n) \in \{-1, 1\}^n$. On remplace donc une contrainte non affine par 2^n contraintes affines³. On obtient ainsi un système linéaire sur \mathbb{R}^{2n} tel que la projection sur les n dernières composantes de son ensemble de solutions correspond à G . La projection étant une application linéaire, le résultat est un ensemble qu'on peut représenter par une conjonction de contraintes affines, autrement écrit, G peut s'écrire $G = \mathcal{M}(\varphi')$ pour un $\varphi' \in \mathcal{L}_{\text{CCL}}$.

Cas où φ est une conjonction de littéraux. Ce cas se traite comme le précédent, sauf que certaines contraintes de φ (littéraux positifs) sont à remplacer par des littéraux négatifs. Dans ce cas, on obtient donc également $G = \mathcal{M}(\varphi')$ où φ' est une conjonction de littéraux.

Cas où φ est sous forme DNF : $\varphi = \bigvee_{k=1}^q c_k$ où c_k est une conjonction de littéraux. On a donc, $\mathcal{M}(\varphi) = \bigcup_{k=1}^q \mathcal{M}(c_k)$. D'après (5), on a donc : $G = \bigcup_{k=1}^q G_k$ où $G_k = G_\lambda^d(\mathcal{M}(c_k))$. D'après le cas précédent, $G_k = \mathcal{M}(c'_k)$ pour c'_k , une conjonction de littéraux. Par conséquent, $G = \mathcal{M}(\varphi')$ où $\varphi' = \bigvee_{k=1}^q c'_k$.

Cas général. Soit $\varphi_{\text{DNF}} \in \mathcal{L}_{\text{CPCL}}$, une formule sous DNF telle que $\varphi \equiv \varphi_{\text{DNF}}$. D'après le cas précédent, il existe $\varphi' \in \mathcal{L}_{\text{CPCL}}$ tel que $G_\lambda^d(\mathcal{M}(\varphi_{\text{DNF}})) = \mathcal{M}(\varphi')$. Or $\mathcal{M}(\varphi_{\text{DNF}}) = \mathcal{M}(\varphi)$. Donc, $G = \mathcal{M}(\varphi')$. \square

4 Des opérateurs de révision en $\mathcal{L}_{\text{CPCL}}$

Cette section a pour objectif d'introduire une famille d'opérateurs de révision en $\mathcal{L}_{\text{CPCL}}$ vérifiant les postulats (*1-6).

³ Ce qui n'est, évidemment, pas très efficace en terme algorithmique. La section 5 proposera une solution moins coûteuse.

L'idée est de définir des opérateurs $*^d$ où d est une distance sur Ω . La difficulté ici est que, pour certaines distances, $*^d$ ne vérifie pas les postulats, en particulier le postulat (*3). Par exemple, si $\mathcal{V} = \{x\}$ et $d : (u, v) \in \mathbb{R}^2 \mapsto |v - u|$, alors la révision de $\psi = (x \leq 0)$ par $\mu = (x > 1)$ donnera une formule dont l'ensemble des modèles est vide (il n'existe pas d'élément de l'intervalle non fermé $]1, +\infty[$ à distance minimale de $] - \infty, 0[$). Notons que ce contre-exemple s'applique à toutes les distances équivalentes à d , notamment les distances issues de normes (telles que la distance euclidienne).

On a cependant la forme affaiblie de (*3) donnée par la proposition suivante :

Proposition 3. *Si μ est cohérent et que $\mathcal{M}(\mu)$ est fermé pour d alors $\psi * \mu$ est cohérent.*

Démonstration. Soit $A = \mathcal{M}(\psi)$ et $B = \mathcal{M}(\mu)$. B est par hypothèse non vide et fermé. Soit $d^* = d(A, B) = \inf_{y \in B} d(A, y)$. Il existe donc une suite $(y_n)_{n \in \mathbb{N}}$ d'éléments de B qui converge dans (Ω, d) vers y^* et tels que $(d(A, y_n))_n$ converge vers d^* . Donc, $d(B, y^*) = d(B, \lim_{n \rightarrow +\infty} y_n) = \lim_{n \rightarrow +\infty} d(B, y_n)$ (4). Or $d(B, y_n) = 0$. Par conséquent, $d(B, y^*) = 0$, donc $B \in G_0^d(B)$. Comme B est fermé, on a $y^* \in B$.

Pour résumer, on a mis en évidence $y^* \in B$ tel que $d(A, y^*) = d(A, B)$. Donc, $y^* \in \mathcal{M}(\psi *^d \mu)$ et donc, $\psi *^d \mu$ est cohérent. \square

L'idée est alors de choisir une distance pour laquelle tout sous-ensemble de \mathbb{R}^n est fermé. Une solution consiste à choisir une distance qui prend ses valeurs dans un ensemble discret. Par exemple, la distance d_1 définie de la façon suivante convient :

$$d_1(x, y) = \lceil d(x, y) \rceil$$

où d est définie par l'équation (10). La distance d_1 prend ses valeurs dans \mathbb{N} . Plus généralement, on considérera la famille des distances d_ε , où ε est un rationnel strictement positif, définies par :

$$d_\varepsilon(x, y) = \varepsilon \left\lceil \frac{d(x, y)}{\varepsilon} \right\rceil \quad (14)$$

qui prend ses valeurs dans $\{0, \varepsilon, 2\varepsilon, \dots\} = \varepsilon\mathbb{N}$. Le réel $d_\varepsilon(x, y)$ est une approximation par excès de $d(x, y)$ à ε près, en effet, d'après (3), on a

$$d(x, y) \leq d_\varepsilon(x, y) < d(x, y) + \varepsilon$$

Par conséquent, si A et B sont deux sous-ensembles de \mathbb{R}^n , on a :

$$d(A, B) \leq d_\varepsilon(A, B) \leq d(A, B) + \varepsilon \quad (15)$$

On peut montrer que d_ε est une distance sur \mathbb{R}^n (5).

4. La dernière égalité est vérifiée car d est une application continue de (Ω^2, Δ) dans \mathbb{R}_+ où $\Delta : ((u, v), (x, y)) \in (\Omega^2)^2 \mapsto \max(d(u, x), d(v, y))$.

5. (i) et (ii) sont faciles à montrer. (iii) s'appuie sur l'équation (1).

Proposition 4. *Tout sous-ensemble A de Ω est un fermé de (Ω, d_ε) .*

Démonstration. Soit $A \subseteq \Omega$ et $y \in G_0^{d_\varepsilon}(A)$. Soit $D_\varepsilon(y) = \{d(x, y) \mid x \in A\}$. On a donc $d_\varepsilon(A, x) = \inf D_\varepsilon(y) = 0$. Or $D_\varepsilon(y) \subseteq \varepsilon\mathbb{N}$ est un ensemble discret de réels, donc sa borne inférieure lui appartient : il existe $x \in A$ tel que $d(x, y) = \inf D_\varepsilon(y)$. Donc, $x = y$ et par conséquent, $y \in A$.

Par conséquent, pour tout $y \in G_0^{d_\varepsilon}(A)$, $y \in A$ et donc, A est fermé. \square

Cette proposition va permettre de montrer que $*^{d_\varepsilon}$ vérifie les postulats (voir plus loin, proposition 6) mais, au préalable, la représentabilité dans \mathcal{L}_{CPCL} de l'ensemble d'interprétations donnant $\mathcal{M}(\psi *^{d_\varepsilon} \mu)$ doit être étudiée.

Proposition 5. *Soit $\psi, \mu \in \mathcal{L}_{CPCL}$. Il existe $\varrho \in \mathcal{L}_{CPCL}$ tel que :*

$$\mathcal{M}(\varrho) = \{y \in \mathcal{M}(\mu) \mid d_\varepsilon(\psi, y) = d_\varepsilon^*\}$$

avec $d_\varepsilon^* = d_\varepsilon(\psi, \mu)$. Par conséquent, la révision par $*^{d_\varepsilon}$ est représentable dans \mathcal{L}_{CPCL} et $\psi *^{d_\varepsilon} \mu \equiv \varrho$.

Démonstration. Dans un premier temps, pour A , sous-ensemble de Ω et $\lambda \in \mathbb{R}_+$, montrons le résultat suivant :

$$G_{\lambda_\varepsilon}^{d_\varepsilon}(A) = G_{\lambda_\varepsilon}^d(A) \quad (16)$$

$$\text{où } \lambda_\varepsilon = \varepsilon \left\lceil \frac{\lambda}{\varepsilon} \right\rceil$$

Soit $x, y \in \Omega$. On a les équivalences :

$$d_\varepsilon(x, y) \leq \lambda \quad \text{ssi} \quad \varepsilon \left\lceil \frac{d(x, y)}{\varepsilon} \right\rceil \leq \lambda$$

$$\text{ssi} \quad \left\lceil \frac{d(x, y)}{\varepsilon} \right\rceil \leq \frac{\lambda}{\varepsilon} \quad \text{ssi} \quad \frac{d(x, y)}{\varepsilon} \leq \left\lfloor \frac{\lambda}{\varepsilon} \right\rfloor \quad (\text{d'après (2)})$$

$$\text{ssi} \quad d(x, y) \leq \lambda_\varepsilon$$

Par conséquent :

$$\inf_{x \in A} d_\varepsilon(x, y) \leq \lambda \quad \text{ssi} \quad \inf_{x \in A} d(x, y) \leq \lambda_\varepsilon$$

Ce qui revient à :

$$d_\varepsilon(A, y) \leq \lambda \quad \text{ssi} \quad d(A, y) \leq \lambda_\varepsilon$$

Ce qui entraîne (16).

Soit $R = \{y \in \mathcal{M}(\mu) \mid d_\varepsilon(\psi, y) = d_\varepsilon^*\}$ (on veut montrer que R est représentable dans \mathcal{L}_{CPCL}). Selon l'équivalence des définitions (11) et (12), avec $\lambda = d_\varepsilon^* = d_\varepsilon(\psi, \mu)$ on arrive à $R = G_{\lambda_\varepsilon}^{d_\varepsilon}(\mathcal{M}(\psi)) \cap \mathcal{M}(\mu) = G_{\lambda_\varepsilon}^d(\mathcal{M}(\psi)) \cap \mathcal{M}(\mu)$ d'après (16). Soit alors $\psi' \in \mathcal{L}_{CPCL}$ tel que $\mathcal{M}(\psi') = G_{\lambda_\varepsilon}^d(\mathcal{M}(\psi))$: un tel ψ' existe d'après la proposition 2. On a donc : $R = \mathcal{M}(\psi') \cap \mathcal{M}(\mu) = \mathcal{M}(\varrho)$ où $\varrho = \psi' \wedge \mu \in \mathcal{L}_{CPCL}$, ce qui prouve la proposition. \square

Proposition 6. $*^{d_\varepsilon}$ satisfait les postulats (*1–6).

Démonstration. La preuve complète de cette proposition est dans [16]. Ici, seules les preuves de (*2) et (*3) sont données, les preuves que $*^{d_\varepsilon}$ satisfait les autres postulats données dans [15] (pour la logique propositionnelle et la distance de Hamming) s'appliquent.

(*2) se prouve ainsi. Soit ψ et μ dont la conjonction est cohérente : $\mathcal{M}(\psi) \cap \mathcal{M}(\mu) \neq \emptyset$. On a donc :

$$\begin{aligned} \mathcal{M}(\psi *^{d_\varepsilon} \mu) &= \{y \in \mathcal{M}(\mu) \mid d_\varepsilon(\mathcal{M}(\psi), y) = 0\} \\ &= G_0^{d_\varepsilon}(\mathcal{M}(\psi)) \cap \mathcal{M}(\mu) = \mathcal{M}(\psi) \cap \mathcal{M}(\mu) \end{aligned}$$

puisque $\mathcal{M}(\psi)$, comme tout sous-ensemble de Ω , est fermé pour d_ε (cf. proposition 4). On a donc $\psi *^{d_\varepsilon} \mu \equiv \psi \wedge \mu$.

(*3) est une conséquence des propositions 3 et 4. \square

5 Un algorithme pour réviser dans $\mathcal{L}_{\text{CPCL}}$

Dans cette section comme dans la précédente, d dénote la distance de Manhattan sur \mathbb{R}^n . Dans un premier temps, un algorithme pour l'opérateur de révision $*^d$ sur \mathcal{L}_{CCL} est défini (section 5.1), s'appuyant sur un travail antérieur. Puis, un algorithme pour $*^{d_\varepsilon}$ dans $\mathcal{L}_{\text{CPCL}}$ est présenté (section 5.2) et sa complexité est rapidement étudiée (section 5.3).

5.1 Un algorithme de $*^d$ dans \mathcal{L}_{CCL}

L'opérateur de révision $*^d$ sur \mathcal{L}_{CCL} vérifie les postulats (*1–6) et son calcul peut se ramener à un problème d'optimisation linéaire [5]. En effet, considérons $\psi, \mu \in \mathcal{L}_{\text{CCL}}$. L'ensemble $\mathcal{M}(\psi *^d \mu)$ est l'ensemble des $y = (y_1, y_2, \dots, y_n) \in \mathbb{R}^n$ tels que $(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n) \in \mathbb{R}^{2n}$ est solution du problème d'optimisation suivant dans \mathbb{R}^{2n} :

$$\begin{aligned} &x \in \mathcal{M}(\psi) \\ &y \in \mathcal{M}(\mu) \\ &\text{minimiser } d(x, y) \end{aligned}$$

Ce problème n'est pas linéaire, mais peut se résoudre en cherchant les

$(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n, z_1, z_2, \dots, z_n) \in \mathbb{R}^{3n}$ qui sont solutions du problème d'optimisation linéaire suivant :

$$\left. \begin{aligned} &x \in \mathcal{M}(\psi) \\ &y \in \mathcal{M}(\mu) \\ &z_j \geq y_j - x_j \\ &z_j \geq x_j - y_j \end{aligned} \right\} \text{ pour tout } j \in \{1, 2, \dots, n\} \quad (17)$$

$$\text{minimiser } \sum_{j=1}^n w_j z_j$$

L'idée est que $z_i \geq |y_i - x_i|$ et qu'à l'optimum, cette inégalité devient une égalité.

Comme l'ensemble des solutions d'un problème d'optimisation linéaire est représentable par une conjonction de contraintes linéaires, pour trouver un ϱ tel que $\varrho \equiv \psi *^d \mu$ il suffit de résoudre le problème d'optimisation linéaire (17), de projeter sur les composantes (y_1, y_2, \dots, y_n) et de trouver $\varrho \in \mathcal{L}_{\text{CCL}}$ tel que $\mathcal{M}(\varrho)$ est l'ensemble de ces y .

Il est important pour la suite de noter que cette optimisation permet non seulement de trouver $\psi *^d \mu$ mais également de calculer $d^* = d(\psi, \mu)$, qui est la valeur à l'optimum de la fonction objectif.

5.2 Un algorithme de $*^{d_\varepsilon}$ dans $\mathcal{L}_{\text{CPCL}}$

Soit $\psi, \mu \in \mathcal{L}_{\text{CPCL}}$. Le calcul de $\varrho = \psi *^{d_\varepsilon} \mu$ est décrit pour trois cas, chacun des deux derniers cas faisant appel au précédent.

5.2.1 Cas où ψ et μ sont des conjonctions de littéraux

Tout d'abord, un test de cohérence est effectué. Si ψ est incohérente ou si μ l'est, $\psi *^{d_\varepsilon} \mu \equiv \mu$ avec $d_\varepsilon^* = +\infty$, d'après (13). Dans la suite de cette section, on supposera ψ et μ cohérents.

D'après (12), $\mathcal{M}(\psi *^{d_\varepsilon} \mu) = G_{d_\varepsilon^*}^{d_\varepsilon}(\mathcal{M}(\psi)) \cap \mathcal{M}(\mu)$. Or, d'après (16), $G_{d_\varepsilon^*}^{d_\varepsilon}(\mathcal{M}(\psi)) = G_{\lambda_\varepsilon}^d(\mathcal{M}(\psi))$ où $\lambda_\varepsilon = \varepsilon \left\lfloor \frac{d_\varepsilon^*}{\varepsilon} \right\rfloor$. Or, d_ε est à valeur dans $\varepsilon\mathbb{N}$, d'où $\lambda_\varepsilon = \varepsilon \frac{d_\varepsilon^*}{\varepsilon} = d_\varepsilon^*$. Par ailleurs, d'après (15) on a :

$$d^* \leq d_\varepsilon^* \leq d^* + \varepsilon \quad (18)$$

Comme $d_\varepsilon^* \in \varepsilon\mathbb{N}$, deux cas sont possibles :

- $d^* \in \varepsilon\mathbb{N}$, auquel cas, l'encadrement (18) donne deux valeurs possibles pour d_ε^* : d^* et $d^* + \varepsilon$ (6) ;
- $d^* \notin \varepsilon\mathbb{N}$, auquel cas, l'encadrement (18) donne une seule valeur possible : $d_\varepsilon^* = \varepsilon \left\lceil \frac{d^*}{\varepsilon} \right\rceil$.

Par conséquent, $\psi *^{d_\varepsilon} \mu \equiv \varrho$ avec

$$\varrho = \psi' \wedge \mu \quad \text{avec } \mathcal{M}(\psi') = G_{\lambda_\varepsilon}^d(\mathcal{M}(\psi)) \text{ et } \lambda_\varepsilon = d_\varepsilon^*$$

Pour calculer ϱ , on procède de la façon suivante :

- Calcul de d^* ;
- Si $d^* \in \varepsilon\mathbb{N}$ alors soit $\lambda_\varepsilon = d^*$ sinon soit $\lambda_\varepsilon = \varepsilon \left\lceil \frac{d^*}{\varepsilon} \right\rceil$;
- On cherche ψ' tel que $\mathcal{M}(\psi') = G_{\lambda_\varepsilon}^d(\mathcal{M}(\psi))$;
- Si $d^* \notin \varepsilon\mathbb{N}$, (cas (b) ci-dessus), le résultat de la révision est $\psi' \wedge \mu$ avec $d_\varepsilon^* = \lambda_\varepsilon$;

6. Dans ce cas, la valeur de d^* n'est pas suffisante pour choisir entre ces deux valeurs. Par exemple, si on considère, dans un formalisme avec une seule variable, ψ, μ_1 et μ_2 les formules dont les ensembles de modèles sont respectivement $[0; 1]$, $[2; 3]$ et $]2, 3]$, on a $d(\psi, \mu_1) = d(\psi, \mu_2) = 1$ alors que $d_\varepsilon(\psi, \mu_1) = 1$ et $d_\varepsilon(\psi, \mu_2) = 1, 2$ pour $\varepsilon = 0, 2$.

5. Sinon (cas (a)), on a soit $d_\varepsilon^* = d^*$, soit $d_\varepsilon^* = d^* + \varepsilon$. On teste si $\psi' \wedge \mu$ est cohérent.

Si c'est le cas, alors cela signifie que $d_\varepsilon^* = d^*$ et $\psi' \wedge \mu$ sera le résultat de la révision ;

6. Sinon on prend $\lambda_\varepsilon = d^* + \varepsilon$ et on cherche à nouveau ψ' tel que $\mathcal{M}(\psi') = G_{\lambda_\varepsilon}^d(\mathcal{M}(\psi))$, le résultat de la révision étant $\psi' \wedge \mu$.

Il suffit donc de savoir calculer d^* et ψ' .

Calcul de d^* . L'optimisation linéaire permet de calculer d^* dans le cas où ψ et μ sont des conjonctions de contraintes. Dans le cas plus général où ψ et μ sont des conjonctions de littéraux, on peut se ramener à ce cas de la façon suivante.

Dans un premier temps, une nouvelle notation est introduite. Pour φ une conjonction de littéraux, on note $\widehat{\varphi}$, la conjonction de contraintes obtenue en remplaçant les inégalités strictes par des inégalités larges, i.e., en remplaçant les littéraux négatifs $\neg(a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b)$ par des contraintes $-a_1x_1 - a_2x_2 - \dots - a_nx_n \leq -b$. Par exemple, si $\varphi = (2x_1 - 3x_2 < 5)$ alors $\widehat{\varphi} \equiv (2x_1 - 3x_2 \leq 5)$. Un premier résultat montre que $\widehat{\varphi}$ est une représentation de l'adhérence de $\mathcal{M}(\varphi)$.

Proposition 7. Soit φ une conjonction cohérente de littéraux de \mathcal{L}_{CPCL} . On a $\mathcal{M}(\widehat{\varphi}) = G_0^d(\mathcal{M}(\varphi))$.

Démonstration. $\varphi = \bigwedge_{i=1}^p \varphi_i$ où chaque φ_i est une conjonction de littéraux. On notera $A_i = \mathcal{M}(\varphi_i)$ et $A = \bigcap_{i=1}^p A_i$, d'où $\mathcal{M}(\varphi) = A$.

Cas $p = 1$. φ est donc un littéral : $\varphi = (\sum_{j=1}^p a_j x_j R b)$ où $R = \leq$ si φ_i est un littéral positif et $R = <$ sinon.

Soit $x \in \mathcal{M}(\widehat{\varphi})$, on a donc $\sum_{j=1}^p a_j x_j \leq b$. Soit $\alpha > 0$ et soit $u(\alpha) = (u_1(\alpha), u_2(\alpha), \dots, u_n(\alpha))$ avec $u_j(\alpha) = x_j - s_j \alpha$ où s_j est le signe de a_j : $s_j = 1$ si $a_j \geq 0$ et $s_j = 0$ sinon (et donc $s_j a_j = |a_j|$). On a donc :

$$\sum_{j=1}^p a_j u_j(\alpha) = \sum_{j=1}^p a_j x_j - \alpha \sum_{j=1}^p |a_j| < b$$

puisque $\sum_{j=1}^p |a_j| > 0$, étant donné que $(a_1, a_2, \dots, a_n) \neq (0, 0, \dots, 0)$. Par conséquent, pour tout $\alpha > 0$, $u(\alpha) \in A$. Comme $u(\alpha)$ tend vers x quand α tend vers 0, on a donc $d(A, x) = 0$ et donc $x \in G_0^d(A)$.

Réciproquement, soit $x \in G_0^d(A) : d(A, x) = 0$. Soit $\alpha > 0$. Il existe donc $y \in A$ tel que $d(y, x) \leq \alpha$. Soit $m = y - x$. On a donc : $\sum_{j=1}^p w_j |m_j| \leq \alpha$ d'où $\sum_{j=1}^p |m_j| \leq \frac{\alpha}{w_{\min}}$ avec $w_{\min} = \min_{j=1}^p w_j$. On a donc :

$$\sum_{j=1}^p a_j x_j = \sum_{j=1}^p a_j (y_j - m_j) = \sum_{j=1}^p a_j y_j - \sum_{j=1}^p a_j m_j$$

Or $|\sum_{j=1}^p a_j w_j| \leq a_{\max} \sum_{j=1}^p w_j \leq a_{\max} \frac{\alpha}{w_{\min}}$ où $a_{\max} = \max_{j=1}^p |a_j|$. Comme $y \in A$, on a donc :

$$\sum_{j=1}^p a_j x_j \leq b + f(\alpha)$$

où $f(\alpha)$ est majoré par $a_{\max} \frac{\alpha}{w_{\min}}$ et donc tend vers 0 quand α tend vers 0. Par conséquent, en faisant ce passage à la limite, on arrive à $\sum_{j=1}^p a_j x_j \leq b$, i.e., $x \in \mathcal{M}(\widehat{\varphi})$.

Cas $p \geq 2$. On veut montrer que $\bigcap_{i=1}^p G_0^d(A_i) = G_0^d(A)$. En effet, si cette égalité est réalisée, puisque $G_0^d(A_i) = \mathcal{M}(\widehat{\varphi}_i)$ (cf. cas $p = 1$) on aura $\mathcal{M}(\widehat{\varphi}) = G_0^d(A)$.

L'inclusion $G_0^d(A) \subseteq \bigcap_{i=1}^p G_0^d(A_i)$ est une conséquence immédiate de (6). Montrons l'inclusion inverse. Soit $x \in \bigcap_{i=1}^p G_0^d(A_i)$, il suffit donc de montrer que $x \in G_0^d(A)$. φ étant cohérente, $A \neq \emptyset$. Soit $y \in A$. Pour $\alpha \in]0; 1]$, soit $u(\alpha) = (1 - \alpha)x + \alpha y$. $u(\alpha) = (u_1(\alpha), u_2(\alpha), \dots, u_n(\alpha)) \in \mathbb{R}^n$. On va montrer que pour tout $i \in \{1, 2, \dots, p\}$, $u(\alpha) \in A_i$. φ_i est un littéral, il s'écrit donc (à l'équivalence près) : $\varphi_i \equiv (\sum_{j=1}^n a_{ij} x_j R b_i)$ où $R \in \{<, \leq\}$. Le fait que $x \in G_0^d(A_i)$ et $y \in A_i$ se traduit par les deux inéquations suivantes :

$$\sum_{j=1}^n a_{ij} x_j \leq b_i \quad \sum_{j=1}^n a_{ij} y_j R b_i$$

En multipliant ces deux inéquations respectivement par $(1 - \alpha)$ et α et en faisant la somme, on obtient :

$$\sum_{j=1}^n a_{ij} u_j(\alpha) R b_i$$

d'où $u(\alpha) \in A_i$. Par conséquent, $u(\alpha) \in A = \bigcap_{i=1}^p A_i$. Par ailleurs, $u(\alpha)$ tend vers x quand α tend vers 0. Par conséquent, $d(A, x) = 0$ et donc, $x \in G_0^d(A)$, ce qui achève la preuve. \square

En s'appuyant sur ce résultat, on peut prouver le résultat suivant :

Proposition 8. Si d est la distance de Manhattan et que ψ et μ sont des conjonctions de littéraux de \mathcal{L}_{CPCL} alors $d(\psi, \mu) = d(\widehat{\psi}, \widehat{\mu})$.

Démonstration. Soit $A = \mathcal{M}(\psi)$, $B = \mathcal{M}(\mu)$, $\widehat{A} = \mathcal{M}(\widehat{\psi})$ et $\widehat{B} = \mathcal{M}(\widehat{\mu})$. On a, d'après la proposition 7, $\widehat{A} = G_0^d(A)$ et $\widehat{B} = G_0^d(B)$, d'où $A \subseteq \widehat{A}$ et $B \subseteq \widehat{B}$. Par conséquent, $d(\psi, \mu) \geq d(\widehat{\psi}, \widehat{\mu})$.

Montrons l'inégalité réciproque. Soit $x \in A$, $y \in B$, $\widehat{x} \in \widehat{A}$ et $\widehat{y} \in \widehat{B}$. D'après l'inégalité triangulaire, on a :

$$d(x, y) \leq d(x, \widehat{x}) + d(\widehat{x}, \widehat{y}) + d(\widehat{y}, y)$$

En passant à la borne inférieure sur x et y , on obtient (en utilisant la symétrie de d) :

$$d(A, B) \leq d(A, \hat{x}) + d(\hat{x}, \hat{y}) + d(B, \hat{y})$$

En passant à la borne inférieure sur \hat{x} et \hat{y} , on obtient

$$d(A, B) \leq 0 + d(\hat{A}, \hat{B}) + 0$$

puisque $\hat{x} \in \hat{A} = G_0^d(A)$ et $\hat{y} \in \hat{B} = G_0^d(B)$. On arrive donc à $d(\psi, \mu) \leq d(\hat{\psi}, \hat{\mu})$ ce qui achève de prouver la proposition. \square

Le calcul de d^* se fait alors en effectuant la révision $\hat{\psi} *^d \hat{\mu}$ telle que décrite à la section 5.1, qui permet d'obtenir $d(\hat{\psi}, \hat{\mu})$, égal à la valeur d^* recherchée.

Calcul de ψ' . ψ' est déterminé par $\mathcal{M}(\psi') = G_{\lambda_\varepsilon}^d(\mathcal{M}(\psi))$. Il suffit donc d'avoir un algorithme qui, étant donné un rationnel $\lambda \geq 0$ et une conjonction de littéraux φ donne une formule φ_λ telle que $\mathcal{M}(\varphi_\lambda) = G_\lambda^d(\mathcal{M}(\varphi))$. La preuve de la proposition 2 met en évidence un tel algorithme, qui a le défaut d'être très coûteux en temps, puisqu'il consiste en l'introduction de 2^n littéraux supplémentaires. La proposition suivante permet de calculer φ_λ de façon plus efficace et s'inspire de la transformation décrite dans la section 5.1.

Proposition 9. Soit φ , une conjonction de littéraux de \mathcal{L}_{CPCL} et $\lambda \in \mathbb{Q}_+$. Le système suivant :

$$\begin{aligned} x &\in \mathcal{M}(\varphi) \\ z_j &\geq y_j - x_j \\ z_j &\geq x_j - y_j \quad \text{pour tout } j \in \{1, 2, \dots, n\} \\ \sum_{j=1}^n w_j z_j &\leq \lambda \end{aligned}$$

peut s'exprimer comme une conjonction de littéraux de \mathcal{L}_{CPCL} avec un ensemble de $3n$ variables $\mathcal{V} = \{x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n, z_1, z_2, \dots, z_n\}$. La projection de ce système sur (y_1, y_2, \dots, y_n) s'exprime comme une conjonction de littéraux φ_λ avec n variables et $\mathcal{M}(\varphi_\lambda) = G_\lambda^d(\mathcal{M}(\varphi))$.

Démonstration. Soit $G_1 = G_\lambda^d(\mathcal{M}(\varphi))$ et G_2 l'ensemble des $y \in \mathbb{R}^n$ vérifiant le système donné dans la proposition, autrement écrit :

$$G_1 = \left\{ y \in \mathbb{R}^n \left| \begin{array}{l} \text{il existe } x \in \mathbb{R}^n, \\ x \in \mathcal{M}(\varphi) \text{ et} \\ \sum_{j=1}^n w_j |y_j - x_j| \leq \lambda \end{array} \right. \right\}$$

$$G_2 = \left\{ y \in \mathbb{R}^n \left| \begin{array}{l} \text{il existe } x, y \in \mathbb{R}^n, \\ x \in \mathcal{M}(\varphi), \\ z_j \geq |y_j - x_j| \text{ pour tout } j \\ \text{et } \sum_{j=1}^n w_j z_j \leq \lambda \end{array} \right. \right\}$$

Il suffit de prouver que $G_1 = G_2$. D'après (9), il suffit de le montrer pour $\mathcal{M}(\varphi)$ fermé. On supposera donc que c'est le cas et donc que G_1 et G_2 sont fermés.

Soit $y \in G_1$. Il existe donc $x \in \mathbb{R}^n$ tel que $x \in \mathcal{M}(\varphi)$ et $\sum_{j=1}^n w_j |y_j - x_j| \leq \lambda$. Soit $z = (z_1, z_2, \dots, z_n) \in \mathbb{R}^n$ tel que $z_j = |y_j - x_j|$ pour tout $j \in \{1, 2, \dots, n\}$. On a donc $\sum_{j=1}^n w_j z_j \leq \lambda$ et donc $x \in G_2$.

Réciproquement, soit $y \in G_2$. Il existe donc $x, z \in \mathbb{R}^n$ tels que $x \in \mathcal{M}(\varphi)$, $z_j \geq |y_j - x_j|$ et $\sum_{j=1}^n w_j z_j \leq \lambda$. Par conséquent, $\sum_{j=1}^n w_j |y_j - x_j| \leq \lambda$ et donc, $y \in G_1$. \square

5.2.2 Cas où ψ et μ sont sous DNF

La proposition suivante et sa preuve sont reprises et adaptées de [12] : dans cet article, il avait été appliqué à la révision dans le formalisme de la clôture propositionnelle d'une algèbre qualitative. Le résultat se transpose à \mathcal{L}_{CPCL} en tenant compte du fait que l'ensemble des interprétations de ce formalisme est continu : toute borne inférieure n'est pas un minimum.

Proposition 10. Soit ψ et μ deux formules de \mathcal{L}_{CPCL} et $\{\psi_k\}_k$ et $\{\mu_\ell\}_\ell$ deux familles finies de \mathcal{L}_{CPCL} telles que $\psi = \bigvee_k \psi_k$ et $\mu = \bigvee_\ell \mu_\ell$. Soit $d_{\varepsilon k\ell}^* = d_\varepsilon(\psi_k, \mu_\ell)$ pour tout k et tout ℓ . On a :

$$\begin{aligned} \psi *^{d_\varepsilon} \mu &\equiv \bigvee_{k, \ell, d_{\varepsilon k\ell}^* = d_\varepsilon} \psi_k *^{d_\varepsilon} \mu_\ell \\ \text{avec } d_\varepsilon^* &= d_\varepsilon(\psi, \mu) = \min_{k\ell} d_{k\ell}^* \end{aligned} \quad (19)$$

Démonstration. (19) est prouvé tout d'abord :

$$\begin{aligned} d_\varepsilon^* &= d_\varepsilon(\psi, \mu) = d_\varepsilon \left(\bigcup_k \mathcal{M}(\psi_k), \bigcup_\ell \mathcal{M}(\mu_\ell) \right) \\ &= \min_{k\ell} d(\psi_k, \mu_\ell) = \min_{k\ell} d_{k\ell}^* \end{aligned}$$

Soit à présent $y \in \mathcal{M}(\psi *^{d_\varepsilon} \mu)$. $d_\varepsilon(\psi, y) = d_\varepsilon^*$, donc il existe un couple (k, ℓ) tel que $y \in \mathcal{M}(\mu_\ell)$ et $d_\varepsilon(\psi_k, y) = d_{\varepsilon k\ell}^* = d_\varepsilon^*$. Or $d_\varepsilon(\psi_k, y) = \inf_{x \in \mathcal{M}(\psi_k)} d_\varepsilon(x, y)$ donc, pour tout $\alpha > 0$, il existe un $x \in \mathcal{M}(\psi_k)$ tel que $d_\varepsilon(x, y) \leq d_\varepsilon(\psi, y) + \alpha$ et on a donc la chaîne de relation :

$$d_\varepsilon^* = d_\varepsilon(\psi, y) \leq d_\varepsilon(x, y) \leq d_\varepsilon(\psi, y) + \alpha = d_\varepsilon^* + \alpha$$

Par conséquent, pour tout $\alpha > 0$ il existe $x \in \mathcal{M}(\psi_k)$ tel que $d_\varepsilon^* \leq d_\varepsilon(x, y) \leq d_\varepsilon^* + \alpha$. En passant à la limite quand α tend vers 0, on peut en déduire que $d_\varepsilon(\psi_k, y) \leq d_\varepsilon^*$. Comme $d_\varepsilon(\psi_k, y) \geq d_\varepsilon^*$ et $d_\varepsilon^* = d_{\varepsilon k\ell}^*$ on en conclut que $y \in \mathcal{M}(\psi_k *^{d_\varepsilon} \mu_\ell)$. En résumé, si $y \in \mathcal{M}(\psi *^{d_\varepsilon} \mu)$ alors il existe (k, ℓ) tel que $y \in \mathcal{M}(\psi_k *^{d_\varepsilon} \mu_\ell)$. Donc $\psi *^{d_\varepsilon} \mu \models \bigvee_{k, \ell, d_{\varepsilon k\ell}^* = d_\varepsilon^*} \psi_k *^{d_\varepsilon} \mu_\ell$.

Réciproquement, soit $y \in \mathcal{M}(\psi_k *^{d_\varepsilon} \mu_\ell)$ pour k et ℓ tels que $d_{\varepsilon k\ell}^* = d_\varepsilon^*$. Cela entraîne que $d_\varepsilon(\psi_k, y) = d_\varepsilon^*$, d'où la chaîne de relations :

$$d_\varepsilon^* \leq d_\varepsilon(\psi, y) \leq d_\varepsilon(\psi_k, y) = d_\varepsilon^*$$

donc $d_\varepsilon(\psi, y) = d_\varepsilon^*$ avec $y \in \mathcal{M}(\mu)$ et, par conséquent, $y \in \mathcal{M}(\psi *^{d_\varepsilon} \mu)$. Donc $\bigvee_{k,\ell, d_{\varepsilon k\ell}^* = d_\varepsilon^*} \psi_k *^{d_\varepsilon} \mu_\ell \models \psi *^{d_\varepsilon} \mu$, ce qui achève de prouver la proposition. \square

En appliquant cette proposition à deux formules ψ et μ sous DNF, on peut donc calculer la révision de ψ par μ selon $*^{d_\varepsilon}$, en la ramenant à la révision de conjonctions de littéraux, laquelle a été traitée dans la section précédente.

5.2.3 Cas général

Soit ψ_{DNF} et μ_{DNF} , deux formules sous DNF équivalentes respectivement à ψ et à μ . D'après le postulat (*4) — que vérifie $*^{d_\varepsilon}$ — $\psi *^{d_\varepsilon} \mu \equiv \psi_{\text{DNF}} *^{d_\varepsilon} \mu_{\text{DNF}}$: le calcul de la révision de ψ par μ se ramène donc au cas précédent.

5.3 Complexité de l'algorithme

Cette section présente une analyse grossière de la complexité de l'algorithme. Celui-ci fait appel à la résolution de plusieurs problèmes :

- La mise sous forme DNF pour ψ et μ ;
- L'optimisation linéaire pour le calcul de d^* ;
- Le test de cohérence d'une conjonction de littéraux de $\mathcal{L}_{\text{CPCL}}$.

Les deux derniers problèmes peuvent se traiter en temps polynomiaux. En revanche, le premier requiert un temps exponentiel dans le pire cas. Par conséquent, la complexité de l'algorithme dans le pire cas est exponentielle.

6 Généralisation aux contraintes mixtes

Un cadre plus général que $\mathcal{L}_{\text{CPCL}}$ est celui dans lequel les variables peuvent être de type entier ou de type réel : on parle de contraintes linéaires mixtes. Dans cette section, nous parcourrons rapidement les sections 3 à 5 pour voir ce que ce cadre plus général change et ce qu'il apporte.

La logique ($\mathcal{L}_{\text{CPCLM}}, \models$). La syntaxe de $\mathcal{L}_{\text{CPCLM}}$ diffère de celle de $\mathcal{L}_{\text{CPCL}}$ en ce que l'ensemble des variables est partagé en deux : les variables de type réel et celles de type entier. Pour indiquer cette différence, on utilise une fonction $\tau : \mathcal{V} \rightarrow \{\mathbb{R}, \mathbb{Z}\}$: si $\tau(x_j) = \mathbb{R}$, x_j sera dit de type réel, sinon, il sera dit de type entier. Le reste de la syntaxe est inchangé.

Pour la sémantique, l'ensemble des interprétations sera $\Omega = \tau(x_1) \times \tau(x_2) \times \dots \times \tau(x_n)$ et le reste est inchangé.

Les propositions 1 (mise sous DNF) et 2 (représentabilité de $G_\lambda^d(\mathcal{M}(\varphi))$) sont toujours vérifiées. Pour cette dernière, la distance de Manhattan est la restriction à Ω de la distance définie par (10).

Il est possible de voir dans cette logique une extension de la logique propositionnelle \mathcal{L}_P avec un nombre fini de variables \mathcal{V} . En effet, considérons la fonction qui à $\varphi \in \mathcal{L}_P$ associe $\underline{\varphi} \in \mathcal{L}_{\text{CPCLM}}$ obtenue en remplaçant chaque variable propositionnelle $a \in \mathcal{V}$ par une contrainte $\underline{a} \geq 1$ avec \underline{a} , une variable de type entier. Par exemple, si $\varphi = \neg(a \vee b)$ alors, $\underline{\varphi} = \neg(\underline{a} \geq 1 \vee \underline{b} \geq 1)$, qui équivaut à $\underline{\varphi} \leq 0 \wedge \underline{b} \leq 0$. On peut montrer que $\varphi_1 \models \varphi_2$ dans \mathcal{L}_P ssi $\underline{\varphi}_1 \models \underline{\varphi}_2$ dans $\mathcal{L}_{\text{CPCLM}}$.

Des opérateurs de révision en $\mathcal{L}_{\text{CPCLM}}$. Soit $\varepsilon \in \mathbb{Q}_+$, $\varepsilon > 0$ et soit d_ε défini par l'équation (14). L'opérateur $*^{d_\varepsilon}$ sur $\mathcal{L}_{\text{CPCLM}}$ partage les propriétés de $*^{d_\varepsilon}$ sur $\mathcal{L}_{\text{CPCL}}$ décrites en section 4, i.e., les propositions 4 (tous les sous-ensembles de Ω sont des fermés de (Ω, d_ε)), 5 (représentabilité dans $\mathcal{L}_{\text{CPCLM}}$ de $\psi *^{d_\varepsilon} \mu$) et 6 ($*^{d_\varepsilon}$ satisfait les postulats (*1–6)).

Par ailleurs, le résultat suivant montre un lien entre une classe d'opérateurs de révision sur la logique propositionnelle \mathcal{L}_P et les opérateurs $*^{d_\varepsilon}$ sur $\mathcal{L}_{\text{CPCLM}}$.

Proposition 11. Soit \mathcal{V} un ensemble fini de n variables propositionnelles, Ω_P , l'ensemble des interprétations construites sur \mathcal{V} et d_H la distance de Hamming pondérée définie sur Ω_P par :

$$d_H(x, y) = \sum_{j=1}^n \begin{cases} 0 & \text{si } x_j = y_j \\ w_j & \text{sinon} \end{cases}$$

Pour toutes formules ψ et μ de \mathcal{L}_P , on a :

$$\underline{\psi *^{d_H} \mu} \wedge \beta \equiv \underline{\psi} *^{d_1} (\underline{\mu} \wedge \beta)$$

pour d , la distance de Manhattan construite sur les mêmes w_j et $\beta \in \mathcal{L}_{\text{CPCLM}}$ tel que $\mathcal{M}(\beta) = \{0, 1\}^n$.

Dans cette proposition, β sert à se restreindre aux entiers de la paire $\{0, 1\}$. Cela permet en outre de calculer $\psi *^{d_H} \mu \in \mathcal{L}_P$ en partant de $\underline{\psi *^{d_H} \mu} \wedge \beta \in \mathcal{L}_{\text{CPCLM}}$.

Principe de la démonstration. La preuve s'appuie sur la bijection f entre les interprétations de \mathcal{L}_P et celles parmi les interprétations y de $\mathcal{L}_{\text{CPCLM}}$ tels que $y \in \{0, 1\}^n = \mathcal{M}(\beta) : f : \omega \mapsto y$ avec $y_j = 1$ si $\omega \models x_j$ (où $\mathcal{V} = \{x_1, x_2, \dots, x_j\}$). \square

Algorithme de $*^{d_\varepsilon}$ dans $\mathcal{L}_{\text{CPCLM}}$. L'algorithme décrit dans la section 5 se transpose au cadre de $\mathcal{L}_{\text{CPCLM}}$, la différence principale étant l'appel à un système d'optimisation linéaire mixte : le problème sous-jacent est NP-difficile (alors qu'il est polynomial quand toutes les variables sont de type réel).

7 Conclusion

Cet article présente une approche pour la révision des croyances dans la clôture propositionnelle des contraintes linéaires (en nombres réels et mixte), les opérateurs de révision définis s'appuyant sur une distance à ensemble de valeurs discrètes, d_ε , approximation par excès à $\varepsilon > 0$ de la distance de Manhattan. Le choix d'une telle distance est justifié par le fait que les distances usuellement utilisées sur \mathbb{R}^n donnent des opérateurs ne vérifiant pas tous les postulats AGM.

Une perspective de cette étude est l'implantation de ces opérateurs de révision $*^{d_\varepsilon}$ et leur utilisation au sein d'un système de RÀPC, en profitant du fait que $*^{d_\varepsilon}$ dans \mathcal{L}_{CPCLM} peut être vu comme une extension d'une révision à la fois en logique propositionnelle et dans le langage des conjonctions de contraintes.

Une autre perspective, développée dans [16], est l'étude de la complémentarité avec l'approche présentée dans [8].

Remerciements

L'auteur tient à remercier les deux relecteurs anonymes pour leur travail encourageant et constructif.

Références

- [1] C. E. Alchourrón, P. Gärdenfors, and D. Makinson. On the Logic of Theory Change : partial meet functions for contraction and revision. *Journal of Symbolic Logic*, 50 :510–530, 1985.
- [2] S. Benferhat, Z. Bouraoui, O. Papini, and É. Würbel. Révision par r-ensembles de bases de connaissances en DL-LiteR. In *Reconnaissance de Formes et Intelligence Artificielle (RFIA) 2014*, France, June 2014.
- [3] I. Bloch and J. Lang. Towards mathematical morphologies. In *Technologies for Constructing Intelligent Systems 2*, pages 367–380. Springer, 2002.
- [4] L. Chang, U. Sattler, and T. Gu. Algorithm for adapting cases represented in a tractable description logic. In *Case-Based Reasoning Research and Development, Proceedings of ICCBR-2014*, pages 63–78. Springer, 2014.
- [5] J. Cojan and J. Lieber. Une approche de l'adaptation en raisonnement à partir de cas fondée sur l'optimisation sous contraintes. In L. Cholvy and S. Konieczny, editors, *Journées d'Intelligence Artificielle Fondamentale*, Marseille France, 2009.
- [6] J. Cojan and J. Lieber. An Algorithm for Adapting Cases Represented in ACC. In *22th International Joint Conference on Artificial Intelligence*, Barcelone Espagne, 07 2011.
- [7] J. Cojan and J. Lieber. Applying Belief Revision to Case-Based Reasoning. In H. Prade and G. Richard, editors, *Computational Approaches to Analogical Reasoning : Current Trends*, volume 548 of *Studies in Computational Intelligence*, pages 133 – 161. Springer, 2014.
- [8] J.-F. Condotta, S. Kaci, P. Marquis, and N. Schwind. Majority Merging : from Boolean Spaces to Affine Spaces. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI-10), Lisbon*, pages 627–632, 2010.
- [9] A. Cordier, V. Dufour-Lussier, J. Lieber, E. Nauer, F. Badra, J. Cojan, E. Gaillard, L. Infante-Blanco, P. Molli, A. Napoli, and H. Skaf-Molli. Taaable : a Case-Based System for personalized Cooking. In Stefania Montani and Lakhmi C. Jain, editors, *Successful Case-based Reasoning Applications-2*, volume 494 of *Studies in Computational Intelligence*, pages 121–162. Springer, January 2014.
- [10] N. Creignou, O. Papini, R. Pichler, and S. Woltran. Belief revision within fragments of propositional logic. *Journal of Computer and System Sciences*, 80(2) :427–449, 2014.
- [11] M. Dalal. Investigations into a theory of knowledge base revision : Preliminary report. In *AAAI*, pages 475–479, 1988.
- [12] V. Dufour-Lussier, A. Hermann, F. Le Ber, and J. Lieber. Révision des croyances dans la clôture propositionnelle d'une algèbre qualitative. In *JIAF-2014 – Huitièmes Journées de l'Intelligence Artificielle Fondamentale*, JIAF - Huitièmes Journées de l'Intelligence Artificielle Fondamentale, page 10, Angers, France, June 2014.
- [13] V. Dufour-Lussier, F. Le Ber, J. Lieber, and L. Martin. Case Adaptation with Qualitative Algebras. In Francesca Rossi, editor, *International Joint Conferences on Artificial Intelligence*, pages 3002–3006, Pékin, Chine, August 2013. AAAI Press.
- [14] J. Hué and M. Westphal. Revising qualitative constraint networks : Definition and implementation. In *Tools with Artificial Intelligence (ICTAI)*, pages 548–555, 2012.
- [15] H. Katsuno and A. Mendelzon. Propositional knowledge base revision and minimal change. *Artificial Intelligence*, 52(3) :263–294, 1991.
- [16] J. Lieber. Révision des croyances dans une clôture propositionnelle de contraintes linéaires (version étendue). Rapport technique du Loria, UMR 7503, Université de Lorraine, CNRS, 2015.
- [17] P. Peppas. Belief Revision. In F. van Harmelen, V. Lifschitz, and B. Porter, editors, *Handbook*

- of Knowledge Representation*, chapter 8, pages 317–359. Elsevier, 2008.
- [18] G. Personeni, A. Hermann, and J. Lieber. Adapting Propositional Cases Based on Tableaux Repairs Using Adaptation Knowledge. In L. Lamontagne and E. Plaza, editors, *ICCBR*, number 8765 in LNAI, pages 390–404, Cork, 2014. Springer.
- [19] C. K. Riesbeck and R. C. Schank. *Inside Case-Based Reasoning*. Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, 1989.
- [20] C. Schwind. From inconsistency to consistency : Knowledge base revision by tableaux opening. In *Advances in Artificial Intelligence–IBERAMIA 2010*, pages 120–132. Springer, 2010.

A natural sequent calculus for Lewis' counterfactual logic ∇

Nicola Olivetti¹ Gian Luca Pozzato²

¹ Aix Marseille Université
CNRS, ENSAM, Université de Toulon
LSIS UMR 7296, 13397, Marseille, France

² Università degli Studi di Torino
Dipartimento di Informatica, c.so Svizzera 185
10049, Torino, Italy

nicola.olivetti@univ-amu.fr gianluca.pozzato@unito.it

Résumé

La logique ∇ est la logique de contre-factuels la plus générale dans la famille des systèmes de Lewis. Elle est caractérisée par la classe de modèles à sphère. Dans cet article, nous proposons un nouveau calcul des séquents pour cette logique. Notre calcul est formulé en utilisant le connecteur de plausibilité comparative \preceq introduit par Lewis : une formule $A \preceq B$ signifie intuitivement que A est au moins aussi plausible que B , de sorte qu'un conditionnel $A \Rightarrow B$ peut être défini comme A est impossible ou $A \wedge \neg B$ est moins plausible que A . Différemment des tentatives précédents, notre calcul est « standard » dans le sens que chaque connecteur est traité par un nombre fini de règles avec un nombre fixe et limité des prémisses. De plus, notre calcul est “interne”, dans le sens que chaque séquent peut être directement traduit dans une formule du langage. La caractéristique de notre calcul est que les séquents contiennent un type particulier de structures, appelées *blocks*, qui représentent ou codent une combinaison finie de formules avec \preceq . Nous montrons que le calcul est terminant, en conséquence il fournit une procédure de décision pour la logique ∇ .

Abstract

The logic ∇ is the basic logic of counterfactuals in the family of Lewis' systems. It is characterized by the whole class of so-called sphere models. We propose a new sequent calculus for this logic. Our calculus takes as primitive Lewis' connective of comparative plausibility \preceq : a formula $A \preceq B$ intuitively means that A is at least as plausible as B , so that a conditional $A \Rightarrow B$ can be defined as A is impossible or $A \wedge \neg B$ is less plausible than A . As a difference with previous attempts,

our calculus is standard in the sense that each connective is handled by a finite number of rules with a fixed and finite number of premises. Moreover our calculus is “internal”, in the sense that each sequent can be directly translated into a formula of the language. The peculiarity of our calculus is that sequents contain a special kind of structures, called *blocks*, which encode a finite combination of \preceq . We show that the calculus is terminating, whence it provides a decision procedure for the logic ∇ .

1 Introduction

In the recent history of conditional logics the work by Lewis [16] has a prominent place (among others [5, 18, 13, 11]), he proposed a formalization of conditional logics in order to represent a kind of hypothetical reasoning (if A were the case then B), that cannot be captured by classical logic with material implication. More precisely, the original motivation by Lewis was to formalize *counterfactual sentences*, i.e. conditionals of the form “if A were the case then B would be the case”, where A is false. But independently from counterfactual reasoning, conditional logics have found then an interest also in several fields of artificial intelligence and knowledge representation. Just to mention a few : they have been used to reason about prototypical properties [8] and to model belief change [11, 9]. Moreover, conditional logics can provide an axiomatic foundation of nonmonotonic reasoning [4, 12], here a conditional $A \Rightarrow B$ is read as “in normal circumstances if A then B ”. Finally, a kind of (multi)-conditional logics [2, 3] have been used to formalize epistemic change in a multi-agent setting and in

some kind of epistemic “games”, each conditional operator expresses the “conditional beliefs” of an agent.

In this paper we concentrate on the logic ∇ of counterfactual reasoning studied by Lewis. This logic is characterized by possible world models structured by a system of spheres. Intuitively, each world is equipped with a set of nested sets of worlds : inner sets represent “most plausible worlds” from the point of view of the given world and worlds belonging only to outer sets represent less plausible worlds. In other words, each sphere represent a degree of plausibility. The (rough) intuition involving the truth condition of a counterfactual $A \Rightarrow B$ at a world x is that B is true at the most plausible worlds where A is true, whenever there are worlds satisfying A . But Lewis is reluctant to assume that *most plausible worlds* A exist (whenever there are A -worlds), for philosophical reasons. He calls this assumption the Limit Assumption and he formulates his semantics in more general terms which do need this assumption (see below). The sphere semantics is the strongest semantics for conditional logics, in the sense that it characterizes only a subset of relatively strong systems ; there are weaker (and more abstract) semantics such as the selection function semantics which characterize a wider range of systems [18].

From the point of view of proof-theory and automated deduction, conditional logics do not have a state of the art comparable with, say, the one of modal logics, where there are well-established alternative calculi, whose proof-theoretical and computational properties are well-understood. This is partially due to the mentioned lack of a unifying semantics. Similarly to modal logics and other extensions/alternative to classical logics two types of calculi have been studied : *external* calculi which make use of labels and relations on them to import the semantics into the syntax, and *internal* calculi which stay within the language, so that a “configuration” (sequent, tableaux node...) can be directly interpreted as a formula of the language. Limiting our account to Lewis' counterfactual logics, some external calculi have been proposed in [10] which presents modular labeled calculi for preferential logic PCL and its extensions, this family includes all counterfactual logics by Lewis. Internal calculi have been proposed by Gent [7] and by de Swart [6] for Lewis' logic $\nabla\mathbb{C}$ and neighbours. These calculi manipulate sets of formulas and provide a decision procedure, although they comprise an *infinite set of rules and rules with a variable number of premises*. Finally in [15] the authors provide internal calculi for Lewis' conditional logic ∇ and some extensions. Their calculi are formulated for a language comprising the comparative plausibility connective, the strong and the weak conditional operator. Both conditional operators can be defined in terms of the *comparative similarity* connective. These calculi are actually an extension of Gent's and de Swart's ones and they comprise an infinite set of rules with a variable number of premises. We mention also a seminal work by Lamarre [13] who pro-

posed a tableaux calculus for Lewis' logic, but it is actually a model building procedure rather than a calculus made of deductive rules.

In this paper we tackle the problem of providing a standard proof-theory for Lewis' logic ∇ in the form of internal calculi. By “standard” we mean that we aim to obtain analytic sequent calculi where each connective is handled by a finite number of rules with a fixed and finite number of premises. As a preliminary result, we propose a new internal calculus for Lewis' logic ∇ . This is the most general logic of Lewis' family and it is complete with respect the whole class of sphere models (moreover, its unnested fragment essentially coincide with KLM rational logic \mathbf{R} [14]). Our calculus takes as primitive Lewis' comparative plausibility connective \preceq : a formula $A \preceq B$ means, intuitively, that A is at least as plausible as B , so that a conditional $A \Rightarrow B$ can be defined as A is impossible or $A \wedge \neg B$ is less plausible than A ¹. As a difference with previous attempts, our calculus comprises *structured* sequents containing *blocks*, where a block is a new syntactic structure encoding a finite combination of \preceq . In other words, we introduce a new modal operator (but still definable in the logic) which encodes finite combinations of \preceq . This is the main ingredient to obtain a standard and internal calculus for ∇ . We show that the calculus is terminating whence it provides a decision procedure. In further research we shall study its complexity and we shall study how to extend it to stronger logics of Lewis' family.

2 Lewis' logic ∇

We consider a propositional language \mathcal{L} generated from a set of propositional variables and boolean connectives plus two special connectives \preceq (comparative plausibility) and \Rightarrow (conditional). A formula $A \preceq B$ is read as “ A is at least as plausible as B ”. The semantics is defined in terms of sphere models, we take the definition by Lewis without the limit assumption.

Definition 1 A model \mathcal{M} has the form $\langle W, \$, [] \rangle$, where W is a non-empty set whose elements are called worlds, $[\] : Var_{prop} \rightarrow Pow(W)$ is the propositional evaluation, and $\$: W \rightarrow Pow(Pow(W))$. We write $\$ _x$ for the value of the function $\$$ for $x \in W$, and we denote the elements of $\$ _x$ by α, β, \dots . Models have the following property :

$$\forall \alpha, \beta \in \$ _x \quad \alpha \subseteq \beta \vee \beta \subseteq \alpha$$

The truth definition is the usual one for boolean cases, for the additional connectives we have :

1. This definition avoids the Limit Assumption, in the sense that it works also for models where *at least* a sphere containing A worlds does not necessarily exist.

- $x \in [A \preceq B]$ iff $\forall \alpha \in \mathcal{S}_x$ if $\alpha \cap [B] \neq \emptyset$ then $\alpha \cap [A] \neq \emptyset$
- $x \in [A \Rightarrow B]$ iff either $\forall \alpha \in \mathcal{S}_x$ $\alpha \cap [A] = \emptyset$ or there is $\alpha \in \mathcal{S}_x$, such that $\alpha \cap [A] \neq \emptyset$ and $\alpha \cap [A \wedge \neg B] = \emptyset$.

The semantic notions, satisfiability and validity are defined as usual.

For the ease of reading we introduce the following conventions and abbreviations : we write $x \models A$, where the model is understood instead of $x \in [A]$. Moreover given $\alpha \in \mathcal{S}_x$, we use the following notations :

- $\alpha \models^\forall A$ if $\alpha \subseteq [A]$, i.e. $\forall y \in \alpha$ $y \models A$
- $\alpha \models^\exists A$ if $\alpha \cap [A] \neq \emptyset$, i.e. $\exists y \in \alpha$ such that $y \models A$

Observe that with this notation, the truths conditions for \preceq and \Rightarrow become :

- $x \models A \preceq B$ iff $\forall \alpha \in \mathcal{S}_x$ either $\alpha \models^\forall \neg B$ or $\alpha \models^\exists A$
- $x \models A \Rightarrow B$ iff $\forall \alpha \in \mathcal{S}_x$ either $\alpha \models^\forall \neg A$ or there is $\beta \in \mathcal{S}_x$, such that $\beta \models^\exists A$ and $\beta \models^\forall A \rightarrow B$.

It can be observed that the two connectives \preceq and \Rightarrow are interdefinable, in particular :

$$A \Rightarrow B \equiv (\perp \preceq A) \vee \neg(A \wedge \neg B \preceq A)$$

Also the \preceq connective can be defined in terms of the conditional \Rightarrow as follows :

$$A \preceq B \equiv (A \vee B) \Rightarrow \perp \vee \neg((A \vee B) \Rightarrow \neg A)$$

A formula of the form $\perp \preceq \neg A$ means $\Box A$.

The logic ∇ can be axiomatized taking as primitive the conditional operator \Rightarrow which gives the axiomatization here below [16] :

- classical axioms and rules
- if $A \leftrightarrow B$ then $(C \Rightarrow A) \leftrightarrow (C \Rightarrow B)$ (RCEC)
- if $A \rightarrow B$ then $(C \Rightarrow A) \rightarrow (C \Rightarrow B)$ (RCK)
- $((A \Rightarrow B) \wedge (A \Rightarrow C)) \rightarrow (A \Rightarrow B \wedge C)$ (AND)
- $A \Rightarrow A$ (ID)
- $((A \Rightarrow B) \wedge (A \Rightarrow C)) \rightarrow (A \wedge B \Rightarrow C)$ (CM)
- $(A \wedge B \Rightarrow C) \rightarrow ((A \Rightarrow B) \rightarrow (A \Rightarrow C))$ (RT)²
- $((A \Rightarrow B) \wedge \neg(A \Rightarrow \neg C)) \rightarrow ((A \wedge C) \Rightarrow B)$ (CV)
- $((A \Rightarrow C) \wedge (B \Rightarrow C)) \rightarrow (A \vee B \Rightarrow C)$ (OR)

together with the definition of \preceq in terms of \Rightarrow given above. From the axiomatization above, it can be shown that the flat fragment (i.e. without nested conditionals) of ∇ corresponds to *rational* logic **R** introduced by Kraus, Lehmann and Magidor in [12].

On the other hand, we can axiomatize ∇ taking as primitive the connective \preceq and the axioms are the following :

2. It is worth noticing that (CM) + (RT) are equivalent (in CK+ID) to the axiom known as (CSO) :

$$((A \Rightarrow B) \wedge (B \Rightarrow A)) \rightarrow ((A \Rightarrow C) \leftrightarrow (B \Rightarrow C)) \quad (CSO)$$

- classical axioms and rules
- if $B \rightarrow (A_1 \vee \dots \vee A_n)$ then $(A_1 \preceq B) \vee \dots \vee (A_n \preceq B)$
- $(A \preceq B) \vee (B \preceq A)$
- $(A \preceq B) \wedge (B \preceq C) \rightarrow (A \preceq C)$
- $A \Rightarrow B \equiv (\perp \preceq A) \vee \neg(A \wedge \neg B \preceq A)$

The family of Lewis' systems contains stronger logics satisfying additional axioms like *centering*, *strong centering*, *uniformity* [16], corresponding to further properties of sphere systems.

3 An internal sequent calculus for ∇

In this section we present \mathcal{I}^∇ , a structured calculus for Lewis' conditional logic ∇ introduced in the previous section. In addition to ordinary formulas, sequents contains also *blocks* of the form :

$$[A_1, \dots, A_m \triangleleft B_1, \dots, B_n]$$

where each A_i, B_j are formulas. The interpretation is the following :

$$x \models [A_1, \dots, A_m \triangleleft B_1, \dots, B_n]$$

iff $\forall \alpha \in \mathcal{S}_x$:

- either $\alpha \models^\forall \neg B_j$ for some j , or
- $\alpha \models^\exists A_i$ for some i .

Observe that

$$[A_1, \dots, A_m \triangleleft B_1, \dots, B_n] \leftrightarrow \bigvee_i^m \bigvee_j^n (A_i \preceq B_j)$$

Therefore a block represents $n \times m$ disjunction of \preceq formulas.

We shall abbreviate multisets of formulas in blocks by Σ, Π , so that we shall write (since the order is irrelevant) :

$[\Sigma \triangleleft \Pi], [\Sigma, A \triangleleft \Pi], [\Sigma \triangleleft \Pi, B]$ and so on.

A *sequent* Γ is a multiset G_1, \dots, G_k , where each G_i is either a formula or a block. A sequent $\Gamma = G_1, \dots, G_k$, is valid if for every model $\mathcal{M} = \langle W, \mathcal{S}, [] \rangle$, for every world $x \in W$, it holds that $x \models G_1 \vee \dots \vee G_k$. The calculus \mathcal{I}^∇ comprises the following axiom and rules :

– Standard Axioms :

$$(i) \Gamma, \top \quad (ii) \Gamma, \neg \perp \quad (iii) \Gamma, P, \neg P$$

– Standard external rules of sequent calculi for boolean connectives

$$(\preceq^+)$$

$$\frac{\Gamma, [A \triangleleft B]}{\Gamma, A \preceq B} (\preceq^+)$$

– (\preceq^-)

$$\frac{\Gamma, \neg(A \preceq B), [B, \Sigma \triangleleft \Pi] \quad \Gamma, \neg(A \preceq B), [\Sigma \triangleleft \Pi, A]}{\Gamma, \neg(A \preceq B), [\Sigma \triangleleft \Pi]} (\preceq^-)$$

– (\Rightarrow^+)

$$\frac{\Gamma, [\perp \triangleleft A], \neg(A \wedge \neg B \preceq A)}{\Gamma, A \Rightarrow B} (\Rightarrow^+)$$

$$\text{— } (\Rightarrow^-) \quad \frac{\Gamma, \neg(\perp \preceq A) \quad \Gamma, [A \wedge \neg B \triangleleft A]}{\Gamma, \neg(A \Rightarrow B)} \quad (\Rightarrow^-)$$

$$\text{— (Communication)} \quad \frac{\Gamma, [\Sigma_1 \triangleleft \Pi_1, \Pi_2], [\Sigma_1, \Sigma_2 \triangleleft \Pi_2] \quad \Gamma, [\Sigma_2 \triangleleft \Pi_1, \Pi_2], [\Sigma_1, \Sigma_2 \triangleleft \Pi_1]}{\Gamma, [\Sigma_1 \triangleleft \Pi_1], [\Sigma_2 \triangleleft \Pi_2]} \quad (Com)$$

$$\text{— (Jump)} \quad \frac{\neg B_i, \Sigma}{\Gamma, [\Sigma \triangleleft B_1, \dots, B_n]} \quad (Jump)$$

Some remark on the rules : the rule $(\preceq+)$ just introduces the block structure, showing that \triangleleft is a generalization of \preceq ; $(\preceq-)$ prescribes case analysis and contribute to expand the blocks; the rules $(\Rightarrow+)$ and $(\Rightarrow-)$ just apply the definition of \Rightarrow in terms of \preceq . The (Communication) rule is directly motivated by the *nesting* of spheres, which means a *linear order* on sphere inclusion; this rule is very similar to the homonymous one used in *hypersequent* calculi for handling truth in linearly ordered structures [1, 17].

As usual, given a formula $G \in \mathcal{L}$, in order to check whether G is valid we look for a derivation of G in the calculus \mathcal{I}^V . Given a sequent Γ , we say that Γ is derivable in \mathcal{I}^V if it admits a *derivation*. A derivation of Γ is a tree where :

- the root is Γ ;
- a leaf is an instance of standard axioms;
- a non-leaf node is (an instance of) the conclusion of a rule having (an instance of) the premises of the rule as parents.

Here below we show a few examples of derivations.

Example 2 A derivation of $(A \preceq B) \vee (B \preceq A)$

$$\frac{\frac{\frac{\neg A, A}{[A \triangleleft B, A], [A, B \triangleleft A]} \quad (Jump) \quad \frac{\neg B, B}{[B \triangleleft B, A], [A, B \triangleleft B]} \quad (Jump)}{\frac{[A \triangleleft B], [B \triangleleft A]}{[A \triangleleft B, A], [A, B \triangleleft A]} \quad (Com)}{\frac{[A \triangleleft B], [B \triangleleft A]}{[A \triangleleft B], B \preceq A} \quad (\preceq+)}{\frac{[A \triangleleft B], B \preceq A}{A \preceq B, B \preceq A} \quad (\preceq+)}{\frac{A \preceq B, B \preceq A}{(A \preceq B) \vee (B \preceq A)} \quad (\vee+)}$$

We terminate this section by proving the soundness of the calculus and by stating some standard structural properties of the calculus.

Theorem 4 (Soundness) *Given a sequent Γ , if Γ is derivable then it is valid.*

Proof: by induction on the height of derivation. The basic case corresponds to proofs where Γ is an instance of standard axioms, is easy and left to the reader. For the inductive step, we have to consider all the rules ending a derivation. We only show the most interesting cases of the derivation ended by an application of $(\preceq-)$ and (Com) as follows :

$$\text{— } (\preceq^-)$$

$$\frac{(i) \Gamma', \neg(A \preceq B), [B, \Sigma \triangleleft \Pi] \quad (ii) \Gamma', \neg(A \preceq B), [\Sigma \triangleleft \Pi, A]}{\Gamma', \neg(A \preceq B), [\Sigma \triangleleft \Pi]} \quad (\preceq^-)$$

By inductive hypothesis, (i) and (ii) are valid sequents. By absurd, suppose that the conclusion is not, that is to say there is a model $\mathcal{M} = \langle W, \$, [] \rangle$ and a world $x \in W$ such that (1) $x \not\models G_i$, for all $G_i \in \Gamma'$, (2) $x \not\models \neg(A \preceq B)$ and (3) $x \not\models [\Sigma \triangleleft \Pi]$. From (1), (2) and the fact that (i) is valid, we conclude that (a) $x \models [B, \Sigma \triangleleft \Pi]$. Reasoning in the same way, from (1), (2) and the validity of (ii), we conclude that (b) $x \models [\Sigma \triangleleft \Pi, A]$. By the interpretation of a block, for all $\alpha \in \$_x$, from (a) we have that either $\alpha \models^\forall \neg B_j$ for some $B_j \in \Pi$ or $\alpha \models^\exists A_i$ for some $A_i \in \Sigma$ or (*) $\alpha \models^\exists B$. Similarly, from (b) we have that either $\alpha \models^\forall \neg B_j$ for some $B_j \in \Pi$ or (**) $\alpha \models^\forall \neg A$ or $\alpha \models^\exists A_i$ for some $A_i \in \Sigma$. If $\alpha \models^\forall \neg B_j$ for some $B_j \in \Pi$, then, by the interpretation of a block, we have that $x \models [\Sigma \triangleleft \Pi]$, and this contradicts (3). For the same reason, it cannot be also the case that $\alpha \models^\exists A_i$ for some $A_i \in \Sigma$. The only case left is when (*) $\alpha \models^\exists B$ and (**) $\alpha \models^\forall \neg A$. This contradicts (2). Indeed, (2) $x \not\models \neg(A \preceq B)$ means that $x \models A \preceq B$, namely, by the truth condition of \preceq , for all $\alpha \in \$_x$ we have that either $\alpha \models^\forall \neg B$, and this contradicts (*), or $\alpha \models^\exists A$, and this contradicts (**);

— (Com)

$$\frac{(i) \Gamma', [\Sigma_1 \triangleleft \Pi_1, \Pi_2], [\Sigma_1, \Sigma_2 \triangleleft \Pi_2] \quad (ii) \Gamma', [\Sigma_2 \triangleleft \Pi_1, \Pi_2], [\Sigma_1, \Sigma_2 \triangleleft \Pi_1]}{\Gamma', [\Sigma_1 \triangleleft \Pi_1], [\Sigma_2 \triangleleft \Pi_2]} \quad (Com)$$

By inductive hypothesis, (i) and (ii) are valid. Suppose the conclusion $\Gamma', [\Sigma_1 \triangleleft \Pi_1], [\Sigma_2 \triangleleft \Pi_2]$ is not, namely there is a model $\mathcal{M} = \langle W, \$, [] \rangle$ and a world $x \in W$ such that (1) $x \not\models G_k$ for all $G_k \in \Gamma'$, (2) $x \not\models [\Sigma_1 \triangleleft \Pi_1]$ and (3) $x \not\models [\Sigma_2 \triangleleft \Pi_2]$. By the interpretation of blocks, from (2) it follows that there is $\alpha \in \$_x$ such that $\alpha \not\models^\exists A_i$, for all $A_i \in \Sigma_1$, and $\alpha \not\models^\forall \neg B_j$ for all $B_j \in \Pi_1$. Similarly, from (3) it follows that there is $\beta \in \$_x$ such that $\beta \not\models^\exists C_k$, for all $C_k \in \Sigma_2$, and $\beta \not\models^\forall \neg D_l$ for all $D_l \in \Pi_2$. By Definition 1, either $\alpha \subseteq \beta$ or $\beta \subseteq \alpha$. If $\alpha \subseteq \beta$, we have also that $\beta \not\models^\exists A_i$, for all $A_i \in \Sigma_1$, and $\beta \not\models^\forall \neg B_j$ for all $B_j \in \Pi_1$. Let us consider (ii) : we have that $\beta \not\models^\exists C_k$, for all $C_k \in \Sigma_2$, as well as $\beta \not\models^\forall \neg B_j$ for all $B_j \in \Pi_1$ and $\beta \not\models^\forall \neg D_l$ for all $D_l \in \Pi_2$: by the definition of interpretation of a block, we have that (4) $x \not\models [\Sigma_2 \triangleleft \Pi_1, \Pi_2]$. Furthermore, since $\beta \not\models^\exists A_i$, for all $A_i \in \Sigma_1$, $\beta \in \$_x$ such that $\beta \not\models^\exists C_k$, for all $C_k \in \Sigma_2$ and $\beta \not\models^\forall \neg B_j$ for all $B_j \in \Pi_1$, then we have that (5) $x \not\models [\Sigma_1, \Sigma_2 \triangleleft \Pi_1]$. However, from (1),

Example 3 A derivation of an instance of Lewis' axiom **CV** $((P \Rightarrow R) \wedge \neg(P \Rightarrow \neg Q)) \rightarrow (P \wedge Q \Rightarrow R)$.

$$\begin{array}{c}
 \frac{\neg P, P, \perp}{\dots, [P, \perp \triangleleft P]} \text{ (Jump)} \quad \frac{\perp, \neg \perp}{\dots, [\perp \triangleleft P, \perp]} \text{ (Jump)} \\
 \hline
 \frac{(P \wedge Q) \Rightarrow R, \neg(\perp \preceq P), [\perp \triangleleft P], \neg(P \wedge \neg \neg Q \preceq P)}{(P \wedge Q) \Rightarrow R, P \Rightarrow \neg Q, \neg(\perp \preceq P)} \text{ (}\Rightarrow^+\text{)} \\
 \hline
 \frac{P \Rightarrow \neg Q, (P \wedge Q) \Rightarrow R, [P \wedge \neg R \triangleleft P]}{\neg(P \Rightarrow R), P \Rightarrow \neg Q, (P \wedge Q) \Rightarrow R} \text{ (}\Rightarrow^-\text{)} \\
 \hline
 \frac{\neg(P \Rightarrow R), \neg \neg(P \Rightarrow \neg Q), (P \wedge Q) \Rightarrow R}{\neg((P \Rightarrow R) \wedge \neg(P \Rightarrow \neg Q)), (P \wedge Q) \Rightarrow R} \text{ (}\wedge^-\text{)} \\
 \hline
 \frac{\neg((P \Rightarrow R) \wedge \neg(P \Rightarrow \neg Q)), (P \wedge Q) \Rightarrow R}{((P \Rightarrow R) \wedge \neg(P \Rightarrow \neg Q)) \rightarrow ((P \wedge Q) \Rightarrow R)} \text{ (}\rightarrow^+\text{)}
 \end{array}$$

where Δ is the following derivation :

$$\begin{array}{c}
 \frac{\neg P, \neg Q, P, P \wedge \neg R \quad \neg P, \neg Q, Q, P \wedge \neg R}{\neg P, \neg Q, P \wedge Q, P \wedge \neg R} \text{ (}\wedge^+\text{)} \quad \frac{P, \neg P, \neg Q, R \quad \neg R, \neg P, \neg Q, R}{P \wedge \neg R, \neg P, \neg Q, R} \text{ (}\wedge^+\text{)} \\
 \hline
 \frac{\neg P, \neg Q, P \wedge Q, P \wedge \neg R}{\neg P, \neg \neg(\neg Q), P \wedge Q, P \wedge \neg R} \text{ (}\neg\text{)} \quad \frac{P \wedge \neg R, \neg P, \neg Q, R}{P \wedge \neg R, \neg(P \wedge Q), R} \text{ (}\wedge^-\text{)} \\
 \hline
 \frac{\neg P, \neg \neg(\neg Q), P \wedge Q, P \wedge \neg R}{\neg(P \wedge \neg \neg Q), P \wedge Q, P \wedge \neg R} \text{ (}\wedge^-\text{)} \quad \frac{P \wedge \neg R, \neg(P \wedge Q), \neg \neg R}{P \wedge \neg R, \neg(P \wedge Q \wedge \neg R)} \text{ (}\wedge^-\text{)} \\
 \hline
 \frac{\dots, [P \wedge Q, P \wedge \neg R \triangleleft P, P \wedge \neg \neg Q]}{\dots, [P \wedge \neg R \triangleleft P, P \wedge \neg \neg Q, P \wedge Q \wedge \neg R]} \text{ (Jump)} \quad \frac{\dots, [P \wedge \neg R \triangleleft P, P \wedge \neg \neg Q, P \wedge Q \wedge \neg R]}{\dots, [P \wedge \neg R \triangleleft P, P \wedge \neg \neg Q, P \wedge Q \wedge \neg R]} \text{ (Jump)} \\
 \hline
 \frac{P, P \wedge \neg R, \neg P}{\dots, [P, P \wedge \neg R \triangleleft P]} \text{ (Jump)} \quad \frac{\dots, [P \wedge \neg R \triangleleft P, P \wedge \neg \neg Q, P \wedge Q \wedge \neg R]}{[\perp \triangleleft P], [\perp \triangleleft P \wedge Q], [P \wedge \neg R \triangleleft P, P \wedge \neg \neg Q], \neg(P \wedge Q \wedge \neg R \preceq P \wedge Q), \neg(P \wedge \neg \neg Q \preceq P)} \text{ (}\preceq^-\text{)} \\
 \hline
 \frac{[\perp \triangleleft P], [P \wedge \neg R \triangleleft P], [\perp \triangleleft P \wedge Q], \neg(P \wedge Q \wedge \neg R \preceq P \wedge Q), \neg(P \wedge \neg \neg Q \preceq P)}{[\perp \triangleleft P], \neg(P \wedge \neg \neg Q \preceq P), (P \wedge Q) \Rightarrow R, [P \wedge \neg R \triangleleft P]} \text{ (}\Rightarrow^+\text{)} \\
 \hline
 \frac{[\perp \triangleleft P], \neg(P \wedge \neg \neg Q \preceq P), (P \wedge Q) \Rightarrow R, [P \wedge \neg R \triangleleft P]}{P \Rightarrow \neg Q, (P \wedge Q) \Rightarrow R, [P \wedge \neg R \triangleleft P]} \text{ (}\Rightarrow^+\text{)}
 \end{array}$$

(4) and (5) we obtain that (ii) is not valid, against the inductive hypothesis ; If $\beta \subseteq \alpha$, the proof is symmetric and left to the reader. ■

Proposition 5 (Weakening) *Weakening is height-preserving admissible in the following cases :*

- if Γ is derivable, then Γ, F is derivable where F is a formula or a block.
- if $\Gamma, [\Sigma \triangleleft \Pi]$ is derivable, so are $\Gamma, [\Sigma, A \triangleleft \Pi]$ and $\Gamma, [\Sigma \triangleleft \Pi, B]$

Proposition 6 (Contraction) *Contraction is height-preserving admissible in the following cases :*

- if $\Gamma, [A, A, \Sigma \triangleleft \Pi]$ is derivable then $\Gamma, [A, \Sigma \triangleleft \Pi]$ is derivable too.
- if $\Gamma, [\Sigma \triangleleft \Pi, B, B]$ is derivable then $\Gamma, [\Sigma \triangleleft \Pi, B]$ is derivable too.
- if Γ, F, F is derivable then Γ, F is derivable too, where F is either a formula or a block.

Proposition 7 (Invertibility) *All rules, except (Jump), are height-preserving invertible : if the conclusion is derivable then the premises must be derivable with a derivation of no-greater height.*

4 Termination and Completeness

In this section we prove both the termination and the completeness of the calculus. Both results make use of the notion of *saturated sequent* : intuitively any sequent that is obtained by backwards applying the rules “as much as possible”. To get termination we show that any derivation without *redundant* application of the rules is finite and its leaves are axioms or saturated sequents. Completeness is proved by induction on the modal degree of a sequent (defined next), by taking advantage of the fact that backward application of the rules does not increase the modal degree of sequent and eventually reduced it (the Jump rule).

Definition 8 *The modal degree md of a formula/sequent is defined as follows :*

$$\begin{aligned}
 md(P) &= 0 \\
 md(A * B) &= \max(md(A), md(B)), \text{ for } * \text{ boolean connective} \\
 md(\neg A) &= md(A) \\
 md(A \preceq B) &= md(A \Rightarrow B) = \max(md(A), md(B)) + 1 \\
 md(\Delta) &= \max\{md(A) \mid A \in \Delta\} \text{ for a multiset}
 \end{aligned}$$

$$\Delta$$

$$md(\Sigma \triangleleft \Pi) = \max(md(\Sigma), md(\Pi)) + 1$$

Definition 9 A sequent Γ is saturated if it has the form

$$\Gamma_N, \Lambda, [\Sigma_1 \triangleleft \Pi_1], \dots, [\Sigma_n \triangleleft \Pi_n]$$

where Γ_N, Λ are possible empty and $n \geq 0$ and :

1. Γ_N is a multi-set of negative \preceq -formulas,
2. Λ is a multi-set of literals,
3. for every $\neg(A \preceq B) \in \Gamma_N$ and every $[\Sigma_i \triangleleft \Pi_i]$ either $B \in \Sigma_i$ or $A \in \Pi_i$
4. for every $[\Sigma_i \triangleleft \Pi_i]$ and $[\Sigma_j \triangleleft \Pi_j]$: either $\Sigma_i \subseteq \Sigma_j$ or $\Sigma_j \subseteq \Sigma_i$ and either $\Pi_i \subseteq \Pi_j$ or $\Pi_j \subseteq \Pi_i$.

Proposition 10 All rules preserve the modal degree, i.e. the premises of rules have a modal degree no-greater than the one of the respective conclusion.

We want to prove now that the calculus terminates, provided we restrict attention to *non-redundant* derivations, a notion that we define next. An application of a rule (R) :

$$\frac{\Gamma_1 \quad \Gamma_2}{\Gamma} (R)$$

is redundant if Γ can be obtained from Γ_i for $i = 1$ or $i = 2$ by *contraction* or *weakening*. A derivation is non-redundant if (a) it does not contain redundant applications of the rules, (b) if a sequent is an axiom then it is a leaf of the derivation. As a consequence of the admissibility of contraction (Proposition 6) and of weakening (Proposition 5), if a sequent is derivable then it has a non-redundant derivation. Thus we can safely restrict proof search to non-redundant derivations.

The proposition below means that for any sequent Γ (derivable or not in the calculus), there is a (non-redundant) derivation tree whose leaves (no matter whether they are derivable or not in the calculus) are saturated sequents with no greater modal degree. In order to prove it, we introduce some complexity measure of sequents. First we define a complexity measure of formulas :

$$Cp(A) = 0 \text{ if } A \text{ is either a literal or it has the form } \neg(C \preceq D),$$

$$Cp(A) = 1 \text{ if } A \text{ has one of the forms } C \preceq D, C \Rightarrow D, \neg(C \Rightarrow D)$$

$$Cp(\neg\neg A) = Cp(A) + 1$$

$$Cp(A * B) = Cp(A) + Cp(B) + 1 \text{ where } * \text{ is a boolean connective.}$$

Next we let

$$CP(\Gamma) = \text{multiset}\{Cp(A) \mid A \in \Gamma\}$$

We further define :

$$CN(\Gamma) = \text{Card}(\{\neg(A \preceq B), [\Sigma \triangleleft \Pi] \mid \neg(A \preceq B), [\Sigma \triangleleft \Pi] \in \Gamma, B \notin \Sigma, A \notin \Pi\})$$

$$CC(\Gamma) = n * (\text{Card}(\Sigma_\Gamma) + \text{Card}(\Pi_\Gamma)) - \sum_{i=1}^n (\text{Card}(\Sigma_i) + \text{Card}(\Pi_i))$$

We finally define the rank of a sequent Γ , $rank(\Gamma)$ as the triple

$$rank(\Gamma) = \langle CP(\Gamma), CN(\Gamma), CC(\Gamma) \rangle$$

taken in *lexicographic* order, where we consider the *multiset ordering* for $CP(\Gamma)$. Observe that a *minimal* rank has the form $\langle 0^*, 0, m \rangle$, where $m \geq 0$.

We are ready to prove the following proposition.

Proposition 11 Given a sequent Γ , every branch of any derivation-tree starting with Γ eventually ends with a saturated sequent with no greater modal degree than that of Γ . Moreover the set of such saturated sequents for a given derivation tree is finite.

Proof: by Proposition 10, no rule applied backward augments the modal degree of a sequent. It can be shown that every (non-redundant) application of a rules (R) with premises Γ_i and conclusion Γ reduces the rank of Γ in the sense that $rank(\Gamma_i) < rank(\Gamma)$. In order to see this, we note :

- the application of classical propositional rule reduces $CP(\Gamma)$
- the application of $(\preceq^+), (\Rightarrow^+), (\Rightarrow^-)$ rules reduces $CP(\Gamma)$
- the application of (\preceq^-) reduces $CN(\Gamma)$, *without increasing* $CP(\Gamma)$
- the application of (Com) reduces $CC(\Gamma)$, *without increasing* neither $CP(\Gamma)$, nor $CN(\Gamma)$. We first show that an application of (Com) rule reduces $CC(\Gamma)$. Let $\Gamma = \Delta, [\Sigma_1 \triangleleft \Pi_1], [\Sigma_2 \triangleleft \Pi_2], \dots, [\Sigma_n \triangleleft \Pi_n]$. To simplify indexing (since the order does not matter) suppose that the application of (Com) concerns the blocks $[\Sigma_1 \triangleleft \Pi_1], [\Sigma_2 \triangleleft \Pi_2]$, so that from the premises of the application of (Com) leading to Γ will be :

$$\Gamma_1 = \Delta, [\Sigma_1 \triangleleft \Pi_1, \Pi_2], [\Sigma_1, \Sigma_2 \triangleleft \Pi_2], [\Sigma_3 \triangleleft \Pi_3], \dots, [\Sigma_n \triangleleft \Pi_n]$$

$$\Gamma_2 = \Delta, [\Sigma_2 \triangleleft \Pi_1, \Pi_2], [\Sigma_1, \Sigma_2 \triangleleft \Pi_1], [\Sigma_3 \triangleleft \Pi_3], \dots, [\Sigma_n \triangleleft \Pi_n]$$

Observe that the overall set of formulas in blocks does not change so that, referring to the above notation :

$$\Sigma_{\Gamma_i} = \Sigma_\Gamma \text{ and } \Pi_{\Gamma_i} = \Pi_\Gamma, \quad \text{for } i = 1, 2$$

Let us abbreviate $a = n * (\text{Card}(\Sigma_\Gamma) + \text{Card}(\Pi_\Gamma))$ and $c = \sum_{i=3}^n (\text{Card}(\Sigma_i) + \text{Card}(\Pi_i))$, so that we have :

$$CC(\Gamma) = a - ((\text{Card}(\Sigma_1) + \text{Card}(\Pi_1)) + (\text{Card}(\Sigma_2) + \text{Card}(\Pi_2))) + c$$

$$CC(\Gamma_1) = a - ((Card(\Sigma_1) + Card(\Pi_1 \cup \Pi_2)) + (Card(\Sigma_1 \cup \Sigma_2) + Card(\Pi_2)) + c)$$

$$CC(\Gamma_2) = a - ((Card(\Sigma_2) + Card(\Pi_1 \cup \Pi_2)) + (Card(\Sigma_1 \cup \Sigma_2) + Card(\Pi_1)) + c)$$

Obviously $CC(\Gamma_1) \leq CC(\Gamma)$ and $CC(\Gamma_2) \leq CC(\Gamma)$, since $Card(\Sigma_1 \cup \Sigma_2) \geq Card(\Sigma_i)$ and $Card(\Pi_1 \cup \Pi_2) \geq Card(\Pi_i)$, $i = 1, 2$. But since the application of (Com) is non-redundant, it respects the restriction $(RestCom)$ and therefore either (a) $\Sigma_1 \not\subseteq \Sigma_2$ and $\Sigma_2 \not\subseteq \Sigma_1$ or (b) $\Pi_1 \not\subseteq \Pi_2$ and $\Pi_2 \not\subseteq \Pi_1$. It is easy to see that some inequalities must be strict by the $(RestCom)$ restriction, so that in both cases (a) and (b) we get $CC(\Gamma_1) < CC(\Gamma)$ and $CC(\Gamma_2) < CC(\Gamma)$. ■

The following theorem shows that the calculus is terminating, whence it provides a decision procedure for ∇ , assuming restriction to non-redundant derivations.

Proposition 12 *Given a sequent Γ , any non-redundant derivation-tree of Γ is finite.*

Proof: by induction on the modal degree m of Γ . If $m = 0$ then we rely on the corresponding property of classical sequent calculus. If $m > 0$, by the previous Proposition 11, Γ has a finite derivation tree ending with a set of saturated sequents Γ_i . For each Γ_i either it is an axiom and Γ_i will be a leaf of the derivation, or the only applicable rule (by non-redundancy restriction) is $(Jump)$, but the premise of $(Jump)$ has a smaller modal degree and we apply the induction hypothesis to the premise of $(Jump)$. ■

The following proposition is the last ingredient we need for the completeness proof.

Proposition 13 (Semantic Invertibility) *All rules, except $(Jump)$ are semantically invertible : if the conclusion is valid then the premises are also valid.*

Theorem 14 (Completeness of the calculus) *If Γ is valid then it is derivable*

Proof: by induction on the modal degree of Γ . If $md(\Gamma) = 0$ then Γ is just a multiset of propositional formulas, and we rely on the completeness of sequent calculus for classical logic.

Suppose now that $md(\Gamma) > 0$, by Proposition 11, Γ can be derived from a set of saturated sequents Γ_i of no greater modal degree. But by previous Proposition 13 (semantic invertibility) since Γ is valid then also each Γ_i is valid. We

are left to prove that any *saturated and valid* sequent Γ_i is derivable.

To this purpose we prove that if Γ_i is valid then either (i) it is an axiom or (ii) there must exist a *valid* sequent Δ such that Γ_i is obtained by $(Jump)$ from Δ . In the first case (i) the result is obvious. In case (ii) we reason as follows : since $md(\Delta) < md(\Gamma_i)$ by induction hypothesis Δ is derivable in the calculus, and so is Γ_i indeed by the $(Jump)$ rule.

Let us prove the fact (ii) : if Γ_i is valid and saturated and it is not an axiom, then there exist a *valid* sequent Δ such that Γ_i is obtained by $(Jump)$ from Δ .

Suppose that Γ_i is valid and it is not an axiom. We let

$$\Gamma_i = \Gamma_N, \Lambda, [\Sigma_1 \triangleleft \Pi_1], \dots, [\Sigma_n \triangleleft \Pi_n]$$

as in the definition of saturated sequent. Observe that Λ does not contain axioms. By saturation (and weakening and contraction) we can assume that the blocks in the sequence are ordered as follows :

$$\begin{aligned} - \Sigma_1 \supseteq \Sigma_2 \supseteq \dots \supseteq \Sigma_n \\ - \Pi_1 \subseteq \Pi_2 \subseteq \dots \subseteq \Pi_n \end{aligned}$$

A quick argument : by saturation blocks are ordered with respect to set-inclusion for both components Σ and Π , consider them ordered first by decreasing Σ : let two blocks in the sequence : $[\Sigma \triangleleft \Pi]$, $[\Sigma' \triangleleft \Pi']$ with $\Sigma' \subseteq \Sigma$ we can assume that $\Pi \subseteq \Pi'$ otherwise it would be $\Pi' \subseteq \Pi$, but then any sequent containing **both** $[\Sigma \triangleleft \Pi]$, $[\Sigma' \triangleleft \Pi']$ is semantically equivalent to a sequent containing **only** $[\Sigma \triangleleft \Pi]$ (syntactically we get rid of $[\Sigma' \triangleleft \Pi']$ by weakening and contraction)³. Thus we let :

$$\begin{aligned} \Pi_1 &= B_{1,1}, \dots, B_{1,k_1} \\ \Pi_2 &= B_{1,1}, \dots, B_{1,k_1}, B_{2,1}, \dots, B_{2,k_2} \\ &\dots \\ \Pi_n &= B_{1,1}, \dots, B_{1,k_1}, \dots, B_{2,k_2}, \dots, B_{n,k_n} \end{aligned}$$

Suppose now by absurdity that no application of $(Jump)$ leads to a valid sequent. Thus for each $l = 1, \dots, n$, and $t = 1, \dots, k_l$, the sequent

$$\neg B_{l,t}, \Sigma_l$$

is not valid. Starting from $l = 1$ up to n , there are increasing sequences of models :

$$\begin{aligned} \mathcal{M}_{1,1}, \dots, \mathcal{M}_{1,k_1}, \\ \mathcal{M}_{1,1}, \dots, \mathcal{M}_{1,k_1}, \mathcal{M}_{2,1}, \dots, \mathcal{M}_{2,k_2} \\ \mathcal{M}_{1,1}, \dots, \mathcal{M}_{1,k_1}, \dots, \mathcal{M}_{2,k_2}, \dots, \mathcal{M}_{n,k_n} \end{aligned}$$

where $\mathcal{M}_{l,t} = (W_{l,t}, \mathcal{S}^{l,t}, []_{l,t})$ for $l = 1, \dots, n$, and $t = 1, \dots, k_l$ and some elements $x_{l,t} \in W_{l,t}$ such that

$$\mathcal{M}_{l,t}, x_{l,t} \models B_{l,t}$$

and

$$\mathcal{M}_{l,t}, x_{l,t} \not\models C \text{ for all } C \in \Sigma_l$$

3. An alternative argument : Γ_i must contain a *valid* subsequent Γ'_i where the blocks satisfy the above ordering conditions. Then the proof carry on considering Γ'_i .

Observe that if $\mathcal{M}_{l,t}, x_{l,t} \not\models C$ for all $C \in \Sigma_s$ and $s < t$ then $\mathcal{M}_{l,t}, x_{l,t} \not\models C$ for all $C \in \Sigma_t$, as $\Sigma_t \subseteq \Sigma_s$.

We suppose that all models are disjoint and we define a new model $\mathcal{M} = \langle \mathcal{W}, \$, [] \rangle$ as follows :

$$\begin{aligned} \mathcal{W} &= (\bigcup_l \bigcup_t (\mathcal{W}_{l,t})) \cup \{x\} \text{ for a new element } x \\ \$_z &= \$_{z,t}^{l,t} \text{ if } z \in \mathcal{W}_{l,t} \text{ for some } l, t \\ [P] &= \bigcup_l \bigcup_t [P]_{l,t} \text{ if } \neg P \notin \Lambda \\ [P] &= \bigcup_l \bigcup_t [P]_{l,t} \cup \{x\} \text{ if } \neg P \in \Lambda \end{aligned}$$

In order to define $I(x)$ we let :

$$\begin{aligned} \alpha_1 &= \{x_{1,1}, \dots, x_{1,k_1}\} \\ \alpha_2 &= \{x_{1,1}, \dots, x_{1,k_1}, x_{2,1}, \dots, x_{2,k_2}\} \\ &\dots \\ \alpha_n &= \{x_{1,1}, \dots, x_{1,k_1}, x_{2,1}, \dots, x_{2,k_2}, \dots, x_{n,k_n}\} \end{aligned}$$

We finally let

$$\$_x = \{\alpha_1, \dots, \alpha_n\}$$

Observe that the ‘‘spheres’’ α_l are nested.

To complete the proof we must show that x falsifies Γ_i in \mathcal{M} . In particular we have to show that :

1. $\mathcal{M}, x \not\models L$ for every $L \in \Lambda$
2. $\mathcal{M}, x \not\models [\Sigma_l \triangleleft \Pi_l]$ for $l = 1, \dots, n$
3. $\mathcal{M}, x \not\models \neg(A \preceq B)$ for every $\neg(A \preceq B) \in \Gamma_N$

Fact (1) is obvious by definition : if $P \in \Lambda$, then $\neg P \notin \Lambda$ (otherwise Γ_i would be an axiom) and $x \notin [P]$, if $\neg P \in \Lambda$, then $x \in [P]$.

To prove Fact (2), first observe that for $z \in \mathcal{W}_{l,t}$ and every formula F , we have

$$z \in [F] \iff z \in [F]_{l,t}$$

This is proved by a straightforward induction on F . Then we prove by induction on l Fact (2). For $l = 1$, we have that for $x_{1,l} \in \alpha_1$

$$\mathcal{M}, x_{1,l} \models B_{1,l}$$

whence :

$$\alpha_1 \not\models^\nabla \neg B_{1,t} \text{ for } t = 1, \dots, k_1.$$

On the other hand putting $\Sigma_1 = C_{1,1}, \dots, C_{1,r_1}$, we have, for every $u = 1, \dots, r_1$ and $x_{1,t}, t = 1, \dots, k_1$

$$\mathcal{M}, x_{1,t} \not\models C_{1,u}$$

but this means that

$$\alpha_1 \not\models^\exists C_{1,u} \text{ for } u = 1, \dots, r_1.$$

Thus we get

$$\mathcal{M}, x \not\models [\Sigma_1 \triangleleft \Pi_1]$$

For $l > 1$, since $\Sigma_l \supseteq \Sigma_{l-1}$ and $\Pi_{l-1} \subseteq \Pi_l$, the argument is the same (using possibly the induction hypothesis).

We consider now (Fact 3) : let $\neg(A \preceq B) \in \Gamma_N$ and let $\alpha_l \in \$_x$. Let us consider $[\Sigma_l \triangleleft \Pi_l]$, by saturation either $A \in \Pi_l$ or $B \in \Sigma_l$. For what we have just shown, in the former case we have $\alpha_l \models^\exists A$ and in the latter case we have $\alpha_l \models^\nabla \neg B$. Thus :

for any $\alpha_l \in \$_x$ either $\alpha_l \models^\exists A$ or $\alpha_l \models^\nabla \neg B$.

whence $\mathcal{M}, x \models A \preceq B$. ■

5 Conclusions

In this paper we begin a proof-theoretical investigation of Lewis' logics of counterfactuals characterized by the sphere-model semantics. We have presented a simple, analytic calculus \mathcal{I}^∇ for logic ∇ , the most general logic characterized by the sphere-model semantics. The calculus is *standard*, that is to say it contains a finite a number of rules with a fixed number of premisses and *internal* in the sense that each sequent denotes a formula of ∇ . The novel ingredient of \mathcal{I}^∇ is that sequents are structured objects containing blocks, where a block is a structure or a sort of n-ary modality encoding a finite combination of formulas with the connective \preceq . The calculus \mathcal{I}^∇ ensures termination, and therefore it provides a decision procedure for ∇ .

In future research, we aim at extending our approach to all the other conditional logics of the Lewis' family, in particular we aim at focusing on the logics $\nabla\mathbb{T}$, $\nabla\mathbb{W}$ and $\nabla\mathbb{C}$. Moreover, we shall study the complexity of the calculus \mathcal{I}^∇ with the hope of obtaining optimal calculi.

Références

- [1] A. Avron. The method of hypersequents in the proof theory of propositional non-classical logics. In Wilfrid Hodges, Martin Hyland, Charles Steinhorn, and John Truss, editors, *Logic : from foundations to applications.*, pages 1–32. Oxford University Press, New York, 1996.
- [2] Alexandru Baltag and Sonja Smets. The logic of conditional doxastic actions. *Texts in Logic and Games, Special Issue on New Perspectives on Games and Interaction*, 4 :9–31, 2008.
- [3] Oliver Board. Dynamic interactive epistemology. *Games and Economic Behavior*, 49(1) :49–80, 2004.
- [4] C. Boutilier. Conditional logics of normality : a modal approach. *Artificial Intelligence*, 68(1) :87–154, 1994.
- [5] B. F. Chellas. Basic conditional logics. *Journal of Philosophical Logic*, 4 :133–153, 1975.
- [6] H. C. M. de Swart. A gentzen- or beth-type system, a practical decision procedure and a constructive completeness proof for the counterfactual logics vc and vcs . *Journal of Symbolic Logic*, 48(1) :1–20, 1983.
- [7] I. P. Gent. A sequent or tableaux-style system for lewis's counterfactual logic vc . *Notre Dame Journal of Formal Logic*, 33(3) :369–382, 1992.

- [8] M. L. Ginsberg. Counterfactuals. *Artificial Intelligence*, 30(1) :35–79, 1986.
- [9] Laura Giordano, Valentina Gliozzi, and Nicola Olivetti. Weak AGM postulates and strong ramsey test : A logical formalization. *Artificial Intelligence*, 168(1-2) :1–37, 2005.
- [10] Laura Giordano, Valentina Gliozzi, Nicola Olivetti, and Camilla Schwind. Tableau calculus for preference-based conditional logics : Pcl and its extensions. *ACM Trans. Comput. Logic*, 10(3), 2009.
- [11] G. Grahne. Updates and counterfactuals. *Journal of Logic and Computation*, 8(1) :87–117, 1998.
- [12] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence*, 44(1-2) :167–207, 1990.
- [13] P. Lamarre. Etude des raisonnements non-monotones : Apports des logiques des conditionnels et des logiques modales. *PhD thesis, Université Paul Sabatier, Toulouse*, 1992.
- [14] D. Lehmann and M. Magidor. What does a conditional knowledge base entail ? *Artificial Intelligence*, 55(1) :1–60, 1992.
- [15] Bjoern Lellmann and Dirk Pattinson. Sequent Systems for Lewis' Conditional Logics. In Jérôme Mengin Luis Fariñas del Cerro, Andreas Herzig, editor, *Logics in Artificial Intelligence - 13th European Conference, JELIA 2012*, volume 7519 of *Lecture Notes in Artificial Intelligence (LNAI)*, page to appear, Toulouse, France, September 2012. Springer-Verlag.
- [16] D. Lewis. Counterfactuals. *Basil Blackwell Ltd*, 1973.
- [17] George Metcalfe, Nicola Olivetti, and Dov Gabbay. *Proof Theory for Fuzzy Logics*. Springer, 2010.
- [18] D. Nute. Topics in conditional logic. *Reidel, Dordrecht*, 1980.

Generalized Qualitative Spatio-Temporal Reasoning: Complexity and Tableau Method

Michael Sioutis Jean-François Condotta Yakoub Salhi Bertrand Mazure

Université Lille-Nord de France, Artois, CRIL-CNRS UMR 8188, Lens, France
 {sioutis,condotta,salhi,mazure}@cril.fr

Résumé

Dans cet article, nous traitons de la logique spatio-temporelle qui résulte de combinaison de la logique temporelle propositionnelle (PTL) avec le langage de contraintes spatiales qualitatives \mathcal{L}_1 . Nous proposons une méthode de résolution à base de tableaux sémantiques basée sur la méthode des tableaux de Wolper pour PTL. Par ailleurs, nous étudions les effets de l'utilisation des propriétés de *compacité* et de *patchwork* pour le raisonnement spatio-temporelle. Nous étudions notamment l'effet de ces propriétés sur la complexité du problème de satisfiabilité de \mathcal{L}_1 en remplaçant notamment la propriété de cohérence globale utilisée habituellement dans la littérature. Ceci permet la généralisation à un plus grand nombre de langages de contraintes spatiales qualitatives. Enfin, les résultats obtenus permettent de prouver la validité de notre méthode des tableaux pour \mathcal{L}_1 .

Abstract

We study the spatiotemporal logic that results by combining the propositional temporal logic (PTL) with a qualitative spatial constraint language, namely, the \mathcal{L}_1 logic, and present a first semantic tableau method that given a \mathcal{L}_1 formula ϕ systematically searches for a model for ϕ . Our approach builds on Wolper's tableau method for PTL, while the ideas provided can be carried to other tableau methods for PTL as well. Further, we investigate the implication of the constraint properties of *compactness* and *patchwork* in spatiotemporal reasoning. We use these properties to strengthen results regarding the complexity of the satisfiability problem in \mathcal{L}_1 , by replacing the stricter global consistency property used in literature and generalizing to more qualitative spatial constraint languages. Finally, the obtained strengthened results allow us to prove the correctness of our tableau method for \mathcal{L}_1 .

1 Introduction

Time and space are fundamental cognitive concepts that have been the focus of study in many scientific disciplines, including Artificial Intelligence and, in particular, Knowledge Representation. Knowledge Representation has been quite successful in dealing with the concepts of time and space, and has developed formalisms that range from temporal and spatial databases [18], to quantitative models developed in computational geometry [14] and qualitative constraint languages and logical theories developed in qualitative reasoning [7, 21].

Towards constraint-based qualitative spatiotemporal reasoning, most of the work has relied on formalisms based on the propositional temporal logic (PTL), also known as linear temporal logic, and the qualitative spatial constraint language RCC-8 [21, 20]. PTL [9] is the well known temporal logic comprising operators \mathcal{U} (until), \bigcirc (next point in time), \square (always), and \diamond (eventually) over various flows in time, such as $\langle \mathbb{N}, < \rangle$. RCC-8 is a fragment of the Region Connection Calculus (RCC) [15] and is used to describe regions that are non-empty regular subsets of some topological space by stating their topological relations to each other. The topological relations comprise relations *DC* (disconnected), *EC* (externally connected), *EQ* (equal), *PO* (partially overlapping), *TPP* (tangential proper part), *TPPi* (tangential proper part inverse), *NTPP* (non-tangential proper part), *NTPPi* (non-tangential proper part inverse). These 8 relations are depicted in [15, Fig. 4]. One of the most important of such formalisms is the ST_1^- logic [5]. For example, one can have the following statement using that formalism: $\diamond TPP(X, Y)$, which translates to “eventually region X will be a tangential proper part of region Y ”.

In this paper, we consider a generalization of the

\mathcal{ST}_1^- logic, denoted by \mathcal{L}_1 , which is the product of the combination of PTL [9] with any qualitative spatial constraint language, such as RCC-8 [15], Cardinal Direction Algebra (CDA) [4, 11], and Block Algebra (BA) [6], and make the following contributions: (i) we show that satisfiability checking of a \mathcal{L}_1 formula is PSPACE-complete if the qualitative spatial constraint language considered has the constraint properties of *compactness* and *patchwork* [12] for atomic networks, thus, strengthening previous related results that required atomic networks to be *globally consistent* [2, 3], and (ii) we present a first semantic tableau method that given a \mathcal{L}_1 formula ϕ systematically searches for a model for ϕ . This method builds on the tableau method for PTL of Wolper [19], and makes use of our strengthened results to ensure soundness and completeness. It is important to note, that Wolper's method serves as the basis to illustrate our line of reasoning, and that the techniques presented can be carried to other more efficient tableau methods for PTL as well.

As opposed to the \mathcal{ST}_1^- logic [5], \mathcal{L}_1 does not rely on the semantics or a particular interpretation of the qualitative spatial constraint language used, but rather on constraint properties, namely, *compactness* and *patchwork* [12]. These properties have been found to hold for RCC-8, Cardinal Direction Algebra (CDA), Block Algebra (BA), and their derivatives [8].

The organization of the paper is as follows. In Section 2 we recall the definition of a qualitative spatial constraint language, along with the properties of compactness, patchwork, and global consistency. Section 3 introduces the \mathcal{L}_1 logic, and in Section 4 we explain its implication with compactness and patchwork. In Section 5 we present our tableau method for checking the satisfiability of a \mathcal{L}_1 formula. In Section 6 we conclude and give directions for future work.

2 Preliminaries

A (binary) qualitative temporal or spatial constraint language [17] is based on a finite set \mathbf{B} of *jointly exhaustive and pairwise disjoint* (JEPD) relations defined on a domain \mathbf{D} , called the set of base relations. The base relations of set \mathbf{B} of a particular qualitative constraint language can be used to represent the definite knowledge between any two entities with respect to the given level of granularity. \mathbf{B} contains the identity relation Id , and is closed under the inverse operation ($^{-1}$). Indefinite knowledge can be specified by disjunctions of possible base relations, and is represented by the set containing them. Hence, $2^{\mathbf{B}}$ represents the total set of relations. $2^{\mathbf{B}}$ is equipped with the usual set-theoretic operations (union and intersection), the

inverse operation, and the weak composition operation denoted by \diamond [17]. A network from any qualitative spatial constraint language, such as RCC-8 [15], Cardinal Direction Algebra (CDA) [4, 11], or Block Algebra (BA) [6], can be formulated as a qualitative constraint network (QCN) as follows (a RCC-8 example of which is shown in Figure 1).

Definition 1 A QCN is a tuple (V, C) where V is a non-empty finite set of variables and C is a mapping that associates a relation $C(v, v') \in 2^{\mathbf{B}}$ to each pair (v, v') of $V \times V$. Mapping C is such that $C(v, v) = \{\text{Id}\}$ and $C(v, v') = (C(v', v))^{-1}$ for every $v, v' \in V$.

If b is a base relation, $\{b\}$ is a singleton relation. An *atomic* QCN is a QCN where each constraint is a singleton relation. Note that we always regard a QCN as a complete network. Given two QCNs $\mathcal{N} = (V, C)$ and $\mathcal{N}' = (V', C')$, $\mathcal{N} \cup \mathcal{N}'$ denotes the QCN $\mathcal{N}'' = (V'', C'')$, where $V'' = V \cup V'$, $C''(u, v) = C''(v, u) = \mathbf{B}$ for all $(u, v) \in (V \setminus V') \times (V' \setminus V)$, $C''(u, v) = C(u, v) \cap C'(u, v)$ for every $u, v \in V \cap V'$, $C''(u, v) = C(u, v)$ for every $u, v \in V \setminus V'$, and $C''(u, v) = C'(u, v)$ for every $u, v \in V' \setminus V$. Given a QCN $\mathcal{N} = (V, C)$ and $u, v \in V$, $C(u, v)$ will be also denoted by $\mathcal{N}[u, v]$.

We can interpret any QCN $\mathcal{N} = (V, C)$ using a structure of the form $\mathcal{M}_{\mathcal{S}} = (\mathbf{D}, \alpha)$, where α is a mapping that associates elements of \mathbf{D} to elements of V . For the case of RCC-8 for example, if \mathcal{T} is some topological space [13], let $\mathcal{R}(\mathcal{T})$ denote the set of all non-empty regular closed subsets in \mathcal{T} . Then, the domain \mathbf{D} of RCC-8 is the set $\mathcal{R}(\mathcal{T})$, which can be infinite. A structure $\mathcal{M}_{\mathcal{S}} = (\mathbf{D}, \alpha)$ is a model for a QCN $\mathcal{N} = (V, C)$, also called a *solution*, if mapping α can yield a spatial configuration where the relations between the spatial variables can be described by C . It follows that a QCN is satisfiable if there exists a model for it. A *partial solution* for \mathcal{N} on $V' \subseteq V$ is the mapping α restricted to V' .

Checking the satisfiability of a RCC-8, CDA, or BA network is \mathcal{NP} -complete in the general case [16, 11, 6]. However, there exist large maximal tractable subclasses of RCC-8, CDA, and BA, which allow for practical and efficient reasoning. In particular, checking the satisfiability of a QCN (V, C) of RCC-8, CDA, or BA comprising only relations from one of its maximal tractable subclasses containing all singleton relations and the universal relation \mathbf{B} , can be done in $O(|V|^3)$ time using the \diamond -consistency algorithm (also called *algebraic closure*), that iteratively performs the following operation until a fixed point \bar{C} is reached: $\forall v, v', v'' \in V, C(v, v') \leftarrow C(v, v') \cap (C(v, v'') \diamond C(v'', v'))$ [17].

Let us recall the definition of global consistency.

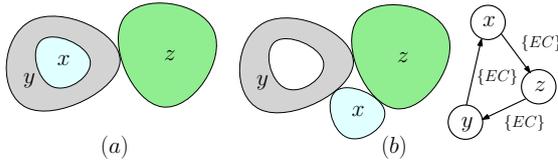


Figure 1: RCC-8 configurations

Definition 2 A QCN $\mathcal{N} = (V, C)$ is globally consistent if and only if, for any $V' \subset V$, every partial solution on V' can be extended to a partial solution on $V' \cup \{v\} \subseteq V$, for any $v \in V \setminus V'$.

We now recall the definitions of the constraint properties of *patchwork* and *compactness* in the context of qualitative reasoning and give an example of how the former properties combined are less strict than global consistency alone. (To be precise, [12] introduced patchwork for atomic QCNs, and [8] generalized it also for non-atomic ones).

Definition 3 ([8, 12]) A qualitative temporal or spatial constraint language has patchwork, if for any finite satisfiable constraint networks $\mathcal{N} = (V, C)$ and $\mathcal{N}' = (V', C')$ defined in this language where for any $u, v \in V \cap V'$ we have that $C(u, v) = C'(u, v)$, the constraint network $\mathcal{N} \cup \mathcal{N}'$ is satisfiable.

In light of patchwork, which concerns finite networks, compactness ensures satisfiability of an infinite sequence of finite satisfiable extensions of a network.

Definition 4 ([8]) A qualitative temporal or spatial constraint language has compactness, if any infinite set of constraints defined in this language is satisfiable whenever all its finite subsets are satisfiable.

Intuitively, patchwork ensures that the combination of two satisfiable constraint networks that agree on their common part, i.e., on the constraints between their common variables, continues to be satisfiable, while compactness allows for defining satisfiable networks of infinite size. Global consistency implies patchwork, but the opposite is not true. Even though RCC-8 has patchwork [8], it does not have global consistency [17].

Example. Let us consider the spatial configuration shown in Figure 1(a). Region y is a doughnut, and region x is externally connected to it, by occupying its hole. Further, region z is externally connected to region y . For RCC-8 we know that the constraint network $\{EC(x, y), EC(y, z), EC(x, z)\}$ is satisfiable as it is \diamond -consistent. However, the valuation of region variables x and y is such that it is impossible to extend it with a valuation of region variable z so

that $EC(x, z)$ may hold. Patchwork allows us to disregard any partial valuations and focus on the satisfiability of the network. Then, we can consider a valuation that respects the constraint network. Such a valuation is, for example, the one presented in Figure 1(b) along with its atomic QCN on the right.

3 The \mathcal{L}_1 spatiotemporal logic

In general, a spatial QCN, as described in Section 2, constitutes a static spatial configuration in some domain, over a set of spatial variables V . To be able to describe a spatial configuration that changes over time, we can combine PTL [9] with a qualitative spatial constraint language in a unique formalism. The domain \mathbb{D} of a QCN will always remain the same, but the spatial variables in it may spatially change with the passing time (e.g., in shape, size, or orientation). We can interpret formulas of such a spatiotemporal formalism using a spatiotemporal structure defined as follows.

Definition 5 A ST-structure is a tuple $\mathcal{M}_{\text{ST}} = (\mathbb{D}, \mathbb{N}, \alpha)$, where α is a mapping that associates elements of \mathbb{D} to the spatial variables of a set V at a point of time $i \in \mathbb{N}$. Thus, $\alpha(i)$ denotes the set of elements of \mathbb{D} that are associated with the spatial variables of V at point of time i . By extending notation, $\alpha(v, i)$, where $v \in V$, denotes the element of \mathbb{D} that is associated with spatial variable v at point of time i .

For example, in the case of RCC-8, α would be a mapping associating elements of $\mathcal{R}(\mathcal{T})$ to spatial region variables at a point of time $i \in \mathbb{N}$. The set of atomic propositions AP in the case of standalone PTL [9] is replaced by the set of base relations \mathbb{B} of the qualitative spatial constraint language considered. We will call such a spatiotemporal formula over \mathbb{B} a \mathcal{L}_0 formula. Thus, the set of \mathcal{L}_0 formulas over \mathbb{B} is inductively defined as follows: if $P \in \mathbb{B}$ then P is a \mathcal{L}_0 formula, and if ψ and ϕ are \mathcal{L}_0 formulas then $\neg\phi$, $\phi \vee \psi$, $\circ\phi$, $\square\phi$, $\diamond\phi$, and $\phi \mathcal{U} \psi$ are \mathcal{L}_0 formulas.

A simple example of a \mathcal{L}_0 formula is $\square NTPP(\text{Athens}, \text{Greece})$, stating that Athens will always be located in Greece. To increase the expressiveness of the \mathcal{L}_0 logic we can allow the application of operator \circ to spatial variables, i.e., we can have the following statement in RCC-8: $\square EQ(\text{Greece}, \circ \text{Greece})$, which translates to ‘‘Greece will never change its borders’’. We call the enriched logic the \mathcal{L}_1 logic.

Definition 6 Given a \mathcal{L}_1 formula ϕ over \mathbb{B} , we write $\langle \mathcal{M}_{\text{ST}}, i \rangle \models \phi$ for the fact that \mathcal{M}_{ST} satisfies ϕ at point

of time i , with $i \in \mathbb{N}$ (or formula ϕ is true in \mathcal{M}_{ST} at point of time i). The semantics is then defined as follows:

- $\langle \mathcal{M}_{\text{ST}}, i \rangle \models P(\circ^n v, \circ^m v')$ iff the relation that holds between $\alpha(v, i+n)$ and $\alpha(v', i+m)$ is the relation P , with $P \in \mathcal{B}$
- $\langle \mathcal{M}_{\text{ST}}, i \rangle \models \neg \phi$ iff $\langle \mathcal{M}_{\text{ST}}, i \rangle \not\models \phi$
- $\langle \mathcal{M}_{\text{ST}}, i \rangle \models \phi \vee \psi$ iff $\langle \mathcal{M}_{\text{ST}}, i \rangle \models \phi$ or $\langle \mathcal{M}_{\text{ST}}, i \rangle \models \psi$
- $\langle \mathcal{M}_{\text{ST}}, i \rangle \models \phi \mathcal{U} \psi$ if there exists a $k \in \mathbb{N}$ such that $i \leq k$, $\langle \mathcal{M}_{\text{ST}}, k \rangle \models \psi$, and for all $j \in \mathbb{N}$, if $i \leq j$ and $j < k$ then $\langle \mathcal{M}_{\text{ST}}, j \rangle \models \phi$

Formulas of the form $\diamond \phi$ and $\square \phi$ are abbreviations for $\top \mathcal{U} \phi$ and $\neg(\top \mathcal{U} \neg \phi)$ respectively. A structure $\mathcal{M}_{\text{ST}} = (\mathcal{D}, \mathbb{N}, \alpha)$, for which $\langle \mathcal{M}_{\text{ST}}, 0 \rangle \models \phi$, is a model for ϕ . It follows that a \mathcal{L}_1 formula ϕ is satisfiable if there exists a model for it. Note that a formula of the form $\circ^k P(\circ^l v, \circ^m v')$ is equivalent to formula $P(\circ^{l+k} v, \circ^{m+k} v')$. The size of $P(\circ^{l+k} v, \circ^{m+k} v')$ is then defined to be equal to $\max\{l+k, m+k\}$. Like in [2], we define the size of any \mathcal{L}_1 formula ϕ , denoted by $|\phi|$, inductively as follows: $P(\circ^l v, \circ^m v') = \max\{l, m\}$; $|\neg \phi| = |\phi|$; $|\phi \vee \psi| = |\phi \mathcal{U} \psi| = \max\{|\phi|, |\psi|\}$. The size of a set of \mathcal{L}_1 formulas $\chi = \{\phi, \psi, \dots\}$, will be the maximum size among its formulas, i.e., $|\chi| = \max\{|\phi|, |\psi|, \dots\}$. The number of occurrences of symbols in a \mathcal{L}_1 formula ϕ will be denoted by $\text{length}(\phi)$.

4 Revisiting the satisfiability problem in \mathcal{L}_1

In this section, we revisit a result regarding the satisfiability of \mathcal{L}_1 formulas in a ST-structure, using patchwork and compactness. These properties strengthen previous results, in that we do not longer need to restrict atomic QCNs to being globally consistent as in [2, 3], but we can consider atomic QCNs that have compactness and patchwork. As explained in Section 2, compactness and patchwork combined are less strict than global consistency alone.

Given a \mathcal{L}_1 formula ϕ , Balbiani and Condotta in [2] show that the satisfiability of a formula ϕ can be checked by characterizing a particular infinite sequence of finite satisfiable atomic QCNs representing an infinite consistent valuation of ϕ . Each of the QCNs of such a sequence represents a set of spatial constraints in a fixed-width window of time. The set of spatial constraints at point of time i , is given by the i -th QCN in the infinite sequence, and shares spatial constraints with the next QCN. Moreover, in such a sequence, there exists a point of time after which the corresponding QCNs replicate the same set of spatial constraints. The global consistency property is then

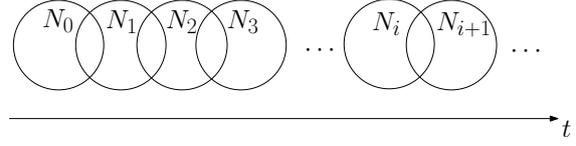


Figure 2: A countably infinite sequence of satisfiable atomic QCNs that agree on their common part

used for the following two tasks:

- to prove that by considering all the QCNs of the aforementioned sequence we obtain a consistent set of constraints;
- to prove that in such an infinite sequence, a subsequence which begins and ends with two QCNs representing the same set of spatial constraints can be reduced to just considering the first QCN.

In the sequel, we formally show that tasks (i) and (ii) can be performed using the properties of patchwork and compactness instead. As a consequence, we can generalize a result regarding the satisfiability of a \mathcal{L}_1 formula ϕ to a larger class of calculi than the previously considered in literature. We now introduce the two aforementioned tasks in the form of two propositions.

Proposition 1 *Let $V = \{v_0, \dots, v_n\}$ be a set of variables, $w \geq 0$ an integer, and $\mathcal{S} = (\mathcal{N}_0 = (V_0, C_0), \mathcal{N}_1 = (V_1, C_1), \dots)$ a countably infinite sequence of satisfiable atomic QCNs, as shown in Figure 2, such that:*

- for each $i \geq 0$, V_i is defined by the set of variables $\{v_0^0, \dots, v_n^0, \dots, v_0^w, \dots, v_n^w\}$,
- for each $i \geq 0$, for all $m, m' \in \{0, \dots, n\}$, and for all $k, k' \in \{1, \dots, w\}$, $C_i(v_m^k, v_{m'}^{k'}) = C_{i+1}(v_m^{k-1}, v_{m'}^{k'-1})$.

We have that if the constraint language considered has compactness and patchwork for atomic QCNs, then \mathcal{S} defines a consistent set of qualitative constraints.

Proof. Given \mathcal{N}_i , we rewrite its set of variables to $\{v_0^i, \dots, v_n^i, \dots, v_0^{w+i}, \dots, v_n^{w+i}\}$. Then, by patchwork we can assert that for each integer $k \geq 0$, $\bigcup_{i \geq k} \mathcal{N}_i$ is a consistent set of qualitative constraints. Suppose though, that $\bigcup_{i \geq 0} \mathcal{N}_i$ is an inconsistent set. By compactness we know that there exists an integer $k' \geq 0$ for which $\bigcup_{i \geq k'} \mathcal{N}_i$ is inconsistent. This is a contradiction. Thus, \mathcal{S} defines a consistent set of qualitative constraints. \dashv

The second proposition follows.

Proposition 2 *Let $V = \{v_0, \dots, v_n\}$ be a set of variables, $w \geq 0, t > t' \geq 0$ three integers, and $\mathcal{S} = (\mathcal{N}_0 = (V_0, C_0), \mathcal{N}_1 = (V_1, C_1), \dots)$ a countably infinite se-*

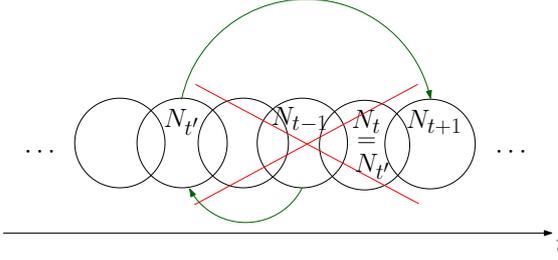


Figure 3: A countably infinite sequence of satisfiable atomic QCNs that contains a sub-sequence which begins and ends with two QCNs representing the same set of spatial constraints; we can reduce the sub-sequence to just considering the first QCN and patch it with the QCN following the sub-sequence

quence of satisfiable atomic QCNs, as shown in Figure 3, such that:

- for each $i \geq 0$, V_i is defined by the set of variables $\{v_0^0, \dots, v_n^0, \dots, v_0^w, \dots, v_n^w\}$,
- for each $i \geq 0$, for all $m, m' \in \{0, \dots, n\}$, and for all $k, k' \in \{1, \dots, w\}$, $C_i(v_m^k, v_{m'}^{k'}) = C_{i+1}(v_m^{k-1}, v_{m'}^{k'-1})$,
- for all $m, m' \in \{0, \dots, n\}$ and all $k, k' \in \{0, \dots, w\}$, $C_{t'}(v_m^k, v_{m'}^{k'}) = C_t(v_m^k, v_{m'}^{k'})$.

Let $\mathcal{S}' = (\mathcal{N}'_0 = (V'_0, C'_0), \mathcal{N}'_1 = (V'_1, C'_1), \dots)$ be the infinite sequence defined by:

- for all $i \in \{0, \dots, t'\}$, $\mathcal{N}'_i = \mathcal{N}_i$,
- for all $i > t'$, $V'_i = V_i$, and for all $m, m' \in \{0, \dots, n\}$ and all $k, k' \in \{0, \dots, w\}$, $C'_i(v_m^k, v_{m'}^{k'}) = C_{i+(t-t')}(v_m^k, v_{m'}^{k'})$.

We have that if the constraint language considered has compactness and patchwork for atomic QCNs, then \mathcal{S}' defines a consistent set of qualitative constraints.

Proof. We have \mathcal{N}_i which is a satisfiable QCN for all $i \geq 0$. From this, we can deduce that \mathcal{N}'_i is a satisfiable QCN for all $i \geq 0$. By Proposition 1 we can deduce that \mathcal{S}' defines a consistent set of qualitative constraints. \dashv

We now can obtain the following result:

Theorem 1 *Checking the satisfiability of a \mathcal{L}_1 formula ϕ in a ST-structure is PSPACE-complete in $\text{length}(\phi)$ if the qualitative spatial constraint language considered has compactness and patchwork for atomic QCNs.*

Proof. (Sketch) Consider the approach in [2] where a proof of PSPACE-completeness is given for a logic that considers qualitative constraint languages for which satisfiable atomic QCNs are globally consistent (see Theorem 1 in [2]). To be able to replace the use of global consistency with the use of patchwork and

compactness, we need to use Propositions 1 and 2 in the proofs of Lemmas 3 and 4 in [2]. The interested reader can verify that the aforementioned proofs make use of global consistency to perform exactly the tasks described by Propositions 1 and 2. Since these propositions build on compactness and patchwork, we can prove PSPACE-completeness using these properties instead. \dashv

Theorem 1 allows us to consider more calculi than the ones considered in literature for which the combination with PTL yields PSPACE-completeness. Due to the lack of global consistency for RCC-8 [17], in [5] the authors restrict themselves to a very particular domain interpretation of RCC-8 to prove that the \mathcal{ST}_1^- logic is PSPACE-complete. As already noted in Section 1, the \mathcal{ST}_1^- logic is the \mathcal{L}_1 logic when the considered qualitative constraint language is RCC-8. \mathcal{L}_1 does not rely on the semantics of the qualitative constraint language used, but rather on the constraint properties of compactness and patchwork [12]. Therefore, \mathcal{L}_1 is by default able to consider all calculi that have these properties, such as RCC-8 [15], Cardinal Direction Algebra (CDA) [4, 11], Block Algebra (BA) [6], and even Interval Algebra (IA) [1] when viewed as a spatial calculus. The most notable languages that have patchwork and compactness are listed in [8].

5 Semantic tableau for \mathcal{L}_1

In this section, we present a semantic tableau method that given a \mathcal{L}_1 formula ϕ systematically searches for a model for ϕ . The method builds on the tableau method for PTL of Wolper [19], and makes use of the results of Section 4 to ensure soundness and completeness.

5.1 Rules for constructing a semantic tableau

The decomposition rules of the temporal operators are based on the following identities, which are called *eventualities* (where \square abbreviates $\neg \diamond \neg$):

- $\diamond \phi \equiv \phi \vee \square \diamond \phi$
- $\phi \mathcal{U} \psi \equiv \psi \vee (\phi \wedge \square(\phi \mathcal{U} \psi))$

Note that decomposing eventualities can lead to an infinite tableau. However, we will construct a finite tableau by identifying nodes that are labeled by the same set of formulas, thus, ensuring that infinite periodicity will not exist. To test a \mathcal{L}_1 formula ϕ for satisfiability, we will construct a directed graph. Each node n of the graph will be labeled by a set of formulas, and initially the graph will contain a single node, labeled by $\{\phi\}$. Similarly to Wolper [19], we distinguish between *elementary* and *non-elementary* formulas:

Definition 7 A \mathcal{L}_1 formula is elementary if its main connective is \circ (viz., \circ -formula), or if it corresponds to a base relation $P \in \mathbf{B}$.

Then, the construction of the graph proceeds by using the following decomposition rules which map each non-elementary formula ϕ into a set of sets of formulas:

- $\neg P(\circ^n v, \circ^m v') \rightarrow \{\{P'(\circ^n v, \circ^m v')\} \mid P' \in \mathbf{B} \setminus \{P\}\}$
- $\neg\neg\phi \rightarrow \{\{\phi\}\}$
- $\neg\circ\phi \rightarrow \{\{\circ\neg\phi\}\}$
- $\phi \wedge \psi \rightarrow \{\{\phi, \psi\}\}$
- $\neg(\phi \wedge \psi) \rightarrow \{\{\neg\phi\}, \{\neg\psi\}\}$
- $\diamond\phi \rightarrow \{\{\phi\}, \{\circ\diamond\phi\}\}$
- $\neg\diamond\phi \rightarrow \{\{\neg\phi, \neg\circ\diamond\phi\}\}$
- $\phi \mathcal{U} \psi \rightarrow \{\{\psi\}, \{\phi, \circ(\phi \mathcal{U} \psi)\}\}$
- $\neg(\phi \mathcal{U} \psi) \rightarrow \{\neg\psi, \neg\phi \vee \neg\circ(\phi \mathcal{U} \psi)\}$

During the construction, we *mark* formulas to which a decomposition rule has been applied to avoid decomposing the same formula twice. If ψ is a formula, ψ^* denotes ψ marked.

5.2 Systematic construction of a semantic tableau

A tableau \mathcal{T} can be seen as a directed graph where each of its nodes n is labeled with a set of formulas $\mathcal{T}(n)$. The root node is labeled with the singleton set $\{\phi\}$ for the \mathcal{L}_1 formula ϕ whose satisfiability we wish to check. The children of the nodes are obtained by applying the rules presented in Section 5.1.

Given a set of \mathcal{L}_1 formulas χ over the set of variables $\{x_0, \dots, x_l\}$, we denote by $expandVars(\chi)$ the set $\{\circ^0 x_0, \dots, \circ^0 x_l, \dots, \circ^{|\chi|} x_0, \dots, \circ^{|\chi|} x_l\}$. We first define a translation of a node of a tableau to a QCN.

Definition 8 Let n be a node of a tableau \mathcal{T} for a \mathcal{L}_1 formula ϕ , and $\{x_0, \dots, x_l\}$ the set of variables in ϕ . Then, $\mathcal{N}(n)$ will denote the QCN $= (V, C)$, where $V = \{v_0^0, \dots, v_l^0, \dots, v_0^{|\phi|}, \dots, v_l^{|\phi|}\}$, and $C(v_m^k, v_{m'}^{k'}) = \{P(\circ^k x_m, \circ^{k'} x_{m'})\}$ if $P(\circ^k x_m, \circ^{k'} x_{m'}) \in \mathcal{T}(n)$, and $C(v_m^k, v_{m'}^{k'}) = (\mathbf{B}$ if $v_m^k \neq v_{m'}^{k'}$ else $\{\text{Id}\})$ otherwise, $\forall m, m' \in \{0, \dots, l\}$ and $\forall k, k' \in \{0, \dots, |\phi|\}$.

Let us also define the notions of a *state* and a *pre-state*, which we will be referring to a lot in what follows.

Definition 9 A node n that contains only elementary and marked formulas and for which we have that $\mathcal{N}(n)$ is atomic is called a state, and a node m that is either the root node or the direct child node of a state (which leaps to the next point of time) is called a pre-state.

We give a definition of eventuality fulfillment that will be of use later on.

Algorithm 1: Clotho(ϕ)

```

in      : A  $\mathcal{L}_1$  formula  $\phi$ .
output : A semantic tableau  $\mathcal{T}$  for  $\phi$ .
1 begin
2   create root node  $\{\phi\}$  and mark it unprocessed;
3   while  $\exists$  unprocessed node  $n$  do
4     if  $\mathcal{T}(n)$  contains an unmarked non-elementary
      formula  $\psi$  then
5       mark node  $n$  processed;
6       foreach  $\gamma \in \Gamma$ , where  $\Gamma$  is the result of
          applying a decomposition rule to  $\psi$  do
7         create a child node  $m$ ;
8          $\mathcal{T}(m) \leftarrow (\mathcal{T}(n) - \{\psi\}) \cup \gamma \cup \{\psi^*\}$ ;
9         mark node  $m$  unprocessed;
10      else if  $\mathcal{T}(n)$  contains only elementary and
          marked formulas then
11        mark node  $n$  processed;
12        filling  $\leftarrow \emptyset$ ;
13        foreach  $u, v \in expandVars(\phi)$  do
14          if  $\nexists P(u, v) \in \mathcal{T}(n)$  then
15            filling  $\leftarrow$  filling  $\cup \{\mathbf{B}(u, v)\}$ ;
16        if filling  $\neq \emptyset$  then
17          create a child node  $m$ ;
18           $\mathcal{T}(m) \leftarrow \mathcal{T}(n) \cup$  filling;
19          mark node  $m$  unprocessed;
20        else if  $\mathcal{T}(n)$  contains  $\circ$ -formulas then
21          create a child node  $m$ ;
22           $\mathcal{T}(m) \leftarrow \{\psi \mid \circ\psi \in \mathcal{T}(n)\}$ ;
23           $\mathcal{T}(m) \leftarrow \mathcal{T}(m) \cup \{P(\circ^{i-1}u, \circ^{j-1}v) \mid$ 
24             $P(\circ^i u, \circ^j v) \in \mathcal{T}(n) \text{ if } i, j \geq 1\}$ ;
            mark node  $m$  unprocessed;

```

Definition 10 Let \mathcal{T} be a tableau, and π a path in \mathcal{T} defined from nodes n_1, n_2, \dots, n_j . Any eventuality $\diamond\epsilon_2$ or $\epsilon_1 \mathcal{U} \epsilon_2 \in \mathcal{T}(n_i)$, with $1 \leq i \leq j$, is fulfilled in π if there exists k , with $i \leq k \leq j$, such that $\epsilon_2 \in \mathcal{T}(n_k)$.

We now present Clotho, an algorithm that constructs a semantic tableau \mathcal{T} for a given formula ϕ , as shown in Algorithm 1. At any given point of time, we construct all the possible atomic QCNs comprising base relations that extend from the given point of time to a future point of time. This is achieved by repeatedly applying the decomposition rules to a node comprising unmarked non-elementary formulas (lines 4 to 9), and sequentially populating a node comprising only elementary and marked formulas with the universal relation \mathbf{B} (lines 10 to 19) so that it may lead to a state. The universal relation \mathbf{B} is only introduced on a pair of variables, if there does not exist any base relation on that same pair. The universal relation \mathbf{B} , as well as any other relation $r \in 2^{\mathbf{B}}$, is essentially the disjunction of base relations, as noted in Section 2. In particular, \mathbf{B} is the disjunction of all the base relations of a given qualitative constraint language. As such, by decomposing \mathbf{B} into base relations using the

Algorithm 2: Atropos(\mathcal{T})

```

in      : A semantic tableau  $\mathcal{T}$ .
output : True or False.
1 begin
2   do
3     flag  $\leftarrow$  False;
4     if there is a node  $n$  such that  $\mathcal{N}(n)$  is an
       unsatisfiable QCN then
5       | eliminate node  $n$ ; flag  $\leftarrow$  True;
6     if all the children of a node  $n$  have been
       eliminated then
7       | eliminate node  $n$ ; flag  $\leftarrow$  True;
8     if a node  $n$  is a pre-state and not Lachesis( $\mathcal{T}, n$ )
       then
9       | eliminate node  $n$ ; flag  $\leftarrow$  True;
10    while flag;
11    if  $\exists$  node  $n \in \mathcal{T}$  then return False else return
       True;

```

Function Lachesis(\mathcal{T}, n)

```

in      : A semantic tableau  $\mathcal{T}$ , and a node  $n$ .
output : True or False.
1 begin
2   foreach eventuality  $\epsilon \in \mathcal{T}(n)$  do
3     | if  $\epsilon$  is not fulfilled in any path  $\pi = \langle n, \dots \rangle$  then
       return False;
4   return True;

```

disjunctive tableau rule, this approach allows us to obtain one or more nodes harboring atomic QCNs for a given point of time (viz., states), that represent a set of atomic spatial constraints in a fixed-width window of time. Once we have obtained our atomic QCNs for a given point of time, and assuming that the states that harbor them contain \circ -formulas, we can leap to the next point of time and create pre-states, including all the atomic spatial constraints of the aforementioned QCNs that extend from the new point of time to a future point of time (lines 20 to 24). This can be seen as making a +1 time shift and maintaining all possible knowledge offered by previous states that extends from the new point of time to a future point of time. It is important to note that when we create the child node of a node n (lines 7, 17, and 21), we only create a new node if there does not already exist a node in the graph labeled by $\mathcal{T}(n)$. Otherwise, we just create an arc from node n to the existing node.

Lemma 1 *Let \mathcal{T} be a tableau for a \mathcal{L}_1 formula ϕ that has resulted after the application of algorithm Clotho. Then, \mathcal{T} is finite. Actually, if ϕ is over a set of l variables, then \mathcal{T} has at most $O(|B|^{l^2 \cdot (|\phi|+1)^3} \cdot 2^{\text{length}(\phi)})$ nodes.*

To decide the satisfiability of a \mathcal{L}_1 formula ϕ using the

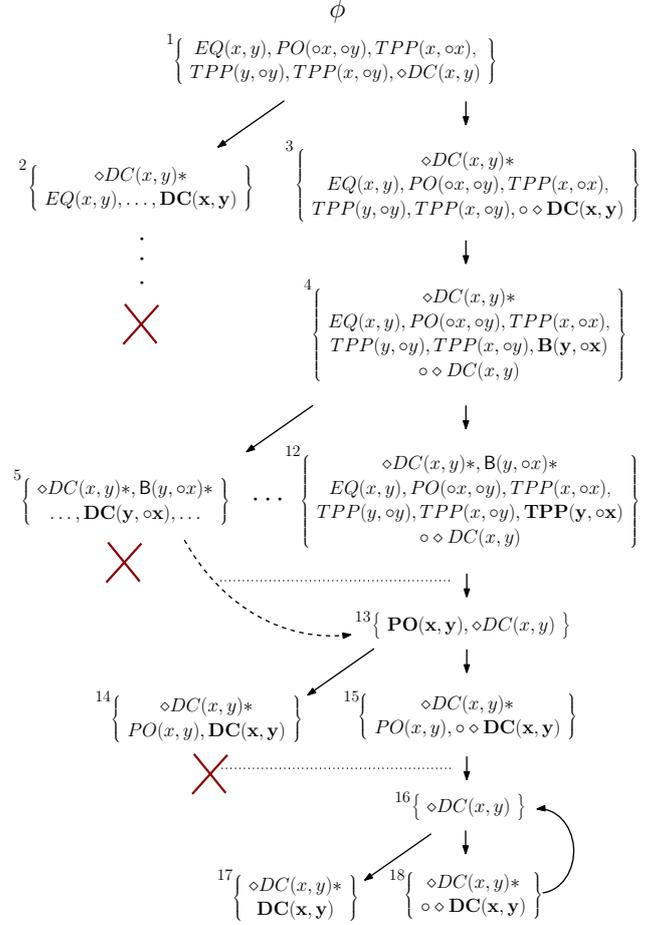

 Figure 4: A \mathcal{L}_1 formula and its simplified tableau

tableau that is generated by Clotho, we have to eliminate unsatisfiable nodes inductively, until a fixed point is reached. We present Atropos, an algorithm that achieves this goal, shown in Algorithm 2. If the root node is eliminated after the application of Atropos, we call the tableau *closed*, and *open* otherwise. Note that function Lachesis essentially searches for a path from a given pre-state to a node that fulfills an eventuality of the pre-state, as defined in Definition 10.

Example. Let us consider formula $\phi = \{EQ(x, y), PO(\circ x, \circ y), TPP(x, \circ x), TPP(y, \circ y), TPP(x, \circ y), \diamond DC(x, y)\}$. (For simplicity we assume that the decomposition rule for \wedge has already been applied and resulted in the current set form for formula ϕ .) The tableau obtained by the application of algorithms Clotho and Atropos for this formula is shown in Figure 4. Horizontal dotted lines distinguish between different points in time, thus, our tableau extends over three points of time. The root node is 1, the states are 5 to 12, 14, 15, 17, and 18, and the pre-states are 1, 13, and 16. By decomposing the initial formula using the tableau rules and populating it with universal

relations where appropriate, we reach states 5 to 12, each one of which harbors a set of base relations that correspond to an atomic QCN. (Inverse relations are not shown to save space.) These atomic QCNs represent a set of atomic spatial constraints in a fixed-width window of time. After leaping to the next point of time and, consequently, obtaining pre-state 13, we include all the atomic spatial constraints of the aforementioned QCNs that extend from the new point of time to a future point of time. In this particular case, the atomic spatial constraints of interest narrow down to the single atomic constraint $PO(\circ x, \circ y)$, common for all states 5 to 12. Of course, since we are now at the next point of time, the constraint is rewritten to $PO(x, y)$. Again, we apply the rules and reach states 14 and 15, each one of which harbors an atomic QCN. We continue repeating the process until all our child nodes are labeled by sets of formulas already met in nodes of the tableau. In this case, the unique child node of state 18 would be labeled by the set of formulas of node 16, thus, we create an arc from 18 to 16. After having constructed our tableau, we delete unsatisfiable nodes 2, 5 to 11, and 14 using the \diamond -consistency operation on QCNs $\mathcal{N}(2)$, $\mathcal{N}(5)$ to $\mathcal{N}(11)$, and $\mathcal{N}(14)$ respectively. Inconsistencies stemming from nodes 2 and 14 are apparent, as there exist different base relations on a same pair of variables, whereas inconsistencies in nodes 5 to 11 stem from the fact that relation $TPP(y, \circ x)$ is inferred by \diamond -consistency, which contradicts with the base relation that is defined on variables y and $\circ x$ in states 5 to 11. Formula ϕ is satisfiable, as the tableau is open, and a model can be constructed out of the sequence of states 12,15,17 which contains a self loop on 17 as relation $DC(x, y)$ repeats itself. These states harbor satisfiable atomic QCNs that completely agree on their common part due to our construction. In particular, we have the sequence $\mathcal{N}(12) \rightarrow \mathcal{N}(15) \rightarrow \mathcal{N}(17) \circ$ that satisfies the prerequisites of Proposition 2, hence, satisfiability is met.

5.3 Soundness and completeness of our semantic tableau method

In this section, we prove that the tableau method as defined by algorithms *Clotho* and *Atropos* is sound and complete for checking the satisfiability of a \mathcal{L}_1 formula ϕ .

Theorem 2 (soundness) *If ϕ has a closed tableau, then ϕ is unsatisfiable.*

Proof. Let \mathcal{T} be a closed tableau for ϕ , that has resulted after the application of algorithms *Clotho* and

Atropos. We prove by induction that if a node n is eliminated, then $\mathcal{T}(n)$ is an unsatisfiable set of formulas. We distinguish three scenarios:

- (i) a node n is eliminated because $\mathcal{N}(n)$ is an unsatisfiable QCN (lines 4 to 5 in *Atropos*), thus, $\mathcal{T}(n)$ is an unsatisfiable set of formulas; unsatisfiability of $\mathcal{N}(n)$ can be detected by use of \diamond -consistency, which also disallows the conjunction of two or more base relations to be defined on a same pair of variables (base relations are jointly exhaustive and pairwise disjoint as noted in Section 2).
- (ii) a node n is eliminated because all of its child nodes are unsatisfiable and have been eliminated (lines 6 to 7 in *Atropos*). Child nodes can be created in the following three cases:
 - (a) the decomposition rule $\psi \rightarrow \Gamma$, where $\psi \in \mathcal{T}(n)$, is applied and a child node is created for each $\gamma \in \Gamma$ (lines 4 to 9 in *Clotho*); we have that ψ is satisfiable iff $\exists \gamma \in \Gamma$ that is satisfiable.
 - (b) implicit knowledge in the parent node n is made explicit in the child node m through the introduction of the universal relation \mathbf{B} (lines 10 to 19 in *Clotho*); by Definition 8 we have that $\mathcal{N}(n) = \mathcal{N}(m)$, thus, $\mathcal{N}(n)$ is satisfiable iff $\mathcal{N}(m)$ is satisfiable, and the same holds for the set of formulas $\mathcal{T}(n)$ and $\mathcal{T}(m)$.
 - (c) node n is a state and generates pre-state m with $\mathcal{T}(m) = \{\psi \mid \circ \psi \in \mathcal{T}(n)\} \cup \{P(\circ^{i-1}u, \circ^{j-1}v) \mid P(\circ^i u, \circ^j v) \in \mathcal{T}(n) \text{ if } i, j \geq 1\}$ (lines 20 to 24 in *Clotho*); clearly, $\mathcal{T}(n)$ is a satisfiable set of formulas iff $\{\psi \mid \circ \psi \in \mathcal{T}(n)\}$ is a satisfiable set of formulas and iff $\mathcal{N}(m)$ is satisfiable.
- (iii) a node n is eliminated if it contains an eventuality that cannot be fulfilled in any path in the tableau (lines 8 to 9 in *Atropos*); since any model will correspond to a path in the tableau, we have that $\mathcal{T}(n)$ is an unsatisfiable set of formulas.

As we have considered all possible scenarios, at this point we conclude our proof. \dashv

Let us obtain a proposition that denotes that two successive states in a path of an open tableau harbor QCNs that completely agree on their common part.

Proposition 3 *Let π be a path going through an open tableau \mathcal{T} for a \mathcal{L}_1 formula ϕ that has resulted after the application of algorithms *Clotho* and *Atropos*, s_t and s_{t+1} two states of π belonging to points of time t and $t + 1$ respectively, and $\{x_0, \dots, x_l\}$ the set of variables in ϕ . Then we have that $\mathcal{N}(s_t)[v_m^k, v_{m'}^{k'}] = \mathcal{N}(s_{t+1})[v_m^{k-1}, v_{m'}^{k'-1}] \forall m, m' \in \{0, \dots, l\}$ and $\forall k, k' \in \{1, \dots, |\phi|\}$.*

Proof. State s_t at point of time t is followed by a pre-state p at point of time $t + 1$ in path π , whose set of base relations is $\{P(\circ^{i-1}u, \circ^{j-1}v) \mid P(\circ^i u, \circ^j v) \in \mathcal{T}(s_t) \text{ if } i, j \geq 1\} \cup \{P(\circ^i u, \circ^j v) \mid \circ P(\circ^i u, \circ^j v) \in \mathcal{T}(s_t)\}$ by construction of our tableau (lines 20 to 24 in *Clotho*). The set of base relations of $\mathcal{T}(p)$ is carried over, possibly enriched, to state s_{t+1} at point of time $t + 1$. As such, let us assume that there exists an additional base relation $b(\circ^{i-1}u, \circ^{j-1}v)$ in the set of base relations of s_{t+1} , with $i, j \in \{1, \dots, |\phi|\}$, such that $b(\circ^i u, \circ^j v) \notin \mathcal{T}(s_t)$. In this case, $\mathcal{N}(s_{t+1})$ is a QCN with two base relations defined on a same pair of variables. This QCN would have been deleted by the application of *Atropos* as specified also in the proof of Theorem 2. Thus, state s_{t+1} could not have been in path π , resulting in a contradiction. Therefore, we have that $\mathcal{N}(s_t)[v_m^k, v_{m'}^{k'}] = \mathcal{N}(s_{t+1})[v_m^{k-1}, v_{m'}^{k'-1}] \forall m, m' \in \{0, \dots, l\}$ and $\forall k, k' \in \{1, \dots, |\phi|\}$, and, as such, $\mathcal{N}(s_t)$ and $\mathcal{N}(s_{t+1})$ completely agree on their common part. \dashv

Theorem 3 (completeness) *If ϕ has an open tableau, then ϕ is satisfiable.*

Proof. Let \mathcal{T} be an open tableau for ϕ , that has resulted after the application of algorithms *Clotho* and *Atropos*. We need to show that there exists a path of nodes π which defines a model for ϕ . We distinguish two scenarios:

- (i) if no eventualities need to be fulfilled, path π can be simply a path starting from the root node and going through the tableau, defining a sequence of states s_0, s_1, \dots, s_t , with $t \in \mathbb{N}$, and, consequently, yielding a sequence of QCNs as follows:

$$\mathcal{N}(s_0) \rightarrow \mathcal{N}(s_1) \dots \rightarrow \mathcal{N}(s_t)$$

The sequence of QCNs is such that for all states s_i and s_{i+1} , with $i \in \{0, \dots, t-1\}$, along with a set of variables $\{x_0, \dots, x_l\}$ in ϕ , we have that $\mathcal{N}(s_i)[v_m^k, v_{m'}^{k'}] = \mathcal{N}(s_{i+1})[v_m^{k-1}, v_{m'}^{k'-1}] \forall m, m' \in \{0, \dots, l\}$ and $\forall k, k' \in \{1, \dots, |\phi|\}$ by Proposition 3. Thus, the sequence of QCNs corresponds to the sequence shown in Figure 2, satisfies the prerequisites of Proposition 1, and is therefore satisfiable.

- (ii) if eventualities need to be fulfilled, we show how we can construct a path π that fulfills all eventualities as follows. For each pre-state $p \in \mathcal{T}$ containing an eventuality, we must find a path $\pi_p = \langle p, \dots \rangle$ starting from p , such that all the eventualities contained in p are fulfilled in π_p . We fulfill all the eventualities of p , one by one, as follows. For a selected eventuality $\epsilon \in \mathcal{T}(p)$, it is possible to find a path $\pi_p = \langle p, \dots, p' \rangle$ in which

ϵ is fulfilled and whose last node is a pre-state p' , as otherwise the node would have been deleted by the application of *Atropos*. By construction of our tableau, p' will also contain the rest of the eventualities that need to be fulfilled (they are carried over from p to p'), and it follows that we can extend path π_p to fulfill a second one, and so on, until all the eventualities of p are fulfilled. By linking together all paths $\pi_p \forall$ pre-states $p \in \mathcal{T}$, we can obtain a path π starting from the initial node and going through the tableau, defining a sequence of states s_0, s_1, \dots, s_{t-1} , with $t \in \mathbb{N}$, with a final loop between state s_{t-1} and a state $s_{t'}$, with $0 \leq t' \leq t-1$. The loop exists due to the fact that at point of time $t-1$ there exists a node n , whose child node m is such that $\mathcal{T}(m) = \mathcal{T}(o)$, where o is a node at point of time t' . In particular, we can view the sequence of states as a sequence of QCNs as follows:

$$\mathcal{N}(s_0) \rightarrow \mathcal{N}(s_1) \dots \rightarrow \underbrace{\mathcal{N}(s_{t'}) \dots \rightarrow \mathcal{N}(s_{t-1})}_{\text{loop}}$$

The sequence of QCNs is such that for all states s_i and s_{i+1} , with $i \in \{0, \dots, t-2\}$, along with a set of variables $\{x_0, \dots, x_l\}$ in ϕ , we have that $\mathcal{N}(s_i)[v_m^k, v_{m'}^{k'}] = \mathcal{N}(s_{i+1})[v_m^{k-1}, v_{m'}^{k'-1}] \forall m, m' \in \{0, \dots, l\}$ and $\forall k, k' \in \{1, \dots, |\phi|\}$ by Proposition 3. Further, if we were to extend path π , we would obtain a state s_t with $\mathcal{N}(s_t)[v_m^k, v_{m'}^{k'}] = \mathcal{N}(s_{t'})[v_m^k, v_{m'}^{k'}] \forall m, m' \in \{0, \dots, l\}$ and $\forall k, k' \in \{0, \dots, |\phi|\}$ (i.e., s_t replicates the same set of spatial constraints with $s_{t'}$, hence, the loop). Thus, the sequence of QCNs corresponds to the sequence shown in Figure 3, satisfies the prerequisites of Proposition 2, and is therefore satisfiable.

As we have considered all possible scenarios, at this point we conclude our proof. \dashv

6 Conclusion

In this paper, we considered a generalized qualitative spatiotemporal formalism, namely, the \mathcal{L}_1 logic, which is the product of the combination of PTL with any qualitative spatial constraint language, such as RCC-8, Cardinal Direction Algebra, and Block Algebra, and showed that satisfiability checking of a \mathcal{L}_1 formula is PSPACE-complete if the constraint language considered has the properties of *compactness* and *patchwork* for atomic networks, thus, strengthening previous results that required atomic networks to be *globally consistent* and generalizing to a larger class of calculi. Further, we presented a first semantic tableau method, that given a \mathcal{L}_1 formula ϕ systematically searches for a model for ϕ . The method presented builds on the

tableau method for PTL of Wolper, and makes use of our strengthened results to ensure soundness and completeness.

7 Future Work

In this paper, we implicitly considered that the satisfiability problem in a qualitative spatial constraint language relies on some canonical model, i.e., a structure that allows to model any (syntactically) consistent QCN. It is a very interesting future direction to consider domain interpretations that involve determined entities (constants) for a given qualitative spatial constraint language, and not just abstract infinite domains as it is normally the case. It has been shown that such domain interpretations do not always yield semantic truth when combined with the algorithms that are typically used in conjunction with some canonical model to derive the satisfiability of a QCN [10]. Therefore, we need to explore this implication in the satisfiability problem in \mathcal{L}_1 .

Acknowledgment

The work was funded by a PhD grant from Université d'Artois and region Nord-Pas-de-Calais.

References

- [1] James F. Allen. Maintaining Knowledge about Temporal Intervals. *Commun. ACM*, 26:832–843, 1983.
- [2] Philippe Balbiani and Jean-François Condotta. Computational Complexity of Propositional Linear Temporal Logics Based on Qualitative Spatial or Temporal Reasoning. In *FroCoS*, 2002.
- [3] Stéphane Demri and Deepak D'Souza. An automata-theoretic approach to constraint LTL. *Inf. Comput.*, 205:380–415, 2007.
- [4] Andrew U. Frank. Qualitative Spatial Reasoning with Cardinal Directions. In *ÖGAI*, 1991.
- [5] David Gabelaia, Roman Kontchakov, Agi Kurucz, Frank Wolter, and Michael Zakharyashev. On the Computational Complexity of Spatio-Temporal Logics. In *FLAIRS*, 2003.
- [6] Hans Werner Guesgen. Spatial Reasoning Based on Allen's Temporal Logic. Technical report, International Computer Science Institute, 1989.
- [7] S.M. Hazarika. *Qualitative Spatio-Temporal Representation and Reasoning: Trends and Future Directions*. Igi Global, 2012.
- [8] Jinbo Huang. Compactness and its implications for qualitative spatial and temporal reasoning. In *KR*, 2012.
- [9] Michael Huth and Mark Ryan. *Logic in Computer Science: Modelling and Reasoning About Systems*. 2004.
- [10] Sanjiang Li, Weiming Liu, and Sheng-sheng Wang. Qualitative constraint satisfaction problems: An extended framework with landmarks. *AIJ*, 201:32–58, 2013.
- [11] Gerard Ligozat. Reasoning about cardinal directions. *J. Vis. Lang. Comput.*, 9(1):23–44, 1998.
- [12] C. Lutz and M. Milicic. A Tableau Algorithm for DLs with Concrete Domains and GCIs. *JAR*, 38:227–259, 2007.
- [13] J.R. Munkres. *Topology*. Prentice Hall, Incorporated, 2000.
- [14] Franco P. Preparata and Michael Ian Shamos. *Computational Geometry - An Introduction*. Springer, 1985.
- [15] David A. Randell, Zhan Cui, and Anthony Cohn. A Spatial Logic Based on Regions and Connection. In *KR*, 1992.
- [16] Jochen Renz. Maximal Tractable Fragments of the Region Connection Calculus: A Complete Analysis. In *IJCAI*, 1999.
- [17] Jochen Renz and Gérard Ligozat. Weak Composition for Qualitative Spatial and Temporal Reasoning. In *CP*, 2005.
- [18] P. A. Story and Michael F. Worboys. A Design Support Environment for Spatio-Temporal Database Applications. In *COSIT*, 1995.
- [19] Pierre Wolper. The tableau method for temporal logic: An overview. *Logique et Analyse*, 28:119–136, 1985.
- [20] Frank Wolter and Michael Zakharyashev. Spatio-temporal representation and reasoning based on RCC-8. In *KR*, 2000.
- [21] Frank Wolter and Michael Zakharyashev. Exploring artificial intelligence in the new millennium. chapter Qualitative Spatiotemporal Representation and Reasoning: A Computational Perspective. Morgan Kaufmann Publishers Inc., 2003.

Sur des propriétés structurelles des formules de Horn

Bruno Zanuttini

GREYC, UMR 6072 ; UNICAEN, CNRS, ENSICAEN
bruno.zanuttini@unicaen.fr

Résumé

Nous étudions les propriétés structurelles des formules de Horn en lien avec leur graphe de dépendance. Nous donnons une caractérisation simple des arcs de ce graphe, en termes d'affectations qui satisfont ou non une formule, et nous en dérivons un algorithme efficace, pour construire le graphe de dépendance de la borne supérieure de Horn d'un ensemble de modèles ; en particulier donc, de la borne supérieure d'un ensemble de modèles caractéristiques. Ce résultat nous permet également de montrer que la pertinence d'une hypothèse, pour un problème d'abduction de Horn, peut être décidée efficacement avec une représentation par ensemble de modèles.

Abstract

We investigate the structural properties of Horn formulas that concern their dependency graph. We give a simple characterisation of the arcs in this graph in terms of assignments that satisfy or falsify a formula, and we derive an efficient algorithm for building the dependency graph of the Horn Least Upper Bound of a set of models ; thus, in particular, of a set of characteristic models. This result also allows us to show that the relevance of a hypothesis for a Horn abduction problem can be decided efficiently under model-based representation.

1 Introduction

Les formules propositionnelles CNF de Horn ont fait l'objet de nombreux travaux dans la littérature d'intelligence artificielle. Ceci est dû essentiellement à l'existence d'algorithmes polynomiaux pour décider la satisfaisabilité et l'implication de clauses pour de telles formules [15], contrairement au cas des formules propositionnelles en général, et même des formules CNF non restreintes. De tels algorithmes permettent en effet à un agent qui utilise une

base de connaissances de Horn, de raisonner plus efficacement que s'il avait une base non restreinte.

Il y a toutefois plusieurs problèmes qui ne peuvent pas être résolus efficacement avec des théories représentées par des formules CNF de Horn. Des exemples de tels problèmes sont la minimisation [7] et l'abduction [17], et ceci a motivé l'étude de restrictions additionnelles sur les formules de Horn. En particulier, l'acyclicité joue un rôle important dans les solutions algorithmiques, et permet la minimisation efficace d'une théorie [8] et l'énumération, en temps total polynomial, des explications, pour plusieurs types de problèmes d'abduction, contrairement au cas des formules de Horn non restreintes [6].

L'acyclicité d'une formule de Horn est définie sur son *graphe de dépendance*, c'est-à-dire le graphe orienté dont les sommets sont les variables de la formule, et qui contient un arc de x_i à x_j si et seulement si il y a une clause contenant $\neg x_i$ et x_j dans la formule. Les résultats de Boros *et al.* [2] montrent que la clôture transitive de ce graphe est invariante par équivalence logique (lorsque l'on considère des formules *premières*). Ceci nous permet d'étudier ce graphe en termes d'ensembles de modèles plutôt que de formules.

Les caractérisations de propriétés de formules CNF en termes de leurs modèles se sont souvent avérées très utiles. Par exemple, il est bien connu que l'ensemble des modèles d'une formule CNF de Horn est clos par le *Et* logique bit-à-bit [14], et ceci a permis la conception de nombreux algorithmes efficaces (voir par exemple le cas de l'identification [4]). Nous donnons ici une caractérisation des arcs du graphe de dépendance d'une formule CNF de Horn. Il s'avère que cette caractérisation est valide même pour un sous-ensemble d'un ensemble de modèles M , pourvu que sa clôture par le *Et* logique bit-à-bit soit exactement M . Notre caractérisation est donc valide en particulier pour l'ensemble des *modèles caractéristiques* d'une formule [11].

Nos principales contributions sont les suivantes. En utilisant notre caractérisation, nous proposons un algorithme efficace pour construire le graphe de dépendance de la borne supérieure de Horn (*Horn-LUB* [16]) d'un ensemble donné de modèles (précisément, pour construire la *clôture transitive* de ce graphe). Cet algorithme fonctionne en temps cubique, et nous en donnons une version plus efficace pour le cas des formules CNF de Horn *définies positives*. Nous pensons que cette caractérisation est intéressante en soi, mais nous en donnons également deux applications. La première, très directe, est un algorithme efficace pour décider si un ensemble de modèles caractéristiques est celui d'une formule CNF de Horn acyclique. La seconde est un algorithme efficace pour décider si une hypothèse donnée est pertinente (pour l'inclusion) pour un problème d'abduction de Horn dont la théorie est représentée par un ensemble de modèles caractéristiques. Ce dernier résultat contraste avec la représentation par une formule, cas pour lequel le problème est NP-complet.

2 Formules CNF de Horn et graphes de dépendance

Dans tout l'article, nous supposons fixé un ensemble fini de variables propositionnelles x_1, \dots, x_n . Un *littéral* ℓ est soit une variable x_i (littéral *positif*), soit la négation $\neg x_i$ d'une variable (littéral *négatif*). Une *clause* est un ensemble de littéraux $\{\ell_1, \dots, \ell_k\}$, vu comme leur disjonction ; nous supposons sans perte de généralité que dans une clause, deux littéraux distincts sont construits sur des variables différentes. Une *formule en forme normale conjonctive* (CNF) est un ensemble fini de clauses, vu comme leur conjonction. La *taille* d'une CNF est définie comme le nombre total d'occurrences de variables, c'est-à-dire comme la somme des cardinalités de ses clauses.

Une *affectation* est une fonction de $\{x_1, \dots, x_n\}$ dans $\{0, 1\}$. Nous écrivons $m[i]$ pour la valeur affectée par m à x_i , et si $I \subseteq \{1, \dots, n\}$ vérifie $\forall i \in I, m[i] = c$ ($c \in \{0, 1\}$), nous notons $m[I] = c$. Nous noterons parfois une affectation m par le mot $m[1]m[2] \dots m[n]$. Enfin, si m_1, \dots, m_k sont des affectations, $ones(m_1, \dots, m_k)$ désigne l'ensemble de tous les indices i vérifiant $m_1[i] = \dots = m_k[i] = 1$.

Nous disons qu'une affectation m *satisfait* un littéral ℓ si l'on a $\ell = x_i$ et $m[i] = 1$, ou $\ell = \neg x_i$ et $m[i] = 0$. Nous disons que c'est un *modèle* d'une CNF φ si elle satisfait au moins un littéral dans chaque clause de φ . L'ensemble de tous les modèles de φ est noté $Mods(\varphi)$. La *taille* d'un ensemble d'affectations est le nombre n de variables, multiplié par le nombre d'affectations dans l'ensemble. On dit qu'une CNF φ *implique* une CNF φ' (ou une clause, vue comme une CNF) si l'on a $Mods(\varphi) \subseteq Mods(\varphi')$ (où les modèles sont pris sur l'ensemble de toutes les variables apparaissant dans φ ou dans φ') ; de la même manière, φ et φ'

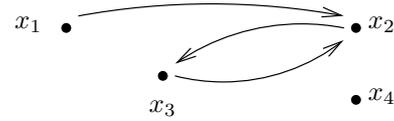


FIGURE 1 – Le graphe de dépendance G_{ex} de φ_{ex}

sont dites *logiquement équivalentes*, ce qu'on note $\varphi \equiv \varphi'$, si l'on a $Mods(\varphi) = Mods(\varphi')$.

Un *impliqué* d'une CNF φ (respectivement, d'un ensemble d'affectations M) est une clause C qui est impliquée par φ (respectivement, qui est satisfaite par chaque affectation $m \in M$). Un impliqué C de φ (respectivement, de M) est dit *premier* si aucune de ses sous-clauses propres $C' \subset C$ n'est un impliqué de φ (respectivement, de M). La conjonction de tous les impliqués premiers de φ (respectivement, de M) est notée $PI(\varphi)$ (respectivement, $PI(M)$) ; par définition, $PI(\varphi)$ et $PI(M)$ sont des CNF, et l'on a $\varphi \equiv PI(\varphi)$ et $M = Mods(PI(M))$. Une CNF est dite *première* si toutes ses clauses sont des impliqués premiers d'elle-même ; en particulier, les CNF $PI(\varphi)$ et $PI(M)$ sont toujours premières.

Une clause est dite *de Horn* si elle contient au plus un littéral positif, et *de Horn définie positive* si elle contient exactement un littéral positif. Une CNF est dite *de Horn (définie positive)* si toutes ses clauses sont de Horn (définies positives). Le *graphe de dépendance* $G(\varphi)$ d'une CNF de Horn φ est défini comme le graphe orienté (V, A) avec $V = \{x_1, \dots, x_n\}$, et $(x_i, x_j) \in A$ si et seulement s'il y a une clause contenant $\neg x_i$ et x_j dans φ . La CNF φ est alors dite *de Horn acyclique* si et seulement si le graphe $G(\varphi)$ est acyclique. Il est important de noter que $G(\varphi)$ peut être construit en temps linéaire en la taille de φ , et que l'on peut décider s'il est acyclique en temps linéaire.

Exemple 1 Soit la CNF

$$\varphi_{ex} = \{\{\neg x_1, x_2\}, \{\neg x_2, x_3\}, \{\neg x_1, \neg x_3, x_2\}, \{\neg x_1, \neg x_4\}\}$$

On voit facilement que φ_{ex} est de Horn ; son graphe de dépendance G_{ex} est le graphe orienté donné sur la figure 1. Ce graphe contient le circuit (x_2, x_3, x_2) , et donc φ_{ex} n'est pas une CNF de Horn acyclique.

3 Clôture transitive des graphes de dépendance

Le graphe de dépendance d'une CNF de Horn, tel que défini au paragraphe 2, dépend de la *syntaxe* de la formule. En d'autres termes, il est possible que deux CNF de Horn φ et φ' soient logiquement équivalentes mais n'aient pas le même graphe de dépendance. Il est même possible que le graphe de l'une soit acyclique alors que l'autre ne l'est pas, comme illustré par l'exemple suivant.

Exemple 2 (suite de l'exemple 1) La CNF φ_{ex} est logiquement équivalente à la CNF

$$\varphi'_{ex} = \{\{\neg x_1, x_2\}, \{\neg x_2, x_3\}, \{\neg x_1, \neg x_4\}\}$$

En effet, φ'_{ex} peut être obtenue de φ_{ex} en supprimant sa troisième clause, ce qui préserve l'équivalence logique puisqu'elle est subsumée par la première. Le graphe $G(\varphi'_{ex})$ n'a pas de circuit, donc φ'_{ex} est acyclique bien que φ_{ex} le ne soit pas et que l'on ait $\varphi_{ex} \equiv \varphi'_{ex}$.

Malgré cette propriété, en apparence négative, Boros *et al.* [2] montrent que l'acyclicité d'une CNF de Horn peut être définie d'une manière *structurelle*, c'est-à-dire de façon à ce qu'elle soit invariante par équivalence logique. Plus précisément, nous utiliserons la proposition suivante [2, théorème 4.3].

Proposition 3 Soient φ_1 et φ_2 deux CNF de Horn premières logiquement équivalentes. Les clôtures transitives de $G(\varphi_1)$ et de $G(\varphi_2)$ sont égales.

4 Ensembles de modèles

Nous donnons maintenant notre caractérisation des arcs du graphe de dépendance d'une CNF de Horn en fonction de son ensemble de modèles. En utilisant la proposition 3, nous pouvons en effet parler du graphe de dépendance d'un ensemble d'affectations, indépendamment des formules. Plus précisément, étant donné l'ensemble M de modèles d'une CNF de Horn, considérons son ensemble d'impliqués premiers $PI(M)$. La proposition 3 montre que $G(PI(M))$ est la clôture transitive de $G(\varphi)$ pour toute CNF première φ vérifiant $Mods(\varphi) = M$; en particulier, il y a au moins une CNF de Horn acyclique φ vérifiant $Mods(\varphi) = M$ si et seulement si $G(PI(M))$ est acyclique.

Exemple 4 (suite de l'exemple 2) L'ensemble d'affectations

$$M_{ex} = \{0000, 0001, 0010, 0011, 0110, 0111, 1110\}$$

vérifie $M_{ex} = Mods(\varphi_{ex})$. On peut voir que $PI(M_{ex})$ est l'ensemble de clauses $\{\{\neg x_1, x_2\}, \{\neg x_1, x_3\}, \{\neg x_1, \neg x_4\}, \{\neg x_2, x_3\}\}$, et nous concluons donc que le graphe $G(PI(M_{ex}))$ est acyclique.

Nous donnons maintenant nos deux résultats principaux. Le second précise le premier dans le but d'obtenir des algorithmes efficaces.

Proposition 5 Soit M l'ensemble de modèles d'une CNF de Horn. L'arc (x_i, x_j) est dans $G(PI(M))$ si et seulement si il existe un ensemble d'indices I avec $i, j \notin I$, et tel que les trois conditions suivantes soient vérifiées :

1. $\exists m_{11}^I \in M, m_{11}^I[i] = m_{11}^I[j] = 1$ et $m_{11}^I[I] = 1$
2. $\exists m_{00}^I \in M, m_{00}^I[i] = m_{00}^I[j] = 0$ et $m_{00}^I[I] = 1$
3. il n'y a pas d'affectation $m_{10}^I \in M$ vérifiant $m_{10}^I[i] = 1, m_{10}^I[j] = 0$ et $m_{10}^I[I] = 1$.

Preuve Par définition, (x_i, x_j) est un arc dans $G(PI(M))$ si et seulement si il y a une clause $C = \{\neg x_i, x_j, \dots\}$ dans $PI(M)$.

Supposons tout d'abord qu'il y a une telle clause, et soit I l'ensemble des indices de tous les littéraux négatifs de C , excepté x_i . Alors la condition 3 est vérifiée, car une telle affectation m_{10}^I ne satisfierait pas C . Ensuite, si la condition 1 n'était pas vérifiée, alors la clause $C \setminus \{x_j\}$ serait un impliqué de φ , donc C ne serait pas première, en contradiction avec la définition de $PI(M)$. La condition 2 se montre avec le même raisonnement appliqué à $C \setminus \{\neg x_i\}$.

Réciproquement, si la condition 3 est vérifiée, alors par définition d'un impliqué, il y a un impliqué de φ de la forme $\{\neg x_i, x_j, \dots\}$, pour lequel les indices de tous les littéraux négatifs, excepté $\neg x_i$, sont dans I . Par les conditions 1 et 2, ni x_i ni x_j ne peuvent être supprimés de cet impliqué, donc il existe un impliqué premier de φ de la forme désirée. \square

Proposition 6 Soit M l'ensemble de modèles d'une CNF de Horn. L'arc (x_i, x_j) est dans $G(PI(M))$ si et seulement si il existe deux affectations $m_{11}, m_{00} \in M$ telles que les trois conditions suivantes soient vérifiées :

1. $m_{11}[i] = m_{11}[j] = 1$
2. $m_{00}[i] = m_{00}[j] = 0$
3. il n'y a pas d'affectation $m_{10} \in M$ vérifiant $m_{10}[i] = 1, m_{10}[j] = 0$ et $m_{10}[ones(m_{11}, m_{00})] = 1$.

Preuve Supposons tout d'abord que (x_i, x_j) est un arc dans $G(PI(M))$. Soient I, m_{11}^I, m_{00}^I comme dans la proposition 5, et soient $m_{11} = m_{11}^I$ et $m_{00} = m_{00}^I$. Alors les conditions 1 et 2 de la proposition 5 impliquent les conditions 1 et 2 de la présente proposition. Elles impliquent également $I \subseteq ones(m_{11}, m_{00})$, et nous pouvons conclure en utilisant la condition 3 de la proposition 5.

Réciproquement, supposons qu'il existe m_{11}, m_{00} vérifiant les trois conditions. On voit alors facilement que $I = ones(m_{11}, m_{00})$, $m_{11}^I = m_{11}$ et $m_{00}^I = m_{00}$ vérifient les conditions de la proposition 5, ce qui permet de conclure directement. \square

Exemple 7 (suite de l'exemple 4) Les deux affectations $m_{11} = 1110$ et $m_{00} = 0000$ de M_{ex} attestent la présence de l'arc (x_1, x_2) dans $G(PI(M_{ex}))$, puisqu'il n'y a pas d'affectation m_{10} dans M_{ex} vérifiant $m_{10}[1] = 1$ et $m_{10}[2] = 0$. Par ailleurs, puisqu'il n'y a pas d'affectation m dans M_{ex} satisfaisant $m[1] = m[4] = 1$, nous pouvons conclure que ni (x_1, x_4) ni (x_4, x_1) ne sont des arcs dans $G(PI(M_{ex}))$.

5 Bornes supérieures de Horn

Les résultats donnés au paragraphe 4 ne sont pas très surprenants, puisque l'ensemble complet des modèles d'une CNF contient énormément d'information explicite. Nous montrons maintenant que, de façon bien plus intéressante, ces résultats se généralisent au cas où l'ensemble de modèles d'une CNF n'est pas donné explicitement, mais par un de ses sous-ensembles. Nous rappelons tout d'abord les notions utilisées.

Lorsqu'une CNF φ est de Horn, il est bien connu que l'ensemble d'affectations $Mods(\varphi)$ est clos par *Et* logique bit-à-bit [14, 4]; ceci signifie que pour toute paire de modèles $\{m, m'\}$ de φ , l'affectation $m \wedge m'$ définie par $ones(m \wedge m') = ones(m, m')$ est également dans $Mods(\varphi)$. C'est par ailleurs une caractérisation, en ce sens que si un ensemble d'affectations a cette propriété, alors c'est nécessairement l'ensemble des modèles d'au moins une CNF de Horn. Nous disons donc d'un ensemble d'affectations clos par *Et* logique bit-à-bit qu'il est *de Horn*.

Il s'ensuit qu'étant donné un ensemble d'affectations M , il existe un unique ensemble d'affectations qui est de Horn, qui inclut M et qui est minimal pour l'inclusion [4]. Cet ensemble peut être obtenu en calculant itérativement la clôture de M par *Et*, et nous l'appelons sa *borne supérieure (de Horn)* (Horn-LUB). Nous notons \overline{M} la Horn-LUB de M ¹.

Si M est de Horn, il existe de plus un unique ensemble d'affectations qui est inclus dans M , qui a M comme Horn-LUB, et qui est minimal pour l'inclusion. Cet ensemble est appelé l'ensemble des *modèles caractéristiques (de Horn)*, ou des *modèles extrémaux*, de M [16, 11], et nous le notons $Char(M)$.

Exemple 8 (suite de l'exemple 7) *L'ensemble d'affectations $Char_{ex} = \{0001, 0011, 0111, 1110\}$ est l'ensemble des modèles caractéristiques de l'ensemble M_{ex} . Il vérifie donc en particulier $\overline{Char_{ex}} = M_{ex}$. L'ensemble $\{0001, 0010, 0011, 0111, 1110\}$ a également M_{ex} comme Horn-LUB, mais ce n'est pas un ensemble de modèles caractéristiques, puisqu'il n'est pas minimal pour l'inclusion (une autre façon de le voir est que l'affectation 0010 peut être obtenue comme le *Et* de deux autres affectations de l'ensemble, en l'occurrence 0011 et 1110).*

De nombreux travaux se sont intéressés aux modèles caractéristiques [10, 16, 11, 6]. Cet intérêt provient du fait que la représentation d'une théorie de Horn par son ensemble de modèles caractéristiques peut économiser un espace exponentiel (même si cela n'est pas garanti et que l'inverse peut également se produire), tout en rendant plus facile bon nombre d'opérations.

1. Notons que l'expression « borne supérieure de Horn de M » désigne en général une CNF ayant pour ensemble de modèles la clôture de M . Toutefois, notre notation \overline{M} empêche toute confusion.

Nous montrons maintenant qu'étant donné un ensemble d'affectations M , toute l'information sur le graphe de dépendance de $PI(\overline{M})$ peut être retrouvée efficacement. En particulier, il s'avère qu'un ensemble de modèles caractéristiques est suffisant pour retrouver cette information. Soulignons que ceci n'était *a priori* pas garanti, car certains problèmes cessent d'être traitables efficacement lorsque l'on passe d'un ensemble complet d'affectations à l'ensemble correspondant de modèles caractéristiques; un exemple est le problème consistant à décider si une CNF de Horn a pour ensemble de modèles la Horn-LUB d'un ensemble donné d'affectations [12].

Proposition 9 (parallèle à la proposition 6) *Soit M un ensemble quelconque d'affectations. L'arc (x_i, x_j) est dans $G(PI(\overline{M}))$ si et seulement s'il existe des affectations $\mu_{11}, \mu_{00} \in M$ telles que les trois conditions suivantes soient vérifiées :*

1. $\mu_{11}[i] = \mu_{11}[j] = 1$
2. $\mu_{00}[i] = \mu_{00}[j] = 0$
3. *il n'y a pas d'affectation $\mu_{10} \in M$ vérifiant $\mu_{10}[i] = 1, \mu_{10}[j] = 0$ et $\mu_{10}[ones(\mu_{11}, \mu_{00})] = 1$.*

Preuve Supposons tout d'abord que (x_i, x_j) est un arc dans $G(PI(\overline{M}))$, et soient $m_{11}, m_{00} \in \overline{M}$ comme dans la proposition 6. Par définition de \overline{M} , il existe un sous-ensemble M_{11} de M , non vide, vérifiant $m_{11} = \bigwedge M_{11}$; puisque l'on a $m_{11}[i] = m_{11}[j] = 1$, on a $\forall m \in M_{11}, m[i] = m[j] = 1$, et nous définissons μ_{11} comme l'une quelconque de ces affectations m . De façon similaire, il y a un sous-ensemble M_{00} de M , non vide, vérifiant $m_{00} = \bigwedge M_{00}$. Puisque l'on a $m_{00}[i] = m_{00}[j] = 0$, on a soit (i) $\exists m \in M_{00}, m[i] = m[j] = 0$, soit (ii) $\exists m, m' \in M_{00}, m[i] = m'[j] = 0$ and $m[j] = m'[i] = 1$. Mais puisque par construction on a $\forall m \in M_{00}, ones(m) \supseteq ones(m_{00}) \supseteq ones(m_{11}, m_{00})$, la possibilité (ii) contredirait la condition 3 de la proposition 6; donc on a $\exists m \in M_{00}, m[i] = m[j] = 0$, et nous définissons μ_{00} comme une telle affectation m . Par construction, μ_{11} et μ_{00} satisfont les conditions 1 et 2. Enfin, puisque par construction on a $ones(m_{11}) \subseteq ones(\mu_{11})$ et $ones(m_{00}) \subseteq ones(\mu_{00})$, et puisque (par la condition 3 de la proposition 6) il n'y a pas d'affectation $m_{10} \in \overline{M}$ vérifiant $m_{10}[i] = 1, m_{10}[j] = 0$ et $m_{10}[ones(\mu_{11}, \mu_{00})] = 1$, il n'y a *a fortiori* pas de telle affectation m_{10} dans M , et la condition 3 est donc vérifiée.

Supposons maintenant qu'il existe des affectations $\mu_{11}, \mu_{00} \in M$ vérifiant les trois conditions de la proposition, et soient $m_{11} = \mu_{11}$ et $m_{00} = \mu_{00}$. Puisque l'on a $M \subseteq \overline{M}$, on a $m_{11}, m_{00} \in \overline{M}$; nous montrons que ces affectations vérifient également les conditions de la proposition 6. Les conditions 1 et 2 en sont évidemment satisfaites. Nous considérons donc la condition 3. Supposons par l'absurde qu'il existe une affectation $m_{10} \in \overline{M}$ vérifiant $m_{10}[i] = 1, m_{10}[j] = 0$ et $m_{10}[ones(m_{11}, m_{00})] = 1$.

1. Alors, comme ci-dessus il existe un ensemble d'affectations $M_{10} \subseteq M$ vérifiant $m_{10} = \bigwedge M_{10}$ et $\exists \mu_{10} \in M_{10}, \mu_{10}[i] = 1, \mu_{10}[j] = 0$ et $\mu_{10}[\text{ones}(m_{11}, m_{00})] = 1$; puisque l'on a $m_{11} = \mu_{11}$ et $m_{00} = \mu_{00}$, ceci contredit la condition 3. \square

Exemple 10 (suite de l'exemple 8) Les deux affectations $\mu_{11} = 1110$ et $\mu_{00} = 0001$ de Char_{ex} attestent la présence de l'arc (x_1, x_2) dans $G(\varphi_{ex})$. Observons que l'affectation 0000 mise en évidence dans l'exemple 7 n'est pas un modèle caractéristique, mais qu'elle peut être obtenue des affectations de Char_{ex} par $0000 = \bigwedge \{0001, 1110\}$.

Proposition 11 Soit M un ensemble quelconque d'affectations. Le graphe $G(\text{PI}(\overline{M}))$ peut être construit en temps cubique en la taille de M .

Preuve L'algorithme s'inspire directement de la caractérisation de la proposition 9. Pour chaque couple (x_i, x_j) de variables, l'algorithme décide si l'arc (x_i, x_j) est dans $G(\text{PI}(\overline{M}))$. Pour ce faire, il considère tous les couples (μ_{11}, μ_{00}) , et pour chacun il décide s'il y a dans M une affectation μ_{10} falsifiant la condition 3 de la proposition 9.

Notons m le nombre d'affectations dans M , et rappelons que n désigne le nombre de variables. Il y a $O(n^2)$ couples de variables, et pour chaque couple il y a $O(m^2)$ couples (μ_{11}, μ_{00}) à considérer. Étant donnés (x_i, x_j) et (μ_{11}, μ_{00}) , la présence d'une affectation μ_{10} telle que dans la proposition 9 peut être décidée en parcourant M une fois, en temps $O(mn)$. La complexité de l'algorithme est donc en $O(n^2 m^2 mn) = O(m^3 n^3)$, c'est-à-dire cubique en la taille de M . \square

6 Bornes supérieures de Horn définies positives

Rappelons qu'une CNF est dite *de Horn définie positive* si chacune de ses clauses contient exactement un littéral positif. On voit facilement que l'ensemble des modèles d'une CNF de Horn est celui d'une CNF de Horn *définie positive* si et seulement s'il contient l'affectation qui associe 1 à toutes les variables. Nous notons cette affectation $\vec{1}$. Le lemme suivant est évident, puisque l'affectation $\vec{1}$ ne peut pas être obtenue par le *Et* bit-à-bit d'autres affectations.

Lemme 12 Soit M un ensemble quelconque d'affectations. L'ensemble \overline{M} contient l'affectation $\vec{1}$ si et seulement si M la contient également.

Nous montrons maintenant que dans le cas défini positif, le graphe de dépendance de $\text{PI}(\overline{M})$ peut être obtenu plus efficacement que dans le cas général.

Proposition 13 Soit M un ensemble d'affectations contenant $\vec{1}$. Le graphe $G(\text{PI}(\overline{M}))$ peut être construit en temps $O(m^2 n^3)$, où m est le nombre d'affectations dans M .

Preuve Pour montrer le résultat, nous montrons qu'étant donné un couple de variables (x_i, x_j) , il y a un arc de x_i à x_j dans $G(\text{PI}(\overline{M}))$ si et seulement s'il existe une affectation $\mu_{00} \in M$ vérifiant $\mu_{00}[i] = \mu_{00}[j] = 0$, et telle qu'il n'y ait pas d'affectation $\mu_{10} \in M$ vérifiant $\mu_{10}[i] = 1, \mu_{10}[j] = 0$ et $\mu_{10}[\text{ones}(\vec{1}, \mu_{00})] = 1$ (c'est-à-dire $\mu_{10}[\text{ones}(\mu_{00})] = 1$). Nous renforçons donc la caractérisation de la proposition 9 en montrant qu'il suffit de considérer $\mu_{11} = \vec{1}$.

Tout d'abord, s'il existe une affectation μ_{00} comme ci-dessus, alors puisque $\mu_{11} = \vec{1}$ est dans M , la proposition 9 permet de conclure que l'arc (x_i, x_j) est dans $G(\text{PI}(\overline{M}))$. Réciproquement, si cet arc est dans $G(\text{PI}(\overline{M}))$, alors il existe des affectations μ_{11} et μ_{00} comme dans la proposition 9. En particulier, il n'y a pas d'affectation $\mu_{10} \in M$ vérifiant $\mu_{10}[i] = 1, \mu_{10}[j] = 0$ et $\mu_{10}[\text{ones}(\mu_{11}, \mu_{00})] = 1$. Mais on voit facilement que l'on a $\text{ones}(\mu_{11}, \mu_{00}) \subseteq \text{ones}(\mu_{00})$, donc il n'y a *a fortiori* pas d'affectation μ_{10} vérifiant $\mu_{10}[i] = 1, \mu_{10}[j] = 0$ et $\mu_{10}[\text{ones}(\mu_{00})] = 1$. Ceci suffit à conclure.

La complexité est établie comme suit. Soit de nouveau m le nombre d'affectations dans M . Pour chacun des $O(n^2)$ couples de variables (x_i, x_j) , il y a m affectations candidates μ_{00} , et pour chacune il suffit de décider la présence d'une affectation μ_{10} comme ci-dessus, en temps $O(mn)$ en parcourant M . \square

7 Bornes supérieures de Horn acycliques

En utilisant les résultats donnés aux paragraphes précédents, nous sommes maintenant en mesure de donner une caractérisation des ensembles d'affectations M tels que \overline{M} est l'ensemble de modèles d'une CNF de Horn acyclique. Les résultats qui suivent sont des corollaires directs des résultats précédents. Rappelons seulement que l'ensemble d'affectations \overline{M} est l'ensemble de modèles d'une CNF de Horn acyclique si et seulement si le graphe $G(\text{PI}(\overline{M}))$ est acyclique. Par ailleurs, ce graphe a un circuit (c'est-à-dire, n'est pas acyclique) si et seulement s'il a un circuit de taille 2 (de la forme (x_i, x_j, x_i)), puisqu'il est transitivement clos. Ce dernier point implique qu'il n'est pas nécessaire de construire le graphe $G(\text{PI}(\overline{M}))$ entièrement pour décider s'il est acyclique, et qu'il suffit de vérifier tous les couples de variables.

Proposition 14 Soit M un ensemble quelconque d'affectations. Il n'existe pas de CNF de Horn acyclique φ vérifiant $\overline{M} = \text{Mods}(\varphi)$ si et seulement s'il existe deux indices i, j et quatre affectations $\mu_{11}, \mu_{00}, \mu'_{11}, \mu'_{00} \in M$ vérifiant les quatre conditions suivantes :

1. $\mu_{11}[i] = \mu_{11}[j] = 1$ et $\mu_{00}[i] = \mu_{00}[j] = 0$
2. il n'existe pas d'affectation $\mu_{10} \in M$ satisfaisant $\mu_{10}[i] = 1, \mu_{10}[j] = 0$ et $\mu_{10}[\text{ones}(\mu_{11}, \mu_{00})] = 1$

3. $\mu'_{11}[i] = \mu'_{11}[j] = 1$ et $\mu'_{00}[i] = \mu'_{00}[j] = 0$
4. il n'existe pas d'affectation $\mu'_{01} \in M$ satisfaisant $\mu'_{01}[i] = 0, \mu'_{01}[j] = 1$ et $\mu'_{01}[\text{ones}(\mu'_{11}, \mu'_{00})] = 1$.

Proposition 15 Soit M un ensemble d'affectations contenant $\vec{1}$. Il n'existe pas de CNF de Horn acyclique φ vérifiant $\overline{M} = \text{Mods}(\varphi)$ si et seulement si il existe deux indices i, j et deux affectations $\mu_{00}, \mu'_{00} \in M$ vérifiant les quatre conditions suivantes :

1. $\mu_{00}[i] = \mu_{00}[j] = 0$
2. il n'existe pas d'affectation $\mu_{10} \in M$ satisfaisant $\mu_{10}[i] = 1, \mu_{10}[j] = 0$ et $\mu_{10}[\text{ones}(\mu_{00})] = 1$
3. $\mu'_{00}[i] = \mu'_{00}[j] = 0$
4. il n'existe pas d'affectation $\mu'_{01} \in M$ satisfaisant $\mu'_{01}[i] = 0, \mu'_{01}[j] = 1$ et $\mu'_{01}[\text{ones}(\mu'_{00})] = 1$.

Proposition 16 On peut décider, pour un ensemble d'affectations M donné (respectivement, contenant $\vec{1}$), si \overline{M} est l'ensemble de modèles d'une CNF de Horn acyclique en temps $O(m^3n^3)$ (respectivement, $O(m^2n^3)$), où m est le nombre d'affectations dans M . Ceci est vrai en particulier si M est un ensemble de modèles caractéristiques.

8 Pertinence pour l'abduction

Nous terminons en donnant une application de nos résultats à un problème lié à l'abduction. L'abduction est le processus consistant à expliquer une observation en fonction d'une théorie et d'un ensemble d'hypothèses. Il est bien connu que ce processus a de nombreuses applications, par exemple au diagnostic médical [3], à l'interprétation de textes en langue naturelle [9], ou à la configuration interactive de produits [1].

Nous adopterons la formalisation suivante du processus d'abduction.

Définition 17 Une instance du problème d'abduction est un triplet (Σ, H, q) , où Σ est une théorie, H est un sous-ensemble de ses variables et q (la manifestation) est l'une de ses variables. Une explication pour (Σ, H, q) est un sous-ensemble E de H tel que la formule $\Sigma \wedge \bigwedge E$ soit cohérente et implique q . Une explication E est dite minimale si il n'existe pas d'explication $E' \subset E$ pour (Σ, H, q) .

Le problème consistant à décider s'il existe une explication pour une instance donnée du problème d'abduction est bien connu comme très difficile pour une représentation de la théorie Σ par une formule. Plus précisément, il est Σ_2^P -complet si la théorie Σ est donnée en CNF, et NP-complet si elle est donnée par une CNF de Horn [5]. *A contrario*, ce problème peut être résolu en temps polynomial si la théorie est de Horn, et représentée par son ensemble de modèles caractéristiques [11].

Il s'avère que le problème de pertinence [3, 5] a la même complexité que le problème d'existence d'une explication, pour une représentation par des formules. Une hypothèse $h \in H$ est dite pertinente pour une instance (Σ, H, q) du problème d'abduction s'il existe au moins une explication minimale qui contient h [5].

Proposition 18 Le problème consistant à décider si une hypothèse $h \in H$ donnée est pertinente pour une instance (Σ, H, q) du problème d'abduction, où Σ est une CNF de Horn, est NP-complet. Il reste NP-complet même si Σ est restreinte à la classe des CNF de Horn définies positives.

Preuve Eiter et Gottlob montrent ce résultat pour le cas où les manifestations sont des conjonctions de variables [5, théorèmes 5.5 et 5.7]. Dans notre cadre où la manifestation est seulement une variable, l'appartenance à NP découle de leur résultat. Pour la difficulté, il suffit d'observer que le problème consistant à expliquer $q_1 \wedge \dots \wedge q_m$ pour une théorie Σ , se réduit au problème consistant à expliquer q pour la théorie $\Sigma \wedge (\neg q_1 \vee \dots \vee \neg q_m \vee q)$, où q est une variable n'apparaissant pas dans Σ . \square

Ce que nous montrons maintenant est que, *a contrario*, la pertinence d'une hypothèse donnée peut être décidée en temps polynomial si Σ est de Horn, et représentée par l'ensemble de ses modèles caractéristiques. Nous utiliserons le lemme, classique, suivant [13].

Lemme 19 Soit (Σ, H, q) une instance du problème d'abduction, et soit $E \subseteq H$. Alors E est une explication minimale pour (Σ, H, q) si et seulement si la clause $q \vee \bigvee_{x \in E} \neg x$ est un impliqué premier de Σ .

Ainsi, h est pertinente pour (Σ, H, q) si et seulement si il existe un impliqué premier de Σ dont toutes les variables sont dans l'ensemble $H \cup \{q\}$, et qui contient $\neg h$ et q . Pour traiter la condition que chaque variable soit dans $H \cup \{q\}$, nous définissons la projection $M|_V$ d'un ensemble M d'affectations aux variables $\{x_1, \dots, x_n\}$, sur un ensemble de variables $V \subseteq \{x_1, \dots, x_n\}$, comme l'ensemble d'affectations à V qui concordent avec au moins une affectation de l'ensemble M [18].

Exemple 20 (suite de l'exemple 4) Soit V l'ensemble $\{x_2, x_3\}$. La projection de M_{ex} sur V est l'ensemble d'affectations $\{00, 01, 11\}$.

On voit aisément que les impliqués de M qui ne contiennent que des variables de l'ensemble V sont exactement les impliqués de la projection de M sur V . On voit tout aussi aisément, en utilisant la clôture par *Et* bit-à-bit, que si M est la Horn-LUB de Σ et que V est un ensemble de variables, alors $M|_V$ est la Horn-LUB de $\Sigma|_V$. Nous obtenons alors le résultat suivant.

Proposition 21 *On peut décider si une hypothèse $h \in H$ est pertinente pour une instance (Σ, H, q) du problème d'abduction en temps $O(m^3n)$ (respectivement, $O(m^2n)$), si Σ est donnée par l'ensemble M de ses modèles caractéristiques (respectivement, et contient \bar{I}), où m est le nombre d'affectations dans M .*

Preuve L'algorithme découle directement des observations ci-dessus. Étant donné (M, H, q) , il projette tout d'abord l'ensemble de modèles caractéristiques M de Σ sur $H \cup \{q\}$, puis il décide s'il existe un impliqué premier de sa Horn-LUB $\overline{M}_{|H \cup \{q\}}$ qui contient à la fois $\neg h$ et q .

La projection de M peut être calculée en temps linéaire $O(mn)$, et il y a au plus m affectations dans l'ensemble $M_{|H \cup \{q\}}$ résultant. Par ailleurs, décider s'il existe un impliqué premier de $\overline{M}_{|H \cup \{q\}}$ qui contient à la fois $\neg h$ et q revient à décider si l'arc (h, q) est dans $G(PI(\overline{M}_{|H \cup \{q\}}))$, ce qui peut être fait en temps $O(m^3n)$ avec la caractérisation de la Proposition 9, et en temps $O(m^2n)$ comme dans la proposition 13 si M (et donc $M_{|H \cup \{q\}}$) contient \bar{I} . \square

9 Conclusion

Nous avons donné une caractérisation simple des arcs présents dans la clôture transitive du graphe de dépendance d'une CNF de Horn (première), en termes d'ensembles de modèles. Ce résultat nous a permis de proposer un algorithme polynomial pour construire le graphe de dépendance de l'ensemble de tous les impliqués premiers d'une théorie de Horn Σ , donnée par l'ensemble de ses modèles. De façon plus importante, cet algorithme fonctionne exactement de la même manière si la théorie Σ est donnée par un sous-ensemble M de ses modèles, dont la Horn-LUB est Σ .

Nous pensons que ce résultat est intéressant en soi. Nous en avons toutefois donné deux applications. Tout d'abord, l'existence d'une CNF de Horn acyclique, ayant pour ensemble de modèles la Horn-LUB d'un ensemble donné d'affectations, peut être décidée en temps polynomial. Ensuite, nous avons montré que la pertinence d'une hypothèse pour une instance du problème d'abduction peut être décidée en temps polynomial si la théorie est de Horn, et donnée par l'ensemble de ses modèles caractéristiques, contrairement au cas où elle est donnée par une CNF de Horn. Ce problème s'ajoute ainsi à la liste des problèmes qui peuvent être résolus plus efficacement pour une représentation par modèles que pour une représentation par formules.

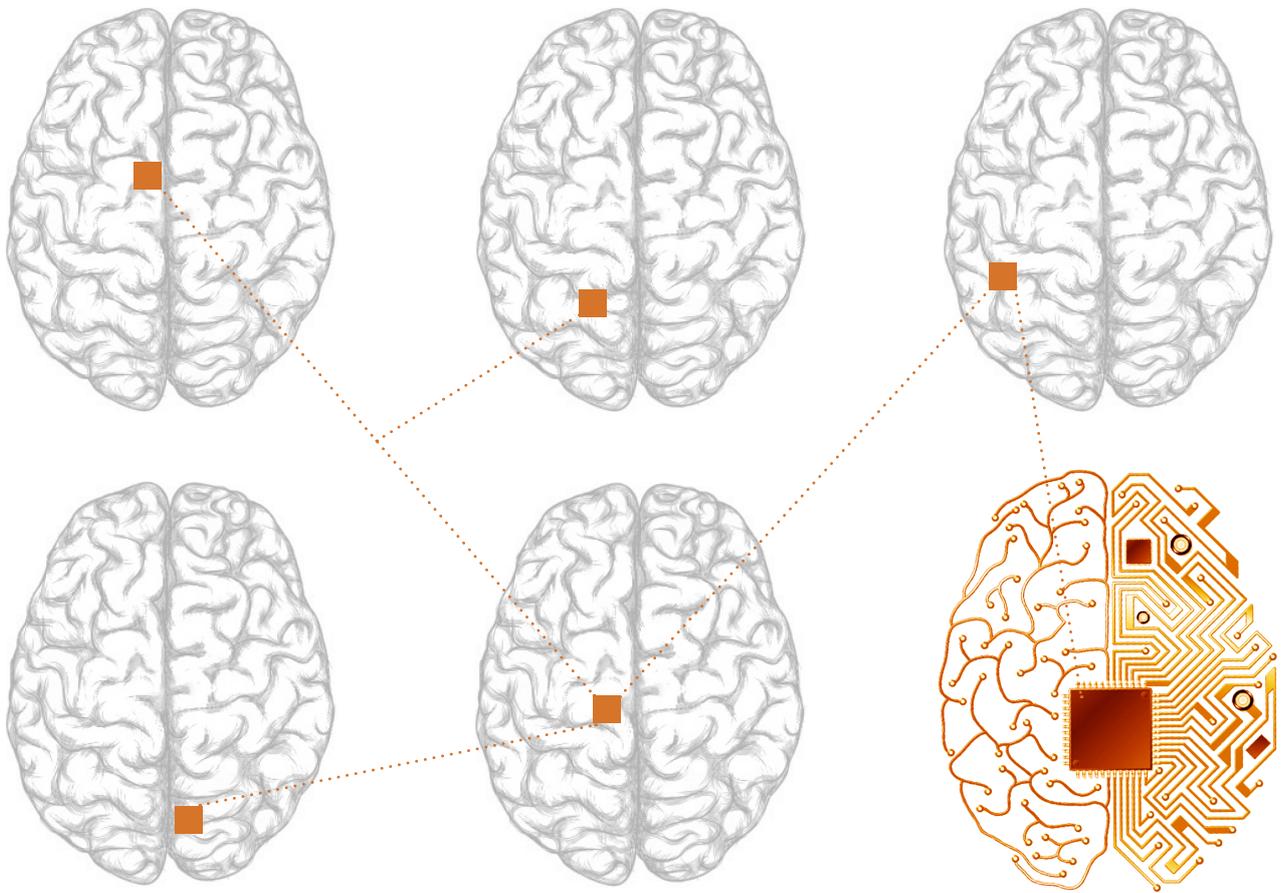
La principale perspective de ce travail est l'application de notre caractérisation à d'autres problèmes. Par exemple, Eiter and Makino ont montré que pour une représentation par des CNF, plusieurs problèmes relatifs à l'énumération des explications pour le problème d'abduction sont plus facile si la théorie (CNF de Horn) est acyclique [6]. Puisque

les problèmes correspondants sont difficiles pour une représentation par modèles, nous espérons que notre caractérisation permettra de tirer parti de l'hypothèse d'acyclicité dans cette représentation.

Références

- [1] J. Amilhastre, H. Fargier, and P. Marquis. Consistency restoration and explanations in dynamic CSPs — application to configuration. *Artificial Intelligence*, 135(1–2) :199–234, 2002.
- [2] E. Boros, O. Čepek, and A. Kogan. Horn minimization by iterative decomposition. *Annals of Mathematics and Artificial Intelligence*, 23(3–4) :321–343, 1998.
- [3] T. Bylander, D. Allemang, M.C. Tanner, and J.R. Josephson. Some results concerning the computational complexity of abduction. In *Proc. 1st International Conference on Principles of Knowledge Representation and Reasoning (KR'89)*, pages 44–54. Morgan Kaufmann, 1989.
- [4] R. Dechter and J. Pearl. Structure identification in relational data. *Artificial Intelligence*, 58 :237–270, 1992.
- [5] T. Eiter and G. Gottlob. The complexity of logic-based abduction. *Journal of the ACM*, 42(1) :3–42, 1995.
- [6] T. Eiter and K. Makino. On computing all abductive explanations from a propositional Horn theory. *Journal of the ACM*, 54(5), 2007.
- [7] P.L. Hammer and A. Kogan. Optimal compression of propositional Horn knowledge bases : complexity and approximation. *Artificial Intelligence*, 64 :131–145, 1993.
- [8] P.L. Hammer and A. Kogan. Quasi-acyclic propositional Horn knowledge bases : optimal compression. *IEEE Transactions on Knowledge and Data Engineering*, 7(5) :751–762, 1995.
- [9] J. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as abduction. *Artificial Intelligence*, 63 :69–142, 1993.
- [10] D. Kavvadias, C.H. Papadimitriou, and M. Sideri. On Horn envelopes and hypergraph transversals (extended abstract). In *Proc. 4th International Symposium on Algorithms And Computation (ISAAC'93)*, number 762 in Springer Lecture Notes in Computer Science, pages 399–405. Springer, 1993.
- [11] R. Khardon and D. Roth. Reasoning with models. *Artificial Intelligence*, 87 :187–213, 1996.

- [12] R. Khardon. Translating between Horn representations and their characteristic models. *Journal of Artificial Intelligence Research*, 3 :349–372, 1995.
- [13] P. Marquis. Consequence finding algorithms. In *Handbook of Defeasible Reasoning and Uncertainty Management Systems (DRUMS)*, volume 5, pages 41–145. Kluwer Academic, 2000.
- [14] J. McKinsey. The decision problem for some classes of sentences without quantifiers. *Journal of Symbolic Logic*, 8 :61–77, 1943.
- [15] M. Minoux. LTUR : a simplified linear-time unit resolution algorithm for Horn formulae and computer implementation. *Information Processing Letters*, 29(1) :1–12, 1988.
- [16] B. Selman and H. Kautz. Knowledge compilation and theory approximation. *Journal of the ACM*, 43(2) :193–224, 1996.
- [17] B. Selman and H.J. Levesque. Abductive and default reasoning : a computational core. In *Proc. 8th National Conference on Artificial Intelligence (AAAI'90)*, pages 343–348. AAAI Press, 1990.
- [18] B. Zanuttini. New polynomial classes for logic-based abduction. *Journal of Artificial Intelligence Research*, 19 :1–10, 2003.



PFIA 2015
<http://pfia2015.inria.fr>

Plate-forme Intelligence Artificielle
Rennes du 29 juin au 3 juillet 2015